情報処理学会論文誌 Vol.56 No.10 1959–1967 (Oct. 2015)

動的報酬予算制限多腕バンディット問題とアルゴリズムの提案

新美 真[1,a)] 伊藤 孝行[1,b)]

受付日 2015年1月9日, 採録日 2015年7月1日

概要：本研究では，多腕バンディット問題を拡張した予算制限多腕バンディット問題を取り扱う．多腕バンディット問題とは，複数台あるスロットマシンをプレイするギャンブラを模した問題である．予算制限多腕バンディット問題は多腕バンディット問題の拡張の 1 つで，コストと予算による制約が存在する．既存の予算制限多腕バンディット問題では静的な報酬確率分布のみを仮定しており，動的な報酬確率分布については想定していない．本研究では予算制限多腕バンディット問題および予算制限バンディットアルゴリズムを拡張し，動的な報酬確率分布を想定する．予算制限多腕バンディット問題の拡張にともない，既存の予算制限バンディットアルゴリズムを拡張した D-KUBE および SW-KUBE を提案する．動的な報酬確率分布による問題空間を設定し，既存手法である KUBE と提案手法である D-KUBE および SW-KUBE との比較実験を行う．実験結果から動的な報酬確率分布において，提案手法である D-KUBE および SW-KUBE は既存手法である KUBE と比較して改善されることを確認する．

キーワード：多腕バンディット問題，強化学習

Budget-limited Multi Armed Bandit Problem with Dynamic Rewards and Proposing Algorithms

Makoto Niimi[1,a)] Takayuki Ito[1,b)]

Received: January 9, 2015, Accepted: July 1, 2015

Abstract: We focus on the budget-limited multi-armed bandit (BL-MAB) problems. In BL-MAB problems, the agent's actions are costly and constrained by a fixed budget. The existing BL-MAB problems assume the reward distributions are static. We assume the reward distributions are dynamic. It is more natural to assume the reward distributions are dynamic. For example, online advertisement's effect is dynamic for the trends and dates. Online advertisement is one of the real-world applications of BL-MAB problems. We make new bandit algorithms D-KUBE and SW-KUBE for dynamic situations. In our experiments, we compared the existing bandit algorithm with our proposed bandit algorithms. Experiment results show that D-KUBE and SW-KUBE are better than KUBE for the dynamic reward distributions.

Keywords: multi armed bandit, reinforcement-learning

1. はじめに

多腕バンディット問題とは，複数台あるスロットマシン（以降アームと呼ぶ）をプレイするギャンブラを模した問題である．アームから得られる報酬は，それぞれ独立で適当な確率分布に従うと仮定する．エージェントの目的は，決められたプレイ回数の中で得られる報酬を最大化することである．得られる報酬を最大化するために，エージェントはアームの探索（Exploration）と活用（Exploitation）をどのように行うかを求められる．探索とは，アームから得られる報酬が既知でない複数のアームを試行することである．探索を行うことで，より良いアームを選択するための情報を獲得する．活用とは，既知の情報をもとに良いアームを選ぶことである．活用を行うことで，探索して得られ

[1] 名古屋工業大学
Nagoya Institute of Technology, Nagoya, Aichi 466–8555, Japan
a) niimi.makoto@itolab.nitech.ac.jp
b) ito.takayuki@nitech.ac.jp

た情報を有効利用することが可能になる．しかし，探索に焦点を置くと正確な情報を得られるが，損失を生み出してしまう恐れがある．一方で，活用に焦点を置くとより良いアームを発見できない．探索と活用の間にあるトレードオフのことを探索と活用のジレンマと呼ぶ．バンディットアルゴリズムとはアームの探索と活用の均衡をとり，報酬を最大化するためのアルゴリズムである．

本研究では予算制限多腕バンディット問題の報酬確率分布を動的に拡張した，動的報酬予算制限多腕バンディット問題を提案する．また，既存の予算制限バンディットアルゴリズムでは動的な報酬確率分布を想定していない．したがって，提案する問題空間に適応した動的報酬予算制限バンディットアルゴリズムを提案する．

ここで予算制限多腕バンディット問題と動的報酬多腕バンディット問題について説明する．予算制限多腕バンディット問題とは，多腕バンディット問題に制約を持たせて拡張した問題の 1 つである．予算制限多腕バンディット問題は予算による制約が存在し，アームを選択する際に予算を消費しなければならない．エージェントは限られた予算の中で得られる報酬を最大化する．予算制限多腕バンディット問題の応用例として，オンライン広告，クラウドソーシング，およびワイヤレスセンサネットワークがあげられる．オンライン広告の応用では，検索キーワードに連動して表示される広告サービスであるスポンサードサーチのオークションにおいて，入札を行うためのバンディットアルゴリズムを提案している [1]．クラウドソーシングへの応用では，クラウドソーシングへの多腕バンディット問題の応用とアルゴリズムの提案を行っている [2]．また，実際にクラウドソーシングにおいて運用されている最新のアルゴリズムと比較して，同様の精度を保ちながらもコストをおさえることのできるアルゴリズムを提案している [3]．ワイヤレスセンサネットワークの応用では，ワイヤレスセンサネットワークの長期的な情報収集のためにバンディットアルゴリズムを応用した手法を提案している [4]．

動的報酬多腕バンディット問題とは，アームの持つ報酬確率分布が時間の経過により変動するという設定を持つ多腕バンディット問題である．動的報酬多腕バンディット問題の応用例として，Web サイトのリアルタイム最適化や，パケットルーティングネットワークへの応用があげられている [11]．Web サイトのリアルタイム最適化は，最も訪問者に評判の良いコンテンツを提供することを目的とする．訪問者が興味を持つコンテンツは，人の興味や関心が流行やトレンドによって左右されることが想定されるためである．したがって，時間の経過による変化に対応しつつ探索と活用により，どのコンテンツを提供すべきかを最適化する必要がある．

パケットルーティングネットワークへの応用では，遅延をいかに小さくするかを目的とする．しかし，パケットの遅延は，時間の経過にともない変化する．たとえば，ネットワーク帯域幅の割当てによって，パケットの伝送に遅延の差が生まれる．したがって，時間の経過による変化に対応しつつ，探索と活用により，どのルートを用いるかを最適化する必要がある．

既存の予算制限多腕バンディット問題における設定の課題は，報酬の確率分布が変化せず，静的であるという仮定を立てていることがあげられる．つまり，アームから得られる報酬の確率分布が 1 度決定されると以降は変化しない．しかし，現実世界の問題では，報酬の確率分布が変化し動的であることが想定される．

そこで本論文では動的な報酬確率分布を仮定し，動的報酬予算制限多腕バンディット問題と呼ぶ．例として，ある企業が自社の商品やキャンペーンを宣伝するためにオンライン広告を用いる場合を考える．これを予算制限多腕バンディット問題ととらえると，企業の資金，Web ページや Web サイト，オンライン広告を打ち出すために必要な金額，およびオンライン広告のクリック数，インプレッション数およびコンバージョン数などがそれぞれ，予算，アーム，コスト，および報酬にあたる．オンライン広告は，時間やイベントの影響および個人の嗜好の変化により得られる報酬が変化することが想定される．たとえば，本橋ら [5] はオンライン広告の 1 つであるバナー広告の効果について，トレンドおよび日付による影響により変動するとしている．

本研究では，既存の予算制限バンディットアルゴリズムである KUBE と，動的報酬予算制限バンディットアルゴリズムである D-KUBE および SW-KUBE の比較実験を行った．実験設定として，静的な報酬確率分布および動的な報酬確率分布を用いる．実験結果から，以下のつの知見が得られた．

- 静的な報酬確率分布では，KUBE と D-KUBE および SW-KUBE はほぼ同等の結果であった．
- 動的な報酬確率分布では，D-KUBE および SW-KUBE のほうが KUBE よりも良い結果が得られた．

以下に本論文の構成を示す．2 章では既存手法の概要について述べるとともに，提案手法との差について述べる．3 章では本論文の関連研究として，確率的多腕バンディット問題，予算制限多腕バンディット問題および動的報酬多腕バンディット問題について説明する．さらに，予算制限多腕バンディット問題および動的報酬多腕バンディット問題のバンディットアルゴリズムについて紹介する．4 章では本論文で提案する動的報酬による予算制限多腕バンディット問題，および動的報酬対応予算制限バンディットアルゴリズムである D-KUBE および SW-KUBE について述べる．5 章では実験設定および結果について述べる．最後に，本論文のまとめを示し，今後の課題を述べる．

2. 関連研究

既存手法として Knapsack based Upper confidence

Bound exploration and Exploitation（以降 KUBE と呼ぶ），LAKUBE，Budget Limited Epsilon-First（以降 BL-ϵ-first と呼ぶ），および Knapsack based Decreasing Epsilon-greedy（以降 KDE と呼ぶ）が存在する．本章では既存手法の概要を述べるとともに，本提案手法と既存手法との差について述べる．

KUBE は，活用と同時に探索も行う予算制限バンディットアルゴリズムである [9]．それぞれのアームを 1 度ずつプレイした後，得られた報酬から評価値を更新する．KUBE と本提案手法の異なる点として，評価値の更新部分が異なる．KUBE は評価値を算出する際に現在の試行までの平均値を用いているのに対して，本提案手法である D-KUBE および SW-KUBE は，減衰率および参照数を用いて報酬の重みを変動させている．評価値の更新部分を変更することにより，動的報酬確率分布における損失率を小さくすることを目的としている点が KUBE と本提案手法の差であるといえる．

LAKUBE は KUBE の探索部分を改善したアルゴリズムである [13]．探索するアーム数を制限することにより，予算の制約が厳しい問題条件下での損失を小さくすることを目的としている．LAKUBE と本提案手法の異なる点は，KUBE と同様で評価値の更新部分が異なる．LAKUBE は評価値を算出する際に現在の試行までの平均値を用いているのに対して，本提案手法である D-KUBE および SW-KUBE は，減衰率および参照数を用いて報酬の重みを変動させている．評価値の更新部分を変更することにより，動的報酬確率分布における損失率を小さくすることを目的としている点が LAKUBE と本提案手法の差であるといえる．

BL-ϵ-first は，探索係数 ϵ を用いて予算を分割する予算制限バンディットアルゴリズムである [8]．全体の予算 B のうち予算 ϵB を探索に用いて，残りの予算を活用に用いる．BL-ϵ-first と本提案手法の異なる点は，活用時における評価値の更新の有無である．BL-ϵ-first では探索での評価のみを用いており，活用では評価を変更することはない．提案手法では，活用時においても得られる報酬をフィードバックとして評価値を更新している．活用において得られる報酬をフィードバックとしている点が BL-ϵ-first と本提案手法の差であるといえる．

KDE は，アームをプレイするごとに減少する値 ϵ を用いて探索と活用の均衡をとるアルゴリズムである [12]．KDE では，最初はランダムにアームを選択するが，徐々にランダムに選択する確率を小さくして得られる報酬が大きくなるアームを選択する確率を増やす．KDE と本提案手法の異なる点は，動的報酬確率分布に対応していない点である．KDE はランダム性を徐々に減少させているために，十分に試行されると同じアームを繰り返しプレイする可能性がある．本提案手法である SW-KUBE および D-KUBE は，評価値の更新部分を変更することによって，報酬の重みを変更し報酬の変化に対応してアームの選択をすることができる．動的報酬確率分布における損失率を小さくすることを目的としている点が KDE と本提案手法の差であるといえる．

3. 多腕バンディット問題とアルゴリズム

3.1 多腕バンディット問題

多腕バンディット問題の起源は，治験計画に関するものが最初である [6]．その後，多腕バンディット問題は Robbins [7] により定式化された．近年は，制約や設定を加え，多腕バンディット問題を拡張した研究がなされている．

確率的多腕バンディット問題とは，多腕バンディット問題の中で標準的な多腕バンディット問題である．確率的多腕バンディット問題には，プレイ可能な K 本のアームが存在する．エージェントは，タイムステップごとにこれらのアームの中から 1 つをプレイする．エージェントは，アームをプレイすることにより報酬を獲得する．アームから得られる報酬はそれぞれ独立した，異なる確率分布に従う．エージェントの目的は，限られたプレイ回数 T の中で，受け取る報酬の合計を最大化することである．受け取る報酬の合計を最大化するために，エージェントはアームから受け取る報酬の期待値を最大化しなければならない．したがって，エージェントの目的は可能な限り少ない探索で，最も期待報酬の高いアームを見つけ繰り返しプレイすることである．

予算制限多腕バンディット問題とは，確率的多腕バンディット問題を拡張した多腕バンディット問題の 1 つである．確率的多腕バンディット問題と予算制限多腕バンディット問題との違いは，アームのプレイ回数が異なる点である．確率的多腕バンディット問題では，限られたプレイ回数 T の中で報酬の合計を最大化している．しかし，予算制限多腕バンディット問題には 2 つの制約があるためにプレイ回数が一意でない．2 つの制約とは，予算 B とコスト c である．予算制限多腕バンディット問題では，それぞれのアームに対してコストが設定される．エージェントは予算を所持しており，アームを選択する際に予算を消費しなくてはならない．エージェントは，アームにコストが設定された中で与えられた予算にしたがって探索および活用を行う．エージェントの目的は限られた予算の中で，利益を最大化することである．

動的報酬多腕バンディット問題とは，確率的多腕バンディット問題を拡張した多腕バンディット問題の 1 つである．確率的多腕バンディット問題と動的報酬多腕バンディット問題の違いは，アームの報酬確率分布の時間経過による変化の有無である．確率的多腕バンディット問題では，報酬確率分布が時間経過により変化しない中で報酬の合計を最大化している．動的報酬多腕バンディット問題で

は，変動期間 Υ が設定されており，アームの報酬確率分布が Υ 経過するごとに変化する設定がある．エージェントの目的は報酬分布の変化に対応し，報酬の合計を最大化することである．

3.2 予算制限バンディットアルゴリズム

KUBE

Knapsack based Upper confidence Bound exploration and Exploitation（以降 KUBE と呼ぶ）は，活用と同時に探索も行う予算制限バンディットアルゴリズムである [9]．KUBE を Algorithm 1 に記述する．各行の先頭の数字は step を表す．

Algorithm 1 The KUBE Algorithm

```
1: t = 1; B_t = B;
2: while pulling is feasible do
3:    if B_t < min_i c_i then
4:       STOP! { pulling is not feasible}
5:    end if
6:    if t ≤ K then
7:       Initial phase: play arm i(t) = t;
8:    else
9:       use density-ordered greedy to calculate M*(B_t) =
         {m*_{i,t}}, the solution of Equation 1;
10:      randomly pull i(t) with P(i(t) = i) = m*_{i,t} / Σ_{k=1}^{K} m*_{k,t};
11:   end if
12:   update the estimated upper bound of arm i(t);
13:   B_{t+1} = B_t − c_{i(t)}; t = t + 1;
14: end while
```

KUBE はそれぞれのタイムステップ t で，アームがプレイ可能かどうかを確認する（steps 3–4）．もしアームがプレイ可能である場合，KUBE は探索フェイズとしてそれぞれのアームを 1 度だけプレイする（steps 6–7）．その後タイムステップ $t > K$ で最も良いと思われるマシンの組合せを推定しアームをプレイする（steps 9–10）．推定には貪欲法を式 (1) に適用して求める．

$$\max \sum_{i=1}^{K} m_{i,t} \left(\hat{\mu}_{i,n_{i,t}} + \sqrt{\frac{2\log t}{n_{i,t}}} \right)$$
$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t} c_i \leq B_t, \forall i, t : m_{i,t} \text{ is integer} \qquad (1)$$

ここで $m_{i,t}$，$\hat{\mu}_{i,n_i}$，および n_i はそれぞれ式 (1) を満たすアームのプレイ回数，アーム i がプレイして得られた報酬の平均から求められた推定報酬，およびタイムステップ t までにアーム i をプレイした回数を表す．$\sqrt{2\log t/n_{i,t}}$ は，タイムステップ t でのアーム i の探索手当を意味する．特に，それぞれのタイムステップ s で選択されたアームを $i(s)$，得られた報酬を $r(s)$ とすると，推定報酬 $\hat{\mu}_{i,n_i}$ は式 $\hat{\mu}_{i,n_i} = \sum_{s=1}^{t} \mathbf{I}_{\{i(s)=i\}} r(s)/n_{i,t}$ を計算することで求められる．エージェントの目標は残りの予算 B_t に対応した KUBE の式 (1) を満たす解 $\{m_{i,t}\}_{i\in K}$ を見つけることである．ここで $\mathbf{I}_{\{i(s)=i\}}$ は，$i(s) = i$ となるとき 1 を返す指示関数である．この問題は NP-hard であるため，貪欲法を用いてアームの組合せを求めている．アーム i の期待報酬密度に基づく評価値は式 $\hat{\mu}_{i,n_{i,t}}/c_i + \sqrt{2\log t/n_{i,t}}/c_i$ により求められる．ここで KUBE の式 (1) を満たす最大となるアームの組合せの解を $M^*(B_t) = \{m^*_{i,t}\}$ とする．$\{m^*_{i,t}\}$ を用いて KUBE はランダムにプレイするアームを選択する．プレイされるアームの確率は式 $P(i(t) = i) = m^*_{i,t} / \sum_{k=1}^{K} m^*_{k,t}$ に従う．プレイされた後，選択されたアームの推定上限と残りの予算 B_t を更新する（steps 12–13）．

3.3 動的報酬バンディットアルゴリズム

D-UCB

Discounted UCB（以降 D-UCB と呼ぶ）は，減衰率 γ を用いて動的な報酬確率分布に適応させたものである [10]．アルゴリズムを Algorithm 2 に記述し，詳細を説明する．各行の先頭の数字は step を表す．

Algorithm 2 Discounted UCB

```
1: for t from 1 to K do
2:    play arm i(t) = t;
3: end for
4: for t from K + 1 to T do
5:    play arm i(t) = arg max_{1≤i≤K} μ̄_t(γ, i) + b_t(γ, i).
6: end for
```

ここで t はタイムステップを表す．D-UCB はそれぞれのアームを 1 度だけプレイする．その後タイムステップごとに，最も良いと推定されたアームをプレイする．即時的な期待報酬 $\bar{\mu}_t(\gamma, i)$ は，式 $\bar{\mu}_t(\gamma, i) = \sum_{s=1}^{t} \gamma^{t-s} r_s(i) \mathbf{I}_{\{i(s)=i\}}/n_t(\gamma, i)$ および式 $n_t(\gamma, i) = \sum_{s=1}^{t} \gamma^{t-s} \mathbf{I}_{\{i(s)=i\}}$ から求められる．

このとき，$n_t(\gamma, i)$，$r_s(i)$，および $\mathbf{I}_{\{i(s)=i\}}$ はそれぞれ，減衰率の和，タイムステップ s でアーム i から得られた報酬，および $i(s) = i$ となるとき 1 を返す指示関数である．$i(t)$ はタイムステップ t で選択されたアームである．つまり，最近得られた報酬に重みをつけて報酬の推定を行うことによって動的な報酬に適応している．$b_t(\gamma, i)$ は，減衰探索手当であり，式 $b_t(\gamma, i) = 2\sqrt{\xi \log n_t(\gamma)/n_t(\gamma, i)}$ から求められる．このとき $n_t(\gamma)$ は式 $n_t(\gamma) = \sum_{i=1}^{K} n_t(\gamma, i)$ より求められる．ξ は定数を設定する．ξ の値の範囲は，$\frac{1}{2} < \xi \leq 1$ である．

SW-UCB

Sliding-Window UCB（以降 SW-UCB と呼ぶ）は，参照数 τ を用いて動的な報酬確率分布に適応させたものである [11]．SW-UCB を Algorithm 3 に記述し，詳細を説明

Algorithm 3 Sliding-Window UCB

1: **for** t from 1 to K **do**
2: play arm $i(t) = t$;
3: **end for**
4: **for** t from $K+1$ to T **do**
5: play arm $i(t) = \arg\max_{1 \le i \le K} \bar{\mu}_t(\tau, i) + b_t(\tau, i)$.
6: **end for**

する．各行の先頭の数字は step を表す．

t はタイムステップを表す．SW-UCB は，それぞれのアームを 1 度だけプレイする．探索後タイムステップごとに，最も良いと推定されたアームをプレイする．即時的な期待報酬 $\bar{\mu}_t(\tau, i)$ は，式 $\bar{\mu}_t(\tau, i) = \sum_{s=t-\tau+1}^{t} r_s(i)\mathbf{I}_{\{i(s)=i\}}/n_t(\tau, i)$ および式 $n_t(\tau, i) = \sum_{s=t-\tau+1}^{t} \mathbf{I}_{\{i(s)=i\}}$ から求められる．

このとき，$n_t(\tau, i)$，$r_s(i)$，および $\mathbf{I}_{i(s)=i}$ はそれぞれ現在のタイムステップ t から $t-\tau+1$ までの選択回数，タイムステップ s でアーム i から得られた報酬，および $i(s) = i$ となるとき 1 を返す指示関数を表す．$i(t)$ はタイムステップ t で選択されたアームである．$b_t(\tau, i)$ は，減衰探索手当であり式 $b_t(\tau, i) = \sqrt{\xi \log(t \wedge \tau)/n_t(\tau, i)}$ によって表される．このとき $t \wedge \tau$ は，t および τ の最小値を意味する．ξ は，定数を設定する．ξ の値の範囲は，$\frac{1}{2} < \xi \le 1$ である．

4. 動的報酬予算制限多腕バンディット問題とアルゴリズム

4.1 動的報酬予算制限多腕バンディット問題

動的報酬予算制限多腕バンディット問題は，既存研究の予算制限多腕バンディット問題を拡張した多腕バンディット問題である．動的報酬予算制限多腕バンディット問題は，それぞれのアームにコストが設定されている，エージェントは，予算を所持しておりアームをプレイする際に，予算を消費しなくてはならない．さらにアームの報酬確率分布が，時間経過により変動する．動的報酬予算制限多腕バンディット問題において，エージェントの目的は，限られた予算の中で報酬確率分布の変動に適応し報酬の合計を最大化することである．

4.2 減衰率に基づく動的報酬予算制限バンディットアルゴリズムの提案

D-KUBE

Decreasing Knapsack based Upper confidence Bound exploration and Exploitation（以降 D-KUBE と呼ぶ）は，KUBE に D-UCB の推定報酬の算出方法を組み合わせたアルゴリズムである．D-KUBE は，D-UCB で用いる減衰率 γ を用いて，KUBE を動的な報酬確率分布に適応させる．D-KUBE アルゴリズムを Algorithm 4 に記述し，詳細に説明する．各行の先頭の数字は step を表す．

D-KUBE はそれぞれのタイムステップ t で，アームがプレイ可能かどうかを確認する（steps 3–4）．もし，アームがプレイ可能である場合，D-KUBE はそれぞれのアームを 1 度だけプレイする（steps 6–7）．その後，タイムステップ $t > K$ で，最も良いと思われるマシンの組合せを推定する．推定には，貪欲法を式 (2) に適用し，問題を解くことで求める．

Algorithm 4 The D-KUBE Algorithm

1: $t = 1; B_t = B$;
2: **while** pulling is feasible **do**
3: **if** $B_t < \min_i c_i$ **then**
4: STOP! { pulling is not feasible}
5: **end if**
6: **if** $t \le K$ **then**
7: Initial phase: play arm $i(t) = t$;
8: **else**
9: use density-ordered greedy to calculate $M^*(B_t) = \{m^*_{i,t}\}$, the solution of Equation 2;
10: randomly pull $i(t)$ with $P(i(t) = i) = \frac{m^*_{i,t}}{\sum_{k=1}^{K} m^*_{k,t}}$;
11: **end if**
12: update the estimated evaluation value of arm $i(t)$ by Equation 7;
13: $B_{t+1} = B_t - c_{i(t)}; t = t + 1$;
14: **end while**

$$\max \sum_{i=1}^{K} m_{i,t} \left(\bar{\mu}_t(\gamma, i) + b_t(\gamma, i)\right)$$
$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t} c_i \le B_t, \forall i, t : m_{i,t} \text{ is integer} \qquad (2)$$

ここで，$m_{i,t}$，$\bar{\mu}_t(\gamma, i)$，および $b_t(\gamma, i)$ はそれぞれ，式 (2) を満たすタイムステップ t でのアーム i のプレイ回数，タイムステップ t でのアーム i の即時的な期待報酬，およびタイムステップ t でのアーム i の減衰探索手当を表す．即時的な期待報酬 $\bar{\mu}_t(\gamma, i)$ は，式 (3) および式 (4) より求められる．

$$\bar{\mu}_t(\gamma, i) = \frac{1}{n_t(\gamma, i)} \sum_{s=1}^{t} \gamma^{t-s} r_s(i) \mathbf{I}_{\{i(s)=i\}} \qquad (3)$$

$$n_t(\gamma, i) = \sum_{s=1}^{t} \gamma^{t-s} \mathbf{I}_{\{i(s)=i\}} \qquad (4)$$

このとき，$n_t(\gamma, i)$，$r_s(i)$，および $\mathbf{I}_{\{i(s)=i\}}$ はそれぞれ，減衰率の和，タイムステップ s でアーム i から得られた報酬，および $i(s) = i$ となるとき 1 を返す指示関数を表す．$i(t)$ はタイムステップ t で選択されたアームである．また減衰探索手当 $b_t(\gamma, i)$ は式 (5) より求められる．

$$b_t(\gamma, i) = 2\sqrt{\frac{\xi \log n_t(\gamma)}{n_t(\gamma, i)}} \qquad (5)$$

このとき

$$n_t(\gamma) = \sum_{i=1}^{K} n_t(\gamma, i) \qquad (6)$$

ここで ξ は $0.5 < \xi \le 1$ となる定数を設定する．

目標は，余りの予算 B_t に対応した D-KUBE の式 (2) を満たす解 $\{m_{i,t}\}_{i\in K}$ を見つけることである．本問題は，NP 困難であるため貪欲法を用いてアームの準最適組合せを求めている．アーム i の期待報酬密度に基づく評価値は，式 (7) より求められる．

$$\frac{\bar{\mu}_t(\gamma,i)}{c_i}+\frac{b_t(\gamma,i)}{c_i} \tag{7}$$

ここで，D-KUBE の式 (2) を満たす最大となるアームの組合せの解を $M^*(B_t)=\{m^*_{i,t}\}$ とする．$\{m^*_{i,t}\}$ を用いて，D-KUBE はランダムにプレイするアームを選択する．プレイされるアームの確率は式 (8) に従う．

$$P(i(t)=i)=\frac{m^*_{i,t}}{\sum_{k=1}^{K} m^*_{k,t}} \tag{8}$$

プレイされた後選択されたアームの評価値と残りの予算 B_t を更新する（steps 12–13）．残り予算がなくなり，アームをプレイすることができなくなるまで繰り返される．

4.3 参照数に基づく動的報酬予算制限バンディットアルゴリズムの提案

SW-KUBE

Sliding-Window Knapsack based Upper confidence Bound exploration and Exploitation（以降 SW-KUBE と呼ぶ）は，KUBE に SW-UCB の推定報酬の算出方法を組み合わせたアルゴリズムである．SW-KUBE は SW-UCB で用いる参照数 τ を用いて，KUBE を動的な報酬確率分布に適応させる．SW-KUBE アルゴリズムを Algorithm 5 に記述し，詳細に説明する．各行の先頭の数字は step を表す．

Algorithm 5 The SW-KUBE Algorithm

1: $t=1; B_t=B$;
2: **while** pulling is feasible **do**
3: **if** $B_t<\min_i c_i$ **then**
4: STOP! { pulling is not feasible}
5: **end if**
6: **if** $t\le K$ **then**
7: Initial phase: play arm $i(t)=t$;
8: **else**
9: use density-ordered greedy to calculate $M^*(B_t)=\{m^*_{i,t}\}$, the solution of Equation 9;
10: randomly pull $i(t)$ with $P(i(t)=i)=\frac{m^*_{i,t}}{\sum_{k=1}^{K} m^*_{k,t}}$;
11: **end if**
12: update the estimated evaluation value of arm $i(t)$ by Equation 13;
13: $B_{t+1}=B_t-c_{i(t)}; t=t+1$;
14: **end while**

SW-KUBE はそれぞれのタイムステップ t で，アームがプレイ可能かどうかを確認する（steps 3–4）．もしアームがプレイ可能である場合，SW-KUBE はそれぞれのアームを 1 度だけプレイする（steps 6–7）．その後タイムステップ $t>K$ で，最も良いと思われるマシンの組合せを推定する．推定には貪欲法を式 (9) に適用して求める．

$$\max \sum_{i=1}^{K} m_{i,t}\left(\bar{\mu}_t(\tau,i)+b_t(\tau,i)\right)$$
$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t}c_i \le B_t, \forall i,t : m_{i,t} \text{ is integer} \tag{9}$$

ここで，$m_{i,t}$ は式 (9) を満たすアームのプレイ回数，$\bar{\mu}_t(\tau,i)$ は即時的な期待報酬，$b_t(\tau,i)$ は減衰探索手当を表す．即時的な期待報酬 $\bar{\mu}_t(\tau,i)$ は，式 (10) および式 (11) より求められる．

$$\bar{\mu}_t(\tau,i)=\frac{1}{n_t(\tau,i)}\sum_{s=t-\tau+1}^{t} r_s(i)\mathbf{I}_{\{i(s)=i\}} \tag{10}$$

$$n_t(\tau,i)=\sum_{s=t-\tau+1}^{t}\mathbf{I}_{\{i(s)=i\}} \tag{11}$$

このとき，$n_t(\tau,i)$，$r_s(i)$，および $\mathbf{I}_{i(s)=i}$ はそれぞれ，現在のタイムステップ t から $t-\tau+1$ までの選択回数，タイムステップ s でアーム i から得られた報酬，および $i(s)=i$ となるとき 1 を返す指示関数である．I_t はタイムステップ t で選択されたアームである．減衰探索手当 $b_t(\tau,i)$ は，式 (12) より求められる．

$$b_t(\tau,i)=\sqrt{\frac{\xi\log(t\wedge\tau)}{n_t(\tau,i)}} \tag{12}$$

このとき，$t\wedge\tau$ は，t および τ の最小値により表される．ここで ξ は $0.5<\xi\le 1$ となる定数を設定する．目標は，余りの予算 B_t に対応した SW-KUBE の式 (9) を満たす解 $\{m_{i,t}\}_{i\in K}$ を見つけることである．本問題は NP-hard であるため貪欲法を用いて，アームの準最適組合せを求めている．アーム i の期待報酬密度に基づく評価値は，

$$\frac{\bar{\mu}_t(\tau,i)}{c_i}+\frac{b_t(\tau,i)}{c_i} \tag{13}$$

より求められる．ここで SW-KUBE の式 (9) を満たす最大となるアームの組合せの解を $M^*(B_t)=\{m^*_{i,t}\}$ とする．$\{m^*_{i,t}\}$ を用いて SW-KUBE はランダムにプレイするアームを選択する．プレイされるアームの確率は式 (14) に従う．

$$P(i(t)=i)=\frac{m^*_{i,t}}{\sum_{k=1}^{K} m^*_{k,t}} \tag{14}$$

プレイされた後，選択されたアームの評価値と残りの予算 B_t を更新する（steps 12–13）．

5. 評価実験

5.1 実験設定

提案手法である D-KUBE および SW-KUBE の評価実験の設定について述べる．本論文の実験設定は，Tran-Thanh

らの実験設定に従う [9]．アームの数 K は 100 とし，アームの報酬の確率分布は，切断正規分布を用いる．切断正規分布の平均，分散，および定義域の設定について述べる．

- 平均 $\mu_i = [10, 20]$
- 分散 $\sigma_i = \frac{\mu_i}{2}$
- 定義域 $[0, 2\mu_i]$

平均 μ は，与えられた [10, 20] の範囲からランダムに選ばれる．分散および定義域は平均値が与えられることで求められる．アームのコストについては，[1, 10] の範囲からそれぞれのアームに対してランダムにコストを設定する．さらに本研究では，既存の問題設定である静的な報酬確率分布を含め以下のように Case 0 および Case 1 を設定する．

- Case 0：静的な報酬確率分布
- Case 1：アームごとに設定された間隔で再設定される動的な報酬確率分布

Case 0 は既存の問題設定と同様の設定を用いる．Case 1 はそれぞれのアームごとにランダムに変動期間を設定する．本評価実験において変動期間 Υ は [100, 200] とした．アームはタイムステップが設定された変動期間を経過するごとに，切断正規分布の平均値をランダムに設定し直す．

Case 0 および Case 1 を設定する理由を述べる．既存の問題設定である Case 0 を設定した理由として，提案手法が動的な報酬確率分布にのみ最適化されてしまい，静的な報酬確率分布で損失を生まないか確認するためである．動的な報酬確率分布として，Case 1 では一定間隔でアームが再設定され変化する変動を想定している．

D-KUBE および SW-KUBE の比較対象として，既存の予算制限バンディットアルゴリズムである KUBE, BL-ϵ-first, KDE，および LAKUBE を用いる．ここで各種バンディットアルゴリズムのパラメータについて述べる．D-KUBE の減衰率 γ および ξ の設定について述べる．D-KUBE の減衰率 γ は，既存の動的報酬バンディットアルゴリズムである D-UCB の問題設定をもとに設定する．D-UCB では，減衰率 γ を式 (15) に基づき設定していた．

$$\gamma = 1 - \frac{1}{4\sqrt{T}} \tag{15}$$

ここで T は総プレイ数を表す．

動的報酬多腕バンディット問題ではタイムステップ t が T に到達したときに終了する．しかし予算制限多腕バンディット問題では，ラウンド数が予算およびアームのコストに依存する．したがって，T を変更し，式 (16) および式 (17) にしたがって設定する．

$$\gamma = 1 - \frac{1}{4\sqrt{\frac{B}{c}}} \tag{16}$$

$$c = \frac{1}{K}\sum_{i=1}^{K} c_i \tag{17}$$

c はアームのコストの平均である．予算 B をコストの平均 c で割ることで T に近似させる．D-KUBE の ξ について，D-UCB と同様の値である 0.6 を設定した．

SW-KUBE の参照数 τ および ξ の設定について述べる．SW-KUBE の参照数 τ は，既存の動的報酬バンディットアルゴリズムである SW-UCB の問題設定をもとに設定する．

$$\tau = 4\sqrt{T \log T} \tag{18}$$

しかし D-KUBE と同様にラウンド数 T が予算およびアームのコストに依存する．したがって T を変更し式 (19) および式 (17) にしたがって設定する．

$$\tau = 4\sqrt{\frac{B}{c}\log\frac{B}{c}} \tag{19}$$

SW-KUBE の ξ は，D-KUBE と同様に，0.6 に設定した．

BL-ϵ-first，KDE，および LAKUBE のパラメータ設定について述べる．BL-ϵ-first では初期探索にどれくらいの予算を用いるのかを探索係数 ϵ によって決定している．本実験では，ϵ を 0.05，0.10 および 0.15 と設定した．KDE では探索と活用の比率を探索係数 ϵ_γ によって決定している．本実験では，ϵ_γ を 5，15 および 25 と設定した．LAKUBE では探索するアーム数を K_α によって決定している．本実験では結果が最も良くなる値をパラメータとして採用する．

5.2 実験結果

既存手法と提案手法である D-KUBE および SW-KUBE を比較した結果について述べる．実験結果の評価軸として，縦軸には損失率，横軸には予算を用いる．損失率は式 (20) から求める．

$$1 - \frac{total\_reward}{total\_reward^*} \tag{20}$$

$total\_reward$ は選択したアームの報酬確率分布の平均の合計，$total\_reward^*$ は，最適なアームの報酬確率分布の平均の合計を表す．最適なアームとは，コストあたりのアームの報酬確率分布の平均が最大となるものである．したがって，損失率が小さいほど，最適なアームを選択できているといえる．損失率は実験を 100 回繰り返した平均値を用いる．アームの報酬確率分布およびコストは 1 回ごとにランダムに設定される．

ここで，**図 1**，**図 2**，および**図 3** はそれぞれ，KUBE と LAKUBE，BL-ϵ-first および KDE と提案手法について Case0 において比較をした図である．

図 1 から分かるように，静的な報酬確率分布において拡張元である KUBE と提案手法である D-KUBE および SW-KUBE の間の損失率の差は小さい．したがって，D-KUBE および SW-KUBE は静的な報酬確率分布において KUBE と比較して損失率を大きくしないことが確認された．しかし，図 1，図 2，および図 3 との比較から，既存手法の方が損失率が下回っている場合が見られる．原因として，提

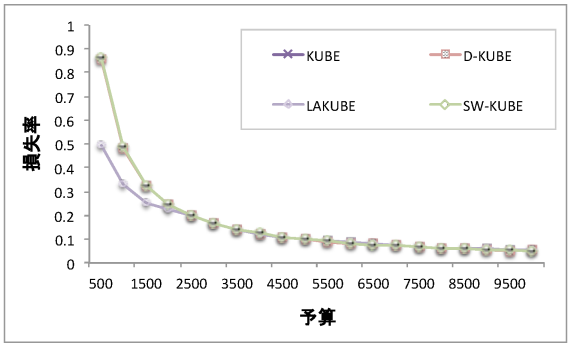

図 1 KUBE および LAKUBE と提案手法の比較（Case 0）
Fig. 1 Comparison of KUBE and LAKUBE with proposing methods: Case 0.

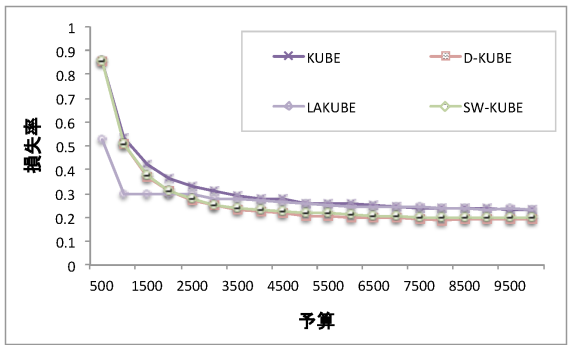

図 4 KUBE および LAKUBE と提案手法の比較（Case 1）
Fig. 4 Comparison of KUBE and LAKUBE with proposing methods: Case 1.

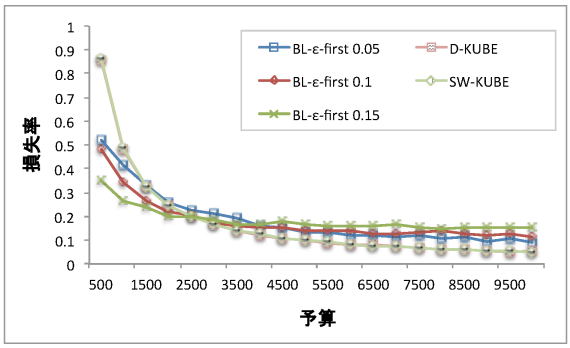

図 2 BL-ϵ-first と提案手法の比較（Case 0）
Fig. 2 Comparison of BL-ϵ-first with proposing methods: Case 0.

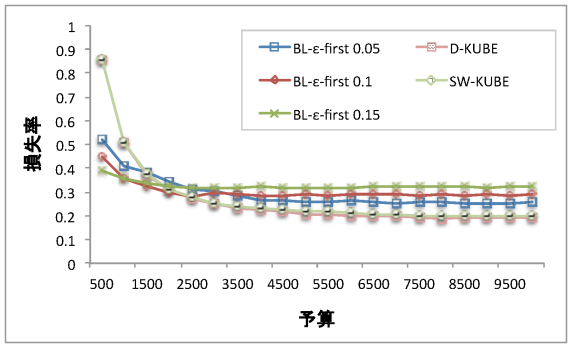

図 5 BL-ϵ-first と提案手法の比較（Case 1）
Fig. 5 Comparison of BL-ϵ-first with proposing methods: Case 1.

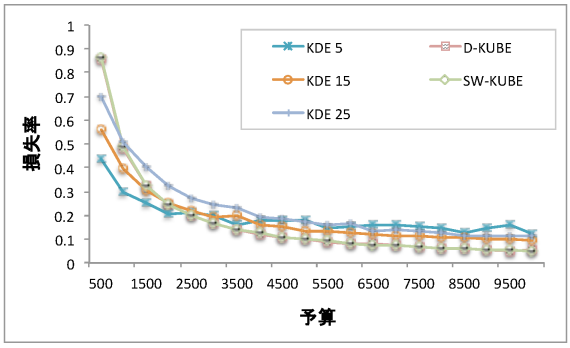

図 3 KDE と提案手法の比較（Case 0）
Fig. 3 Comparison of KDE with proposing methods: Case 0.

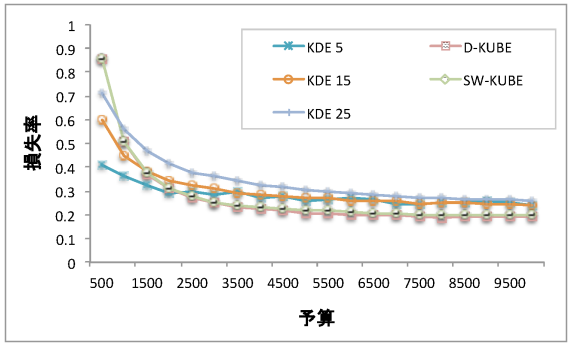

図 6 KDE と提案手法の比較（Case 1）
Fig. 6 Comparison of KDE with proposing methods: Case 1.

案手法では探索のときにすべてのアームを 1 度プレイする必要があり，予算が小さいときは活用時に予算を使用することができないためだと想定される．しかし，予算が大きくなってくると，提案手法の損失率が同等以下になっていることが分かる．

図 4，**図 5**，および**図 6** はそれぞれ，KUBE，BL-ϵ-first，KDE および LAKUBE と提案手法について Case1 において比較をした図である．

図 4 からは予算が大きくなるにつれて，KUBE と比較して提案手法の方が損失率が小さくなっていることが分かる．予算が 1000 以上のとき，有意水準 1%で有意差があり，提案手法の方が改善されていることを確認した．また，LAKUBE との比較では予算が小さいとき提案手法の方が損失率が上回っているが，予算が大きくなるにつれて，提案手法の損失率が下回っていることが分かる．特に，予算が 2500 以上のとき，有意水準 1%で有意差があり，提案手法の方が改善されていることを確認した．

図 5 からは予算が小さいときには BL-ϵ-first と比較して提案手法の方が損失率が上回っているが，予算が大きくなるにつれて，提案手法の損失率が下回っていることが分かる．特に，予算が 3000 以上のとき，有意水準 1%で有意差があり，提案手法の方が改善されていることを確認した．

図 6 からは予算が小さいときには KDE と比較して提案手法の方が損失率が上回っているが，予算が大きくなるにつれて，提案手法の損失率が下回っていることが分かる．特に，予算が 3000 以上のとき，有意水準 1%で有意差があり，提案手法の方が改善されていることを確認した．

以下に評価実験から得られた知見をまとめる．

- 静的な報酬確率分布では，KUBE と D-KUBE および SW-KUBE はほぼ同等の結果であった．
- 動的な報酬確率分布では，D-KUBE および SW-KUBE のほうが KUBE よりも良い結果が得られた．

6. おわりに

本研究で提案した D-KUBE および SW-KUBE は既存手法である KUBE と比較して，静的な報酬確率分布では損失率をあまり増加させず，動的な報酬確率分布では損失率が小さくなり改善された．今後の課題として，損失率を小さくするためにアルゴリズムを改善することがあげられる．また，本研究では人工的な実験設定を用いて，提案手法と既存手法の比較および評価を行った．より現実の問題に則した評価のために，具体的なデータなどを用いた評価を行うことも課題としてあげられる．

参考文献

[1] Tran-Thanh, L. et al.: Efficient regret bounds for online bid optimisation in budget-limited sponsored search auctions, *30th Conference on Uncertainty in Artificial Intelligence*, Vol.58, pp.527–535 (2014).

[2] Tran-Thanh, L. et al.: Efficient Budget Allocation with Accuracy Guarantees for Crowdsourcing Classification Tasks, *Proc. 2013 international conference on Autonomous agents and multi-agent systems*, pp.901–908 (2013).

[3] Tran-Thanh, L. et al.: BudgetFix: Budget limited crowdsourcing for interdependent task allocation with quality guarantees, *Proc. 2014 international conference on Autonomous agents and multi-agent systems*, pp.477–484 (2014).

[4] Tran-Thanh, L., Alex R. and Jennings, N.R.: Long-term information collection with energy harvesting wireless sensors: A multi-armed bandit based approach, *Autonomous Agents and Multi-Agent Systems*, pp.352–394 (2012).

[5] 本橋永至ほか：状態空間モデルによるインターネット広告のクリック率予測，オペレーションズ・リサーチ：経営の科学=Operations research as a management science research 57.10, pp.574–583 (2012).

[6] Thompson, W.R.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples, *Biometrika*, Vol.25, No.3/4, pp.285–294 (1933).

[7] Robbins, H.: Some aspects of the sequential design of experiments, *Bulletin of the American Mathematical Society*, Vol.58, No.5, pp.527–535, (1952).

[8] Tran-Thanh, L. et al.: Epsilon-First Policies for Budget-Limited Multi-Armed Bandits, *24th AAAI Conference on Artificial Intelligence*, pp.1211–1216 (2010).

[9] Tran-Thanh, L. et al.: Knapsack based optimal policies for budget-limited multi-armed bandits, *26th AAAI Conference on Artificial Intelligence*, pp.1134–1140 (2012).

[10] Kocsis, L. and Csaba, S.: Discounted ucb, *2nd PASCAL Challenges Workshop* (2006).

[11] Garivier, A. and Moulines, E.: On upper-confidence bound policies for switching bandit problems, *Algorithmic learning theory* (*ALT'11*), pp.174–188 (2011).

[12] Tran-Thanh, L.: Budget-Limited Multi-Armed Bandits, University of Southampton, Faculty of Physical and Applied Science, Doctoral Thesis (2012).

[13] Kadono, Y. and Fukuta, N.: LAKUBE: An Improved Multi-armed Bandit Algorithm for Strongly Budget-Constrained Conditions on Collecting Large-Scale Sensor Network Data, *Proc. 13th Pacific Rim International Conference on Artificial Intelligence* (*PRICAI2014*), pp.1089–1095 (2014).

新美 真 （学生会員）

平成 27 年名古屋工業大学大学院情報工学専攻入学．同大学院在学中．人工知能学会学生会員．

伊藤 孝行 （正会員）

平成 12 年名古屋工業大学大学院工学研究科博士後期課程修了．博士（工学）．平成 11 年 JSPS 特別研究員．平成 12 年 USC/ISI 客員研究員．平成 13 年北陸先端科学技術大学院大学助教授．平成 15 年名古屋工業大学大学院情報工学専攻助教授．平成 17 年ハーバード大学と MIT 客員研究員．平成 18 年名古屋工業大学大学院産業戦略工学専攻准教授．平成 20 年 MIT 客員研究員．平成 21 年 JST さきがけ大挑戦型研究員．平成 22 年東京大学客員研究員．平成 26 年より名古屋工業大学大学院産業戦略工学専攻/情報工学教育類教授，現在に至る．平成 23 年内閣府最先端・次世代研究開発プロジェクト代表研究者．平成 26 年日本学術振興会賞受賞．平成 25 年文部科学大臣表彰科学技術賞受賞．平成 19 年文部科学大臣表彰若手科学者賞受賞．情報処理学会長尾真記念特別賞受賞．平成 18 年 AAMAS2006 最優秀論文賞受賞．平成 17 年日本ソフトウェア科学会論文賞受賞．平成 16 年度 IPA 未踏ソフトウェア創造事業スーパークリエータ認定．マルチエージェントシステム国際財団（IFAAMAS）理事，ACM および IEEE 上級会員．