2R-08

# 予算制限多腕バンディット問題の動的報酬への拡張とアルゴリズムの提案

新美　真†　　伊藤　孝行‡

† 名古屋工業大学工学部情報工学科　　‡ 名古屋工業大学大学院産業戦略工学専攻

## 1　はじめに

本研究では，BL-MAB 問題を拡張し，動的な報酬確率分布を仮定する．また既存のバンディットアルゴリズムを拡張する．多腕バンディット（MAB）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ー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．ギャンブラーの役割をするエージェントの目的は，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探索と活用が求められる．探索は既知でない複数のアームを試行することで，活用は既知の情報をもとに良いアームを選択することである．本研究では MAB 問題の拡張の一つである予算制限多腕バンディット（BL-MAB）問題を取り扱う．BL-MAB 問題の制約としてコスト及び予算が存在する．エージェントはアームをプレイする時にコスト分の予算を消費する．

既存の BL-MAB 問題は静的な報酬確率分布を仮定している．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．例えば，BL-MAB 問題の応用例の一つであるオンライン広告では，トレンドや日付による広告効果の変動がある [1].

本論文の構成を以下に示す．第２章は既存の BL-MAB 問題のアルゴリズムについて述べる．第３章では提案手法である D-KUBE 及び SW-KUBE について言及する．第４章では実験設定及び結果について述べる．最後に本論文をまとめる．

## 2　予算制限バンディットアルゴリズム

Knapsack based Upper confidence Bound exploration and Exploitation（KUBE）は，活用と同時に探索を行う予算制限バンディットアルゴリズムである [2]．KUBE は探索時にすべてのアームを一度プレイする．活用時にプレイされるアームは式 (1) を最大化するアームである．

$$\max \sum_{i=1}^{K} m_{i,t}\left(\hat{\mu}_{i,n_{i,t}} + \sqrt{\frac{2\ln t}{n_{i,t}}}\right)$$

$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t}c_i \le B_t, \forall i,t : m_{i,t}\text{integer} \qquad (1)$$

$m_{i,t}$，$\hat{\mu}_{i,n_i}$，及び $n_i$ はそれぞれ式 (1) を最大化するタイムステップ t におけるアーム $i$ のプレイ回数，アーム $i$ をプレイして得られた報酬の平均から求められた期待報酬，及びタイムステップ $t$ までにアーム $i$ をプレイした回数を表す．

エージェントの目標は残りの予算 $B_t$ に対応した式 (1) を満たすアーム $i$ のプレイ回数 $\{m_{i,t}\}_{i\in K}$ を見つけることである．本問題はナップザック問題を解くこととみなせる．ナップザック問題は NP-hard であるため，貪欲法を用いてアームの組み合わせを求めている．アーム $i$ の期待報酬密度に基づく評価値は式 $\frac{\hat{\mu}_{i,n_{i,t}}}{c_i} + \frac{\sqrt{\frac{2\ln t}{n_{i,t}}}}{c_i}$ より求められる．ここで KUBE の式 (1) を最大化するアームの組み合わせの解を $M^*(B_t) = \{m^*_{i,t}\}$ とする．$\{m^*_{i,t}\}$ を用いて KUBE はプレイするアームの確率を決定する．プレイされるアームの確率は式 $P(i(t)=i) = \frac{m^*_{i,t}}{\sum_{k=1}^{K} m^*_{k,t}}$ に従う．

## 3　動的報酬予算制限バンディットアルゴリズム

### 3.1　D-KUBE

Decreasing Knapsack based Upper confidence Bound exploration and Exploitation（D-KUBE）は，KUBE に D-UCB[3] 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-UCB は，動的報酬に適応したバンディットアルゴリズムの一つである．D-KUBE は，D-UCB で用いる減衰率 $\gamma$ を使用する．

D-KUBE は探索時にすべてのアームを一度プレイする．活用時にプレイするアームは式 (2) を最大化するアームである．

$$\max \sum_{i=1}^{K} m_{i,t}\left(\bar{\mu}_t(\gamma,i) + b_t(\gamma,i)\right)$$

$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t}c_i \le B_t, \forall i,t : m_{i,t}\text{integer} \qquad (2)$$

$m_{i,t}$，$\bar{\mu}_t(\gamma,i)$，及び $b_t(\gamma,i)$ はそれぞれ，式 (2) を最大化するタイムステップ $t$ 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，及びタイムステップ $t$ でのアーム $i$ の減衰探索手当を表す．D-KUBE の即時的な期待報酬 $\bar{\mu}_t(\gamma,i)$ は，式 $\bar{\mu}_t(\gamma,i) = \frac{1}{n_t(\gamma,i)}\sum_{s=1}^{t}\gamma^{t-s}r_s(i)\mathbf{I}_{\{i(s)=i\}}$ より求める．$n_t(\gamma,i)$ は，

Budget-Limited multi armed bandit problem with dynamic rewards and proposing algorithms
†Makoto NIIMI　‡Takayuki ITO
†Computer Science, Nagoya Institute of Technology
‡School of Techno-Business Administration, Nagoya Institute of Technology

$n_t(\gamma, i) = \sum_{s=1}^{t} \gamma^{t-s} \mathbf{I}_{\{i(s)=i\}}$ である．$n_t(\gamma, i)$，$r_s(i)$，及び $\mathbf{I}_{\{i(s)=i\}}$ 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，及び $i(s) = i$ となる時 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 $b_t(\gamma, i) = 2\sqrt{\frac{\xi \log n_t(\gamma)}{n_t(\gamma, i)}}$ より求める．$n_t(\gamma) = \sum_{i=1}^{K} n_t(\gamma, i)$．$\xi$ は $0.5 < \xi \leq 1$ となる定数を設定する．アーム $i$ の期待報酬密度に基づく評価値は，式 $\frac{\bar{\mu}_t(\gamma, i)}{c_i} + \frac{b_t(\gamma, i)}{c_i}$ より求める．プレイされるアームの選択確率は KUBE と同様の式で求める．

### 3.2 SW-KUBE

Sliding-Window Knapsack based Upper confidence Bound exploration and Exploitation（SW-KUBE）は，KUBE に SW-UCB[3] 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-UCB は，動的報酬に適応したバンディットアルゴリズムの一つである．SW-KUBE は SW-UCB で用いる参照数 $\tau$ を使用する．

SW-KUBE はまず探索時にすべてのアームを一度プレイする．活用時にプレイされるアームは式 (3) を最大化するアームである．

$$\max \sum_{i=1}^{K} m_{i,t} \left(\bar{\mu}_t(\tau, i) + b_t(\tau, i)\right)$$
$$\text{s.t.} \quad \sum_{i=1}^{K} m_{i,t} c_i \leq B_t, \forall i, t : m_{i,t} \text{integer} \qquad (3)$$

$m_{i,t}$ は式 (3) 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 $\bar{\mu}_t(\tau, i) = \frac{1}{n_t(\tau, i)} \sum_{s=t-\tau+1}^{t} r_s(i) \mathbf{I}_{\{i(s)=i\}}$ より求める．$n_t(\tau, i)$ は式 $n_t(\tau, i) = \sum_{s=t-\tau+1}^{t} \mathbf{I}_{\{i(s)=i\}}$ で表される．減衰探索手当 $b_t(\tau, i)$ は，式 $b_t(\tau, i) = \sqrt{\frac{\xi \log(t \wedge \tau)}{n_t(\tau, i)}}$ より求める．$t \wedge \tau$ は，$t$ 及び $\tau$ の最小値により表される．$\xi$ は $0.5 < \xi \leq 1$ となる定数を設定する．アーム $i$ の期待報酬密度に基づく評価値は，$\frac{\bar{\mu}_t(\tau, i)}{c_i} + \frac{b_t(\tau, i)}{c_i}$ より求める．プレイされるアームの選択確率は KUBE と同様の式で求める．

## 4 評価実験

### 4.1 実験設定

提案手法である D-KUBE 及び 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 らの実験設定に従う [2]．アームのコストは，[1，10] の範囲からそれぞれのアームに対してランダムに設定する．Tran らの実験との違いは報酬確率分布が動的な点である．本評価実験ではそれぞれのアームごとに [100，200] からランダムに変動期間を設定する．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，アームの報酬確率分布の平均値をランダムに設定し直す．

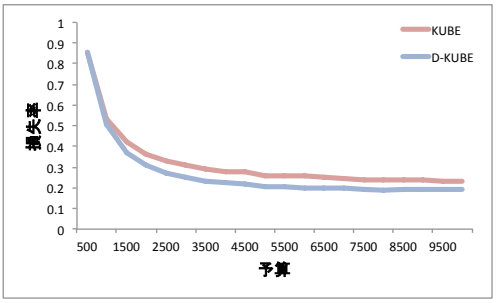


図 1: KUBE と D-KUBE の損失率の比較

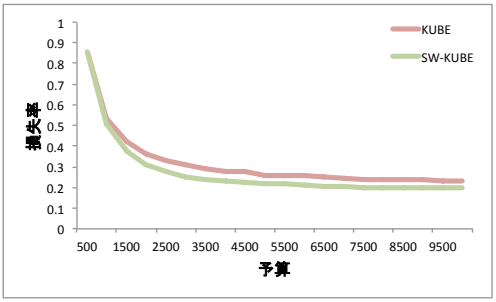


図 2: KUBE と SW-KUBE の損失率の比較

### 4.2 実験結果

KUBE，D-KUBE，及び SW-KUBE を比較した結果について述べる．図 1 及び図 2 はそれぞれ，KUBE と D-KUBE，及び KUBE と SW-KUBE の損失率を比較した図になる．損失率は式 $1 - \frac{total\_reward}{total\_reward^*}$ より求める．$total\_reward$ 及び $total\_reward^*$ はそれぞれ，アームを選択した時のアームの平均報酬の合計，及び得られる平均報酬が最も大きいアームを選択した時の平均報酬の合計を表す．図 1 及び図 2 から，既存手法である KUBE よりも提案手法である D-KUBE 及び SW-KUBE のほうが損失率が小さい．

## 5 まとめと今後の課題

本研究では BL-MAB 問題の報酬確率分布を拡張し動的に変更した．既存のアルゴリズム KUBE を拡張した D-KUBE 及び SW-KUBE は KUBE と比較して損失率が小さくなることを実験により確認した．今後の課題として損失率をさらに小さくすることが挙げられる．また，現実的な設定の実験を行うことも課題である．

## 参考文献

[1] 本橋永至ら:”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 57.10, pp. 574-583, 2012.

[2] Tran-Thanh, Long, et al:”Knapsack Based Optimal Policies for Budget-Limited Multi-Armed Bandits.” AAAI-12, pp. 1134-1140, 2012.

[3] Aurélien Garivier and Eric Moulines:”On upper-confidence bound policies for switching bandit problems.” ALT’11, pp. 174-188, 2011.