Near-Optimal Regret Bounds for Contextual Combinatorial Semi-Bandits with Linear Payoff Functions*

**Kei Takemura,[1] Shinji Ito,[1] Daisuke Hatano,[2] Hanna Sumita,[3] Takuro Fukunaga,[4,2,5]
Naonori Kakimura,[6] Ken-ichi Kawarabayashi[7]**

[1] NEC Corporation
[2] RIKEN AIP
[3] Tokyo Institute of Technology
[4] Chuo University
[5] JST PRESTO
[6] Keio University
[7] National Institute of Informatics
{kei_takemura, i-shinji}@nec.com, daisuke.hatano@riken.jp, sumita@c.titech.ac.jp, fukunaga.07s@chuo-u.ac.jp,
kakimura@math.keio.ac.jp, k_keniti@nii.ac.jp

Abstract

The contextual combinatorial semi-bandit problem with linear payoff functions is a decision-making problem in which a learner chooses a set of arms with the feature vectors in each round under given constraints so as to maximize the sum of rewards of arms. Several existing algorithms have regret bounds that are optimal with respect to the number of rounds T. However, there is a gap of $\tilde{O}(\max(\sqrt{d}, \sqrt{k}))$ between the current best upper and lower bounds, where d is the dimension of the feature vectors, k is the number of the chosen arms in a round, and $\tilde{O}(\cdot)$ ignores the logarithmic factors. The dependence of k and d is of practical importance because k may be larger than T in real-world applications such as recommender systems. In this paper, we fill the gap by improving the upper and lower bounds. More precisely, we show that the C^{2}UCB algorithm proposed by Qin, Chen, and Zhu (2014) has the optimal regret bound $\tilde{O}(d\sqrt{kT} + dk)$ for the partition matroid constraints. For general constraints, we propose an algorithm that modifies the reward estimates of arms in the C^{2}UCB algorithm and demonstrate that it enjoys the optimal regret bound for a more general problem that can take into account other objectives simultaneously. We also show that our technique would be applicable to related problems. Numerical experiments support our theoretical results and considerations.

Introduction

This paper investigates the contextual combinatorial semi-bandit problem with linear payoff functions, which we call CCS problem (Qin, Chen, and Zhu 2014; Takemura and Ito 2019; Wen, Kveton, and Ashkan 2015). In this problem, a learner iterates the following process T times. First, the learner observes d-dimensional vectors, called *arms*, and a set of feasible combinations of arms, where the size of each combination is k. Each arm offers a reward defined by a common linear function over the arms, but the reward is not revealed to the learner at this point. Next, the learner chooses a feasible combination of arms. At the end, the learner observes the rewards of the chosen arms. The objective of the learner is to maximize the sum of rewards.

The CCS problem includes the linear bandit (LB) problem (Abbasi-Yadkori, Pál, and Szepesvári 2011; Agrawal and Goyal 2013; Auer 2002; Chu et al. 2011; Dani, Hayes, and Kakade 2008) and the combinatorial semi-bandit (CS) problem[1] (Chen et al. 2016a,b; Combes et al. 2015; Gai, Krishnamachari, and Jain 2012; Kveton et al. 2015; Wang et al. 2017; Wen, Kveton, and Ashkan 2015) as special cases. The difference from the LB problem is that, in the CCS problem, the learner chooses multiple arms at once. Moreover, while the given arms are fixed over the rounds and orthogonal to each other in the CS problem, they may be changed in each round and correlated in the CCS problem.

These differences enable the CCS problem to model more realistic situations of applications such as routing networks (Kveton et al. 2014), shortest paths (Gai, Krishnamachari, and Jain 2012; Wen, Kveton, and Ashkan 2015), and recommender systems (Li et al. 2010; Qin, Chen, and Zhu 2014; Wang et al. 2017). For example, when a recommender system is modeled with the LB problem, it is assumed that once a recommendation result is obtained, the internal predictive model is updated before the next recommendation. However, in a real recommender system, it is more common to update the predictive model after multiple recommendations, e.g., periodic updates (Chapelle and Li 2011). Such a situation can be modeled with the CCS problem, where the number of recommendations between the updates is k and the number of the updates is T (Takemura and Ito 2019)[2].

*We omit most of our proofs due to the page limit. The full version is available at https://arxiv.org/abs/2101.07957.

[1] Here, the CS problem denotes the problem of maximizing the sum of rewards (Combes et al. 2015; Kveton et al. 2014, 2015), while Chen et al. (2016a,b) deal with a more general objective.

[2] Strictly speaking, the LB problem with periodic updates is a

	Upper bound	Lower bound
The best known	$\tilde{O}(\max(\sqrt{d}, \sqrt{k})\sqrt{dkT})$ (Qin, Chen, and Zhu 2014; Takemura and Ito 2019)	$\Omega(\min(\sqrt{dkT}, kT))$ (Kveton et al. 2015)
This work	$\tilde{O}(d\sqrt{kT} + dk)$	$\Omega(\min(d\sqrt{kT} + dk, kT))$

Table 1: Regret bounds for CCS problem ($\tilde{O}(\cdot)$ ignores the logarithmic factors).

As in numerous previous studies on bandit algorithms, we measure the performance of an algorithm by its regret, which is the difference between the sum of the rewards of the optimal choices and that of the algorithm's choices. The existing regret bounds are summarized in Table 1, where $\tilde{O}(\cdot)$ means that the logarithmic factors are ignored. The best known upper bound on the regret is achieved by C^{2}UCB algorithm, which is given by Qin, Chen, and Zhu (2014). Takemura and Ito (2019) refined their analysis to improve the dependence on other parameters in the regret bound. The best lower bound is given for the CS problem by Kveton et al. (2015). Note that any lower bound for the CS problem is also a lower bound for the CCS problem, as the CCS problem covers the CS problem.

Although these regret upper and lower bounds match with respect to T, there is a gap of $\tilde{O}(\max(\sqrt{d}, \sqrt{k}))$ between them. In the literature on regret analysis, the degree of dependence on T in the regret bound usually draws much attention. However, for the CCS problem, the degree of dependence on k is also important because there are real-world applications of the CCS problem such that k is large. In recommender systems with periodic updates, for example, the number of recommendations between the updates could be large. An alternative example is the sending promotion problem, in which the number of users to send a promotion at once is much larger than the number of times to send the promotion, i.e., $k \gg T$ (Takemura and Ito 2019).

Our contribution is two-fold. First, we improve dependence on d and k in both the regret upper and lower bounds. Our upper and lower bounds match up to logarithmic factors. Second, we clarify a drawback of the UCB-type algorithms for other related problems and propose general techniques to overcome the drawback.

To improve the upper bound of the CCS problem, we first revisit the C^{2}UCB algorithm. This algorithm optimistically estimates rewards of arms using confidence intervals of estimates and then chooses a set of arms based on the optimistic estimates. Existing upper bounds have $k\sqrt{T}$ factor, which leads to the gap from the lower bound. In our analysis, however, we reveal that the linear dependence on k in the regret comes from the arms of large confidence intervals and obtain $\tilde{O}(d\sqrt{kT} + dk^2)$ regret by handling such arms separately. For further improvement, we focus on the case where the feasible combinations of arms are given by partition matroids. We show that the algorithm has the optimal regret bound in this case. Unfortunately, this analysis cannot apply to the general constraints, and we do not know whether the C^{2}UCB algorithm achieves the optimal regret upper bound. Instead, based on these analyses, we propose another algorithm that estimates the rewards of arms of large confidence intervals more rigorously; the algorithm divides the given arms into two groups based on their confidence intervals and underestimates the rewards of the arms with large confidence intervals. We show that the proposed algorithm enjoys the optimal regret bound for the CCS problem with any feasible combinations of arms, and is also optimal for a more general problem that can take into account both the sum of rewards and other objectives. For example, recommender systems often require diversity of recommended items (Qin and Zhu 2013; Qin, Chen, and Zhu 2014).

We support our theoretical analysis through numerical experiments. We first evaluate the performance of the algorithms on instances in which constraints are not represented by the partition matroid. We observe that the proposed algorithm is superior to the C^{2}UCB algorithm on these instances, which confirms our theoretical analysis that the C^{2}UCB algorithm may not achieve the optimal regret bound while our proposed algorithm does. We also evaluate the algorithms on instances with partition matroid constraints. For these instances, we observe that the C^{2}UCB and our proposed algorithms perform similarly.

Our theoretical and numerical analyses indicate that the sub-optimality of the C^{2}UCB algorithm arises from the combinatorial structure of the CCS problem, i.e., choosing a set of arms in each round. More precisely, the existence of an arm with a confidence interval that is too large makes the algorithm choose a bad set of arms. This is an interesting phenomenon that does not occur in the LB problem (the CCS problem when $k = 1$) or the case of partition matroid constraints. Since the technique we propose for the CCS problem is so general that it is independent of the linearity of the linear payoff functions, we believe it could be generalized to overcome the same issue for other semi-bandit problems.

Problem Setting

In this section, we present the formal definition of the CCS problem and the required assumptions. The CCS problem consists of T rounds. Let N denote the number of arms, and each arm is indexed by an integer in $[N] := \{1, 2, \ldots, N\}$. We denote by S_t a set of combinations of arms we can choose in the t-th round. We assume that each combination is of size k. Thus, $S_t \subseteq \{I \subseteq [N] \mid |I| = k\}$.

The learner progresses through each round as follows. At the beginning of the t-th round, the learner observes the set

little more restrictive than the CCS problem. However, most algorithms for the CCS problem, including the ones proposed in this paper, are applicable to the problem.

Algorithm 1 C^{2}UCB (Qin, Chen, and Zhu 2014)

```
Input: $\lambda > 0$ and $\{\alpha_t\}_{t\in[T]}$ s.t. $\alpha_t > 0$ for all $t \in [T]$.
 1: $V_0 \leftarrow \lambda I$ and $b_0 \leftarrow \mathbf{0}$.
 2: for $t = 1, 2, \ldots, T$ do
 3:     Observe $\{x_t(i)\}_{i\in[N]}$ and $S_t$.
 4:     $\hat{\theta}_t \leftarrow V_{t-1}^{-1} b_{t-1}$.
 5:     for $i \in [N]$ do
 6:         $\hat{r}_t(i) \leftarrow \hat{\theta}_t^\top x_t(i) + \alpha_t \sqrt{x_t(i)^\top V_{t-1}^{-1} x_t(i)}$.
 7:     end for
 8:     Play a set of arms $I_t = \mathrm{argmax}_{I\in S_t} \sum_{i\in I} \hat{r}_t(i)$ and
        observe rewards $\{r_t(i)\}_{i\in I_t}$.
 9:     $V_t \leftarrow V_{t-1} + \sum_{i\in I_t} x_t(i)x_t(i)^\top$ and $b_t \leftarrow b_{t-1} +$
        $\sum_{i\in I_t} r_t(i)x_t(i)$.
10: end for
```

of arms with the associated feature vectors $\{x_t(i)\}_{i\in[N]} \subseteq \mathbb{R}^d$ and the set of combinations of arms S_t. Then, the learner chooses $I_t \in S_t$. At the end of the round, the learner obtains the rewards $\{r_t(i)\}_{i\in I_t}$, where for all $i \in I_t$, $r_t(i) = {\theta^*}^\top x_t(i) + \eta_t(i)$ for some $\theta^* \in \mathbb{R}^d$ and $\eta_t(i) \in \mathbb{R}$ is a random noise with zero mean.

We evaluate the performance of an algorithm by the expected regret $R(T)$, which is defined as

$$R(T) = \sum_{t=1}^{T} \left(\sum_{i\in I_t^*} {\theta^*}^\top x_t(i) - \sum_{i\in I_t} {\theta^*}^\top x_t(i) \right),$$

where $I_t^* = \mathrm{argmax}_{I\in S_t} \sum_{i\in I} {\theta^*}^\top x_t(i)$.

As in previous work (Qin, Chen, and Zhu 2014; Takemura and Ito 2019), we assume the following:

Assumption 1. *$\forall t \in [T]$ and $\forall i \in I_t$, the random noise $\eta_t(i)$ is conditionally R-sub-Gaussian, i.e.,*

$$\forall \lambda \in \mathbb{R}, \mathbb{E}\left[\exp(\lambda \eta_t(i)) \mid \mathcal{F}_t\right] \le \exp\left(\lambda^2 R^2/2\right),$$

where $\mathcal{F}_t = \sigma\left(\{\{x_s(j)\}_{j\in I_s}\}_{s\in[t]}, \{\{\eta_s(j)\}_{j\in I_s}\}_{s\in[t-1]}\right)$.

In addition, we define the following parameters of the CCS problem: (i) $L > 0$ such that $\forall i \in [N]$ and $\forall t \in [T]$, $\|x_t(i)\|_2 \le L$, (ii) $S > 0$ such that $\|\theta^*\|_2 \le S$, and (iii) $B > 0$ such that $\forall i \in [N]$ and $\forall t \in [T]$, $|{\theta^*}^\top x_t(i)| \le B$. Note that LS is an obvious upper bound of B.

Regret Analysis of the C^{2}UCB Algorithm

Existing Analyses

Qin, Chen, and Zhu (2014) proposed the C^{2}UCB algorithm (Algorithm 1), which chooses a set of arms based on optimistically estimated rewards in a similar way to other UCB-type algorithms (Auer 2002; Chen et al. 2016b; Chu et al. 2011; Li et al. 2010).

The C^{2}UCB algorithm works as follows. At the beginning of each round, it constructs an estimator of θ^* using the arms chosen so far and its rewards (line 3). It then computes an optimistic reward estimator $\hat{r}_t(i)$ for each observed arm i (line 6), where $\alpha_t \sqrt{x_t(i)^\top V_{t-1}^{-1} x_t(i)}$ represents the size of the confidence interval of the estimated reward of arm i. Then, it chooses arms I_t obtained by solving the optimization problem based on $\{\hat{r}_t(i)\}_{i\in[N]}$ (line 8). Finally, it observes the reward of the chosen arms and updates the internal parameters b_t and V_t (line 9).

Qin, Chen, and Zhu (2014) showed that the algorithm admits a sublinear regret bound with respect to T. Takemura and Ito (2019) refined their analysis to improve the dependence on R, S, and L as follows. Here, for $\delta \in (0, 1)$, we define $\beta_t(\delta) = R\sqrt{d \log\left(\frac{1+L^2kt/\lambda}{\delta}\right)} + S\sqrt{\lambda}$.

Theorem 1 (Theorem 4 of Takemura and Ito (2019))**.** *If $\alpha_t = \beta_t(\delta)$ and $\lambda = R^2 S^{-2} d$, the C^{2}UCB algorithm has the following regret bound with probability $1 - \delta$:*

$$R(T) = \begin{cases} \tilde{O}\left(Rd\sqrt{kT}\right) & \text{if } \lambda \ge L^2 k \\ \tilde{O}\left(LSk\sqrt{dT}\right) & \text{otherwise.} \end{cases}$$

To prove Theorem 1, it suffices to bound the cumulative estimating error of rewards, i.e., $\sum_{t\in[T]} \sum_{i\in I_t} (\theta^* - \hat{\theta}_t)^\top x_t(i)$. Let $\|x_t(i)\|_{V_{t-1}^{-1}}$ denote $\sqrt{x_t(i)^\top V_{t-1}^{-1} x_t(i)}$ for all $i \in [N]$ and $t \in [T]$. To bound the error, Takemura and Ito (2019) showed that

$$\sum_{t\in[T]} \sum_{i\in I_t} (\theta^* - \hat{\theta}_t)^\top x_t(i) \le \beta_T(\delta) \sum_{t\in[T]} \sum_{i\in I_t} \|x_t(i)\|_{V_{t-1}^{-1}}. \tag{1}$$

The right-hand side is then bounded by the following lemma:

Lemma 1 (Lemma 5 of Takemura and Ito (2019))**.** *Let $\lambda > 0$. Let $\{\{x_t(i)\}_{i\in[k]}\}_{t\in[T]}$ be any sequence such that $x_t(i) \in \mathbb{R}^d$ and $\|x_t(i)\|_2 \le L$ for all $i \in [k]$ and $t \in [T]$. Let $V_t = \lambda I + \sum_{t'\in[t]} \sum_{i\in[k]} x_{t'}(i)x_{t'}(i)^\top$ for all $t \in [T]$. Then, we have*

$$\sum_{t\in[T]} \sum_{i\in[k]} \|x_t(i)\|_{V_{t-1}^{-1}} = \tilde{O}\left(L\sqrt{dk^2T/\lambda}\right).$$

This bound is tight up to logarithmic factors because we have $\sum_{t\in[T]} \sum_{i\in[k]} \|x_t(i)\|_{V_{t-1}^{-1}} = Ldk/\sqrt{\lambda}$ when $T = d$ and $x_t(i) = Le_t$ for all $i \in [k]$ and $t \in [T]$, where for all $l \in [d]$, $e_l \in \mathbb{R}^d$ is a vector in which the l-th element is 1 and the other elements are 0.

Improved Regret Bound

In this subsection, we improve the regret bound of the C^{2}UCB algorithm. A key observation of our analysis is that Lemma 1 is not tight for sufficiently large T. To improve Lemma 1, we divide $\{\{x_t(i)\}_{i\in[k]}\}_{t\in[T]}$ into two groups: the family of $x_t(i)$ such that $\|x_t(i)\|_{V_{t-1}^{-1}} \le 1/\sqrt{k}$, and the others. As shown in Lemma 2 below, the sum of $\|x_t(i)\|_{V_{t-1}^{-1}}$ in the former group is $\tilde{O}(\sqrt{dkT})$, which is smaller than Lemma 1. Moreover, the number of arms in the latter group is shown to be $\tilde{O}(dk)$, which means that not so many arms $x_t(i)$ have large $\|x_t(i)\|_{V_{t-1}^{-1}}$.

Lemma 2. Let $\lambda > 0$. Let $\{\{x_t(i)\}_{i\in[k]}\}_{t\in[T]}$ be any sequence such that $x_t(i) \in \mathbb{R}^d$ and $\|x_t(i)\|_2 \le L$ for all $i \in [k]$ and $t \in [T]$. Let $V_t = \lambda I + \sum_{t'\in[t]} \sum_{i\in[k]} x_{t'}(i) x_{t'}(i)^\top$ for all $t \in [T]$. Then, we have

$$\sum_{t\in[T]} \sum_{i\in[k]} \min\left(\frac{1}{\sqrt{k}}, \|x_t(i)\|_{V_{t-1}^{-1}}\right) = \tilde{O}\left(\sqrt{dkT}\right) \qquad (2)$$

and

$$\sum_{t\in[T]} \sum_{i\in[k]} \mathbb{1}\left(\|x_t(i)\|_{V_{t-1}^{-1}} > 1/\sqrt{k}\right) = \tilde{O}(dk). \qquad (3)$$

Based on Lemma 2, we can bound the right-hand side of (1) to obtain a better regret upper bound. The regret bound given by this theorem is optimal when $LS = B$.

Theorem 2. If $\alpha_t = \beta_t(\delta)$ and $\lambda = R^2 S^{-2} d$, the C^{2}UCB algorithm has the following regret bound with probability $1 - \delta$:

$$R(T) = \tilde{O}\left(Rd\sqrt{kT} + \min\left(LS, Bk\right) dk\right).$$

Proof sketch. Let $J_t = \{i \in [N] \mid \|x_t(i)\|_{V_{t-1}^{-1}} > 1/\sqrt{k}\}$ and $J'_t = I_t \cap J_t$. We separate chosen arms into two groups[3]: $\{J'_t\}_{t\in[T]}$ and the remaining arms. For $\{J'_t\}_{t\in[T]}$, replacing Lemma 1 with Lemma 2 in the proof of Theorem 1 gives the first term of the regret bound. There are two ways to bound the regret caused by the other group. In one way, we use the same proof as the former group, which obtains $\tilde{O}(LSdk)$. In the other way, by Lemma 2, we bound the number of rounds in which the arms of this group are chosen. Then, we have an upper bound of the regret in a round that is $2Bk$. Thus, we obtain $\tilde{O}(Bdk^2)$ in this way. The second term of the regret bound can be obtained by combining these two ways. □

Next, we show that Theorem 2 is better than Theorem 1. We first consider the case $\lambda \ge L^2 k$. From the definition of λ, we have $LSk\sqrt{dT} \le Rd\sqrt{kT}$. Since Theorem 1 implies $\tilde{O}(Rd\sqrt{kT})$ regret, it suffices to compare $LSk\sqrt{dT}$ with $\min(LS, Bk)dk$. If $T < d$, the C^{2}UCB algorithm has an obvious regret upper bound $2BkT$, which satisfies $\tilde{O}(LSk\sqrt{dT})$ and $\tilde{O}(\min(LS, Bk)dk)$; otherwise, we have $LSdk \le LSk\sqrt{dT}$. In the other case, Theorem 1 implies $\tilde{O}(LSk\sqrt{dT})$ regret and we have $Rd\sqrt{kT} \le LSk\sqrt{dT}$. Thus, it also suffices to compare $LSk\sqrt{dT}$ with $\min(LS, Bk)dk$. By the discussion in the first case, we obtain the desired result.

Improved Regret Bound for the CCS Problem with Partition Matroid Constraints

In this subsection, we show that the C^{2}UCB algorithm admits an improved regret upper bound for the CCS problem with the partition matroid constraint, that matches the regret lower bound shown in Table 1.

Now we define the partition matroid constraint. Let $\{B_t(j)\}_{j\in[M]}$ be a partition of $[N]$ into M subsets. Let $\{d_t(j)\}_{j\in[M]}$ be a set of M natural numbers. Then the partition matroid constraint S_t is defined from $\{B_t(j)\}_{j\in[M]}$ and $\{d_t(j)\}_{j\in[M]}$ as

$$S_t = \{I \subseteq [N] \mid |I \cap B_t(j)| = d_t(j), \forall j \in [M]\}. \qquad (4)$$

Such S_t is known as the set of the bases of a partition matroid. It is also known that linear optimization problems on a partition matroid constraint can be solved by the greedy algorithm. The class of S_t is so large that many fundamental classes are included. Indeed, the CCS problem with these constraints leads to the CCS problem with the uniform matroid constraints (i.e., the cardinality constraint) when $M = 1$ and $d_t(1) = k$ for all $t \in [T]$, and the LB problem with periodic updates when $M = k$ and $d_t(j) = 1$ for all $j \in [M]$ and $t \in [T]$.

We show that the C^{2}UCB algorithm achieves the optimal regret bound for the CCS problem with constraints satisfying (4):

Theorem 3. Assume that S_t is defined by (4) for all $t \in [T]$. Then, if $\alpha_t = \beta_t(\delta)$ and $\lambda = R^2 S^{-2} d$, the C^{2}UCB algorithm has the following regret bound with probability $1 - \delta$:

$$R(T) = \tilde{O}\left(Rd\sqrt{kT} + Bdk\right).$$

Proof sketch. Recall that I_t is the set of arms chosen by the C^{2}UCB algorithm in the t-th round. Let $J_t = \{i \in [N] \mid \|x_t(i)\|^2_{V_{t-1}^{-1}} > 1/k\}$ and $J'_t = I_t \cap J_t$. As in the proof of Theorem 2, we separate chosen arms into two groups: $I_t \setminus J'_t$ and J'_t. From the definition of I_t and J'_t, we obtain $I_t \setminus J'_t = \operatorname{argmax}_{I\in S'_t} \sum_{i\in I} \hat{r}_t(i)$ for all $t \in [T]$, where

$$S'_t = \{I \subseteq [N] \setminus J'_t \mid \forall j \in [M], |I \cap B_t(j)| = d_t(j) - |B_t(j) \cap J'_t|\}.$$

Let J^*_t be a subset of I^*_t that consists of the arms in $I^*_t \cap J'_t$, and $|B_t(j) \cap J'_t| - |I^*_t \cap J'_t \cap B_t(j)|$ arms chosen arbitrarily from $I^*_t \cap B_t(j)$ for each $j \in [M]$. Then, $I^*_t \setminus J^*_t \in S'_t$ and $|J^*_t| = |J'_t|$ for all $t \in [T]$. Similar to I_t, we divide I^*_t into $I^*_t \setminus J^*_t$ and J^*_t. This gives

$$\begin{aligned} R(T) = &\sum_{t\in[T]} \left(\sum_{i\in I^*_t \setminus J^*_t} {\theta^*}^\top x_t(i) - \sum_{i\in I_t \setminus J'_t} {\theta^*}^\top x_t(i)\right) \\ &+ \sum_{t\in[T]} \left(\sum_{i\in J^*_t} {\theta^*}^\top x_t(i) - \sum_{i\in J'_t} {\theta^*}^\top x_t(i)\right). \end{aligned}$$

The former term in the right-hand side of this equation is $\tilde{O}(Rd\sqrt{kT})$ by the optimality of $I_t \setminus J'_t$ and the discussion in the proof of Theorem 2. The latter term is $\tilde{O}(Bdk)$ by the definition of B and Lemma 2. □

Note that for the LB problem with periodic updates, the C^{2}UCB algorithm reduces to the LinUCB algorithm (Chu et al. 2011; Li et al. 2010) with periodic updates, and has the

[3] To show the regret bound of the LinUCB algorithm (Chu et al. 2011; Li et al. 2010), i.e., the C^{2}UCB algorithm for the case $k = 1$, Lattimore and Szepesvári (2020) take a similar approach in the note of exercise 19.3.

optimal regret bound. Note also that we can show a similar result for related problems if we have a UCB-type algorithm and an upper bound of the number of chosen arms that have large confidence bounds.

Proposed Algorithm

In this section, we propose an algorithm for a more general problem than the CCS problem. We will show the optimal regret bound of the proposed algorithm for the general problem.

First, let us define the general CCS problem. Let $X_t = \{x_t(i)\}_{i \in [N]}$ and $r_t^* = \{{\theta^*}^\top x_t(i)\}_{i \in [N]}$ for all $t \in [T]$. In this problem, the learner aims to maximize the sum of values $\sum_{t \in [T]} f_{r_t^*, X_t}(I_t)$ instead of the sum of rewards, where $f_{r_t^*, X_t}(I_t)$ measures the quality of the chosen arms. As in Qin, Chen, and Zhu (2014), we assume that the learner has access to an α-approximation oracle $\mathcal{O}_S(r, X)$, which provides $I \in S$ such that $f_{r,X}(I) \geq \alpha \max_{I' \in S} f_{r,X}(I')$ for some $\alpha \in (0, 1]$. Thus, we evaluate the performance of an algorithm by the α-regret $R^\alpha(T)$, which is defined as

$$R^\alpha(T) = \sum_{t=1}^{T} \left(\alpha f_{r_t^*, X_t}(I_t^*) - f_{r_t^*, X_t}(I_t) \right),$$

where $I_t^* = \sum_{i \in I} f_{r_t^*, X_t}(I)$. Note that the regret of the CCS problem is recovered if $\alpha = 1$ and $f_{r,X}(I)$ is the sum of rewards. We make the following assumptions that are almost identical to those in Qin, Chen, and Zhu (2014).

Assumption 2. For all $t \in [T]$ and $I \in S_t$, if a pair of rewards r and r' satisfies $r(i) \leq r'(i)$ for all $i \in [N]$, we have $f_{r, X_t}(I) \leq f_{r', X_t}(I)$.

Assumption 3. There exists a constant $C > 0$ such that for all $t \in [T]$, all $I \in S_t$, and any pair of rewards r and r', we have $|f_{r,X}(I) - f_{r',X}(I)| \leq C \sum_{i \in I} |r(i) - r'(i)|$.

The class of functions that satisfies the assumptions includes practically useful functions. For example, the sum of rewards with the entropy regularizer (Qin and Zhu 2013; Qin, Chen, and Zhu 2014), which has been applied to recommender systems in order to take into account both the sum of rewards and the diversity of the chosen arms, satisfies the assumptions with $C = 1$.

The proposed algorithm is described in Algorithm 2. When $f_{r,X}(I)$ is the sum of rewards, the difference between the C^{2}UCB and the proposed algorithms is the definition of $\hat{r}_t(i)$. We show the effectiveness of this difference. In the analysis of the C^{2}UCB algorithm, the regret can be decomposed as

$$R(T) = \sum_{t \in [T]} \left(\sum_{i \in I_t^*} {\theta^*}^\top x_t(i) - \sum_{i \in I_t} \hat{r}_t(i) \right) + \sum_{t \in [T]} \sum_{i \in I_t} \left(\hat{r}_t(i) - {\theta^*}^\top x_t(i) \right),$$

and the first term can be bounded by 0 since I_t is an optimal solution to the problem $\max_{I \in S_t} \sum_{i \in I} \hat{r}_t(i)$. Then, the right-hand side is bounded by

$$R(T) \leq \sum_{t \in [T]} \sum_{i \in I_t \setminus J_t'} (\hat{r}_t(i) - {\theta^*}^\top x_t(i)) + \sum_{t \in [T]} \sum_{i \in J_t'} (\hat{r}_t(i) - {\theta^*}^\top x_t(i)),$$

where we recall that $J_t' \subseteq I_t$ is the set of arms such that $\|x_t(i)\|_{V_{t-1}^{-1}} > 1/\sqrt{k}$. In the proof of Theorem 2, the first term of the right-hand side is shown to be $\tilde{O}(Rd\sqrt{kT})$, which is optimal, while the second term can be $\tilde{O}(\max(LS, Bk)dk)$. The reason the second term is so large is that each arm $i \in J_t'$ may have an overly optimistic reward estimate (i.e., $\hat{r}_t(i)$ may be large). To overcome this issue, we reduce $\hat{r}_t(i)$ when arm i has an overly optimistic reward estimate, keeping that the reduced value is an optimistic estimate required by UCB-type algorithms. As described in Algorithm 2, we adopt the maximum value of the average reward B as $\hat{r}_t(i)$ when $i \in J_t$.

Algorithm 2 Proposed algorithm

```
Input: $\lambda > 0$ and $\{\alpha_t\}_{t \in [T]}$ s.t. $\alpha_t > 0$ for all $t \in [T]$.
$V_0 \leftarrow \lambda I$ and $b_0 \leftarrow \mathbf{0}$.
for $t = 1, 2, \ldots, T$ do
    Observe $X_t = \{x_t(i)\}_{i \in [N]}$ and $S_t$, and let $J_t = \{i \in [N] \mid x_t(i)^\top V_{t-1}^{-1} x_t(i) > 1/k\}$.
    $\hat{\theta}_t \leftarrow V_{t-1}^{-1} b_{t-1}$.
    for $i \in [N]$ do
        If $i \in J_t$ then $\hat{r}_t(i) \leftarrow B$; otherwise $\hat{r}_t(i) \leftarrow \hat{\theta}_t^\top x_t(i) + \alpha_t \sqrt{x_t(i)^\top V_{t-1}^{-1} x_t(i)}$.
    end for
    Play a set of arms $I_t = \mathcal{O}_{S_t}(\{\hat{r}_t(i)\}_{i \in [N]}, X_t)$ and observe rewards $\{r_t(i)\}_{i \in I_t}$.
    $V_t \leftarrow V_{t-1} + \sum_{i \in I_t} x_t(i) x_t(i)^\top$ and $b_t \leftarrow b_{t-1} + \sum_{i \in I_t} r_t(i) x_t(i)$.
end for
```

Similar to the above, we can show that the proposed algorithm (Algorithm 2) has the following regret bound:

Theorem 4. If $\alpha_t = \beta_t(\delta)$ and $\lambda = R^2 S^{-2} d$, the proposed algorithm has the following regret bound with probability $1 - \delta$:

$$R^\alpha(T) = \tilde{O}\left(C \left(Rd\sqrt{kT} + Bdk \right) \right).$$

We show that this regret bound is optimal. We can define an instance of the general problem with any $C > 0$ from any instance of the CCS problem. Indeed, for any $C > 0$, we can define $f_{r,X}(I) = C \sum_{i \in I} r(i)$. Thus, the optimal degree of dependence on C in the regret is linear. For other parameters, we will show the lower bound in the next section.

Lower Bounds

In this section, we show the regret lower bound that matches the regret upper bound shown in Theorems 3 and 4 up to logarithmic factors. To achieve the lower bound, we mix two types of instances, which provide $\Omega(Rd\sqrt{kT})$ and $\Omega(Bdk)$ regret, respectively. While the first type of instance represents the difficulty of learning due to the noise added to the rewards, the second represents the minimum sample size required to learn the d-dimensional vector θ^* in the CCS problem.

We first consider instances that achieve $\Omega(Rd\sqrt{kT})$ and are analogous to the instances for the LB problem. Since the lower bound of the LB problem is known to be $\Omega(d\sqrt{T})$ with $R = 1$, the CCS problem in which the number of arms to select is kT would yield $\Omega(Rd\sqrt{kT})$. In these instances, the learner chooses k vertices from a d-dimensional hyper cube. Note that the duplication of vertices is allowed.

Theorem 5. Let $\{x_t(i)\}_{i=(s-1)2^d+1}^{s2^d} = \{-1, 1\}^d$ and $S_t = \{I \subseteq [k2^d] \mid |I| = k\}$ for any $s \in [k]$ and $t \in [T]$. Let $\Theta = \{-R/\sqrt{kT}, R/\sqrt{kT}\}^d$. Assume that $\eta_t(i) \sim \mathcal{N}(0, R^2)$ independently. Then, for any algorithm, there exists $\theta^* \in \Theta$ such that $R(T) = \Omega(Rd\sqrt{kT})$.

Proof. We first consider instances that achieve the lower bound of the LB problem. Using the discussion of Theorem 24.1 of Lattimore and Szepesvári (2020), we obtain the lower bound of $\Omega(Rd\sqrt{T})$ for a certain $\theta \in \Theta$ when $k = 1$. Note that this lower bound holds even if the algorithm knows in advance the given set of arms of all rounds.

Then, we observe that the set of algorithms for the above instances with kT rounds includes any algorithm for the CCS problem, which proves the theorem. □

We next introduce the instances of $\Omega(dk)$, based on the fact that no feedback can be received until k arms are selected in the CCS problem. More specifically, these instances consist of $\Theta(d)$ independent 2-armed bandit problems with delayed feedback. In each problem, the learner suffers $\Omega(Bk)$ regret due to the delayed feedback.

Theorem 6. Let $d = 2d'$ and $S_t = \{I \subseteq [2k] \mid |I| = k\}$. Assume that $\eta_t(i) = 0$. Define $\min(d', T)$ groups by dividing rounds. For each group $j \in [\min(d', T)]$, the given arms are defined as $x_t(i) = B\sqrt{d}e_{2j-1}$ for $i \leq k$ and $x_t(i) = B\sqrt{d}e_{2j}$ for $i > k$, where $\{e_l\}_{l\in[d]}$ is the normalized standard basis. Let $\Theta = \{-1/\sqrt{d}, 1/\sqrt{d}\}^d$. Then, for any algorithm, there exists $\theta^* \in \Theta$ such that $R(T) = \Omega(\min(Bdk, BkT))$.

Proof. As in Appendix A of Auer et al. (2002), it is sufficient to consider only the deterministic algorithms. In the first round of each group, any algorithm selects $k/2$ or more from one of the two types of arms. Therefore, we can choose $\theta^* \in \Theta$ so that for each group, the majority type of chosen arms is not optimal, in which case the algorithm suffers $\Theta(Bk)$ regret. □

Finally, by combining the two types of instance above, we have instances achieving the matching regret lower bound:

Theorem 7. Suppose that $kT = \Omega((Rd/B)^2)$ and $d = 2d'$. Then, for any given algorithm, if we use instances of Theorem 5 and Theorem 6 constructed using different d' dimensions in the first and second halves of the round, respectively, that instance achieves the following:

$$R(T) = \Omega(\min(Rd\sqrt{kT} + Bdk, BkT)).$$

Proof. From $kT = \Omega((Rd/B)^2)$, we have $|\theta^{*\top} x_t(i)| < B$ for all $i \in [N]$ and $t \in [T]$. Hence, we obtain $R(T) = O(BkT)$. Alternatively, from Theorem 5 and Theorem 6, we have $\Omega(Rd\sqrt{kT} + \min(Bdk, BkT))$. □

Note that we can set $R > 0$ and $B > 0$ arbitrarily in the instances of Theorem 7, but L and S are automatically determined as $L = O(\max(1, B\sqrt{d}))$ and $S = O(1)$.

Numerical Experiments

Setup

In this section, we evaluate the performance of the C^{2}UCB and the proposed algorithms through numerical experiments. Two types of instance are prepared: one in which the constraints are not represented by the partition matroid and one in thich they are. We call these types *grouped type* and *uniform matroid type*, respectively. Our analysis suggests that the C^{2}UCB algorithm performs well on the uniform matroid type only and that our proposed algorithm does well on both types. The aim of our experiments is to verify this.

Let us explain the details of the instances. The grouped type is given by combining the instances of Theorem 5 with $d = 4$ and $R = 1$ and an instance defined as follows. Suppose that $d = 3$, $N = 2k$, and $\theta^* = (0, 0.1, 0.9)^\top$. Let $f(t)$ be $t - k\lfloor t/k \rfloor$. The feature vectors are defined as

$$x_t(i) = \begin{cases} 2^{f(t)}e_1 & \text{if } i = 1 \\ e_2 & \text{if } 1 < i \leq k \\ e_3 & \text{if } i > k \end{cases}$$

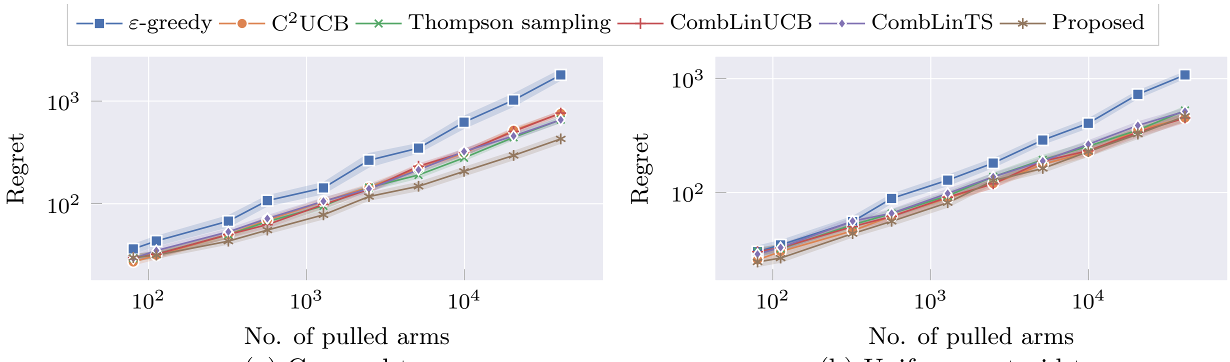
for all $t \in [T]$. The random noise $\eta_t(i)$ follows $\mathcal{N}(0, 1)$ independently for all $t \in [T]$ and $i \in [N]$. The feasible combinations are defined as $S_t = \{\{1, 2, \ldots, k\}, \{k+1, k+2, \ldots, 2k\}\}$ for all $t \in [T]$. Note that this is not represented by the partition matroid. As for the uniform matroid type, the feasible combinations are defined as $S_t = \{I \subseteq [N] \mid |I| = k\}$ for all $t \in [T]$. This is one of the uniform matroid constraints, which forms a subclass of partition matroid constraints. The other parameters are the same as the grouped type.

We start with $k = 2$ and $T = 40$, and increase k and T so that they satisfy $k = \Theta(T)$. We run 100 simulations to obtain the means of the regrets. We evaluate the performance of an algorithm by the means of the regrets for the worst θ^*: We compare the means for all θ^* for the largest kT and choose the θ^* with the largest mean.

We compare the proposed algorithm with five existing algorithms as baselines using the parameters described in Table 2. The ε-greedy algorithm has two ways of estimating the rewards of given arms: one is to use the values sampled from $\mathcal{N}(0, 1)$ independently, and the other is to estimate the rewards as in line 6 of Algorithm 1 with $\alpha_t = 0$. This algorithm chooses the former way with probability ε and the latter way otherwise. Then, it plays a set of arms as in line 8 of Algorithm 1.

Algorithm	Parameters
ε-greedy	$\varepsilon = 0.05$ and $\lambda = 1$
C^{2}UCB (Algorithm 1) (Qin, Chen, and Zhu 2014)	$\lambda = d$ and $\forall t, \alpha_t = \sqrt{d}$
Thompson sampling (Takemura and Ito 2019)	$\lambda = d$ and $\forall t, v_t = \sqrt{d}$
CombLinUCB (Wen, Kveton, and Ashkan 2015)	$\lambda = 1$, $\sigma = 1$, and $c = \sqrt{d}$
CombLinTS (Wen, Kveton, and Ashkan 2015)	$\lambda = 1$ and $\sigma = 1$
Proposed (Algorithm 2)	$\lambda = d$ and $\forall t, \alpha_t = \sqrt{d}$

Table 2: Algorithms in the numerical experiments.



(a) Grouped type

(b) Uniform matroid type

Figure 1: Experimental results.

Results

Figure 1(a) and (b) show the relation between the number of pulled arms (i.e., kT) and the regret for the grouped type and the uniform matroid type, respectively. Error bars represent the standard error.

As we can see in Figure 1(a), the regret of the proposed algorithm increased most slowly, which indicates that the regrets of the existing and proposed algorithms have different degrees of dependence on the number of pulled arms. We can explain this phenomenon from the viewpoint of the overly optimistic estimates of rewards. Since $\|x_t(1)\|_2$ increased exponentially until the k-th round, the C^{2}UCB algorithm often gave the arm an overly optimistic reward in these rounds. It follows from this optimistic estimate that the sum of optimistic rewards in the first group $\{1, 2, \ldots, k\}$ was often greater than that in the other group. Hence, the C^{2}UCB algorithm often chose the sub-optimal group and suffered $\Theta(Bk)$ regret in a round. Note that this phenomenon is almost completely independent of the linearity of the linear payoff function, which implies that the negative effect of the overly optimistic estimates could appear in UCB-type algorithms for related problems with semi-bandit feedback.

On the other hand, as shown in Figure 1(b), the regrets of all the algorithms except the ε-greedy algorithm were almost the same. This is because the constraints of the uniform matroid type satisfy the condition (4), and then the C^{2}UCB algorithm has the optimal regret bound described in Theorem 3. More precisely, as opposed to the grouped type, the regret suffered from the overly optimistic estimates is at most $\Theta(B)$ in a round.

Conclusion

We have discussed the CCS problem and shown matching upper and lower bounds of the regret. Our analysis has improved the existing regret bound of the C^{2}UCB algorithm and clarified the negative effect of the overly optimistic estimates of rewards in bandit problems with semi-bandit feedback. We have solved this issue in two ways: introducing partition matroid constraints and providing other optimistic rewards to arms with large confidence intervals. Our theoretical and numerical analyses have demonstrated the impact of the overly optimistic estimation and the effectiveness of our approaches.

As we discussed, the negative effect of the overly optimistic estimation could appear in related problems as well. Since the ideas of our approaches do not depend on the linearity of the linear payoff functions, we believe they are applicable to overly optimistic estimation in related problems.

Although the proposed algorithm achieves the optimal regret bound, it uses B explicitly as opposed to the C^{2}UCB algorithm. It is an open question whether there exists some algorithm that achieves the optimal regret bound for general constraints without knowledge of the tight upper bound of B.

Acknowledgements

SI was supported by JST, ACT-I, Grant Number JPMJPR18U5, Japan. TF was supported by JST, PRESTO, Grant Number JPMJPR1759, Japan. NK and KK were supported by JSPS, KAKENHI, Grant Number JP18H05291, Japan.

References

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, 2312–2320.

Agrawal, S.; and Goyal, N. 2013. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on Machine Learning*, 127–135.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3(Nov): 397–422.

Auer, P.; Cesa-Bianchi, N.; Freund, Y.; and Schapire, R. E. 2002. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing* 32(1): 48–77.

Chapelle, O.; and Li, L. 2011. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems*, 2249–2257.

Chen, W.; Hu, W.; Li, F.; Li, J.; Liu, Y.; and Lu, P. 2016a. Combinatorial multi-armed bandit with general reward functions. In *Advances in Neural Information Processing Systems*, 1659–1667.

Chen, W.; Wang, Y.; Yuan, Y.; and Wang, Q. 2016b. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research* 17(1): 1746–1778.

Chu, W.; Li, L.; Reyzin, L.; and Schapire, R. 2011. Contextual Bandits with Linear Payoff Functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 208–214.

Combes, R.; Shahi, M. S. T. M.; Proutiere, A.; et al. 2015. Combinatorial bandits revisited. In *Advances in Neural Information Processing Systems*, 2116–2124.

Dani, V.; Hayes, T. P.; and Kakade, S. M. 2008. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning Theory*, 355–366.

Gai, Y.; Krishnamachari, B.; and Jain, R. 2012. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking* 20(5): 1466–1478.

Kveton, B.; Wen, Z.; Ashkan, A.; Eydgahi, H.; and Eriksson, B. 2014. Matroid bandits: Fast combinatorial optimization with learning. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, 420–429.

Kveton, B.; Wen, Z.; Ashkan, A.; and Szepesvári, C. 2015. Tight regret bounds for stochastic combinatorial semi-bandits. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 535–543.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit Algorithms*. Cambridge University Press.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A Contextual-Bandit Approach to Personalized News Article Recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, 661–670.

Qin, L.; Chen, S.; and Zhu, X. 2014. Contextual Combinatorial Bandit and its Application on Diversified Online Recommendation. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, 461–469.

Qin, L.; and Zhu, X. 2013. Promoting diversity in recommendation by entropy regularizer. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

Takemura, K.; and Ito, S. 2019. An Arm-Wise Randomization Approach to Combinatorial Linear Semi-Bandits. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, 1318–1323.

Wang, Y.; Ouyang, H.; Wang, C.; Chen, J.; Asamov, T.; and Chang, Y. 2017. Efficient Ordered Combinatorial Semi-Bandits for Whole-Page Recommendation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2746–2753.

Wen, Z.; Kveton, B.; and Ashkan, A. 2015. Efficient Learning in Large-Scale Combinatorial Semi-Bandits. In *Proceedings of the 32nd International Conference on Machine Learning*, 1113–1122.