Vol.2015-MPS-102 No.14
2015/3/4

確率的バンディットアルゴリズム Successive Elimination に於ける最適アーム探索に必要なサンプル数削減

Reduction in Sample-Complexity of Successive Elimination Algorithm for Stochastic Multi-armed Bandit

福勢 晋 *1　　若原 恭 *1

Shin Fukuse　　Yasushi Wakahara

Abstract

Recently best-arm identification in multi-armed bandit problem(MAB) has become intensively investigated. At each time, a player pulls an arm and gets a reward. The arm that has the largest mean reward is called optimal. The goal of this problem is to identify the optimal arm with the smallest number of pulls. This problem has various applications: clinical trials, psychological experiments, and so on. In previous works, it was revealed that sequential hypothesis testing is closely related with this problem and some best-arm identification algorithms that make use of sequential tests were proposed. However, sequential tests used by the algorithms are far from optimal. We propose two sequential tests of the mean of subgaussian random variables and show that one of the tests is optimal in the aspect of LIL(Law of the Iterated Logarithm). We propose two improved successive elimination algorithms using these tests and demonstrate that these algorithms have smaller sample-complexity than conventional best-arm identification algorithms by computer simulation.

1　序論

マルチアームドバンディット問題 (Multi-Armed Bandit Problem) は、レバーをプルすると確率的に報酬が得られるスロットマシン (アーム) が複数台ある状況で、どのようにスロットマシンを選んでいけば累積報酬を最大化できるかという問題である。広告表示最適化や臨床試験に応用できることに加えて、探索 (Exploration) と活用 (Exploitation) のトレードオフを内在している問題の中でも比較的単純であるため、理論的な研究の価値も高い。今回はアームから得られる報酬として、アームごとに関連付けられた確率分布から i.i,d(independent and identically distributed) にサンプルされることを仮定した確率的マルチアームドバンディット (stochastic multi-armed bandit) を扱う。従来、マルチアームドバンディットではリグレットと呼ばれる、最適アームを常にプルした場合に得られる累積報酬と実際に得られた累積報酬の差の期待値に注目して、これの最小化 (Regret Minimization) を目標に様々なアルゴリズムが提案されてきたが、近年、報酬期待値が最も高いアームをいかに少ないサンプル数 (総プル回数) で見つけるかという、最適アーム探索 (Best-arm Identification) 問題が研究されるようになった。さらに、最適アーム探索は fixed confidence と fixed budget の 2 種類の問題設定に大別することが出来るが、今回は、エラー率 δ が与えられた上で、$1-\delta$ 以上の確率で最適アームを見つけるために必要なサンプル数の最小化を目標にする fixed confidence を扱う。

fixed confidence に対応した最適アーム探索アルゴリズムは、successive elimination[1], Exp-Gap[2], lil'UCB[3] がある。これら 3 つの手法は元々最適アーム探索を目的として作られた手法だが、[3] によって提案された LS(LIL Stopping) というものを使うと、任意のアルゴリズムを最適アーム探索に対応させることが出来る。[3] によると、LUCB[4] と LS を組み合わせたときに実験的に小さい sample-complexity を達成できることが分かっている。sample-complexity とは、最適アームを見つけるために必要なサンプル数のことである。LS は、subgaussian 確率変数の平均に関する sequential test(逐次検定) を元にして作られていて、sequential test の sample-complexity がアルゴリズムの性能に大きく影響する。調べた結果、[3] で使われている sequential test は、第一種エラー率に対してかなり保守的に行われていることがわかった。第一種エラー率に対してよりタイトに sequential test を行う研究は [5] でされているが、LIL(Law of the Iterated Logairthm) から考えて最適では無いことが分かっている。

本研究では、従来よりもタイトに sequential test を行う方法を 2 つ提案する。そのうちの 1 つは LIL から考えて最適な sequential test であることを示す。2 つの sequential test を用いて successive elimination の改良を行う。改良したアルゴリズムの sample-complexity はオーダー的に最適では無いが、実際の sample-complexity

[1]東京大学大学院工学系研究科 Graduate School of Engineering, The University of Tokyo

が従来よりも小さくなるということをシミュレーションによって示す。

2 最適アーム探索

本研究のターゲットとしている fixed confidence 設定での最適アーム探索問題を説明する。そのために、まず数式による定式化を行う。N 個のアームを期待値が高い順に整数で表して $1, 2, \cdots, N$ とする。それぞれのアームに関連付けられている確率分布 (報酬分布) を $\lambda_1, \lambda_2, \cdots, \lambda_N$ として、その期待値を $\mu_1 > \mu_2 \geq \cdots \geq \mu_N$ とする。ただし、最適アーム ($i = 1$) が一意に決まるということを仮定している。つまり $\mu_1 > \mu_2$ としていることに注意する。時刻 t でプレイヤーがプルするアームを $I(t)$ として、時刻 $t-1$ までにアーム i をプルした回数を $T_i(t) = \sum_{s=1}^{t-1} \mathbb{I}\{I(s) = i\}$ とする。ただし、$\mathbb{I}\{e\}$ は事象 e が真のとき 1 で偽のとき 0 である関数とする。アーム i の $T_i(t)$ 回目のプルで得られる報酬を $X_{i,T_i(t)}$ とする。報酬は i.i.d(independent and identically distributed) を仮定してかつアーム間の独立性も仮定する。アーム i から時刻 $t-1$ までに得られた報酬の平均値を $\hat{\mu}_i(t) = \sum_{s=1}^{T_i(t)} X_{i,s}$ とする。アーム i の最適アームとの期待値の差を $\Delta_i = \mu_1 - \mu_i$ とする。本論文では各アームの報酬分布は $(1/2)-$subgaussian であると仮定する。目的は、最適アームを少ない総プル回数で探し出すことである。sample-complexity は総プル回数の期待値で測る考え方もあるが、現在最適アーム探索アルゴリズムの評価や分析として主流なのは、最適アームを見つけて停止した場合の総プル回数を sample-complexity とするものである。これは、アルゴリズムが停止しない可能性を許していることに注意する。従って、fixed confidence 設定での最適アーム探索問題は式 (1) で表現できる。ただし、τ はアルゴリズムが最適アームを見つけて停止する停止時刻、δ は第一種エラー率、$b(\tau)$ はアルゴリズムが停止したときに出力する最適アーム候補とする。

$$\begin{aligned} &\text{minimize} && E[\tau | \tau \text{ is finite}] \\ &\text{subject to} && P[b(\tau) = 1, \tau \text{ is finite}] \geq 1 - \delta \end{aligned} \tag{1}$$

[6] によって式 (2) のような sample-complexity の下限が示されている。$c > 0$ は正の定数である。[7] で提案されている式 (3) で表現されるような H_1 を用いると、sample-complexity の下限は $\Omega(H_1 \log(1/\delta))$ と表現することが出来る。

$$E[\tau] \geq c \left(\sum_{i=2}^{N} \frac{1}{\Delta_i^2} \right) \log \frac{1}{8\delta} \tag{2}$$

$$H_1 = \sum_{i=2}^{N} \frac{1}{\Delta_i^2} \tag{3}$$

3 Sequential test の改良

sequential test に関して簡単な説明を行う。t-検定や二項検定などの基本的な統計学的仮説検定では、まず、サンプル数 T を適当に決めて T 個のサンプルを集める。そして、例えばサンプル元の分布の平均が 0 に等しいという帰無仮説を設定して、T 個のサンプルがその帰無仮説に従うかどうかを検定する。ここで、T 個のサンプルを集めている途中で検定を行うことが出来ないということに注意しないといけない。もし、T 個のサンプルを集めている途中で何度も検定を行うと、全体としての第一種エラー率が最初に与えた上限 δ を超えてしまうからである。一方で、sequential test というのは、T 個のサンプルを集めている途中でも検定を行う方式である。T は有限の値に設定することもあれば、本稿のように ∞ を想定する場合もある。この利点は、サンプル数が N に到達する前に帰無仮説が間違っているということが分かれば、検定に必要なサンプル数を低く抑えることが出来るところにある。普通の検定では、第一種エラー率が同じ 2 つの検定の良し悪しを比べる場合、第二種エラー率が低い方が検出力が高くて良い検定だとみなされるが、∞ 個のサンプルを想定する場合、帰無仮説が間違っていると言えない場合、検定を終了することが無いので第二種エラーが発生するということは無い。その代わりに、sequential test では検定が終了するまでのサンプル数が少ない方を良い検定だとみなす。実際には、sequential test は毎時刻である範囲を計算して、平均がそこからはみ出たら検定を終了する。この毎時刻計算される範囲を本稿では信頼区間列と呼ぶことにする。信頼区間列が狭いほうが検定が終了するのが早いので良い検定だと言える。

3.1 LIL(Law of the Iterated logarithm)

Sequential Test と関係の深い Law of the Iterated Logarithm という法則を紹介する。まず、ランダムウォークを考える。このランダムウォークが 0 より大きい確率で全ての t で収まるような領域はどんな領域かを示すのが、law of the iterated logarithm である。具体的には、平均が 0 で分散が 1 の i.i.d(independent and identically distributed) な確率変数列 $X_1, X_2, \cdots, X_t$ を考える。その和を $S_t = X_1 + X_2 + \cdots + X_t$ とする。このとき以下のような式が成り立つ。ただし、a.s. は almost surely である。

$$\limsup_{t \to +\infty} \frac{S_t}{\sqrt{t \log\log(t)}} = \sqrt{2} \quad a.s. \tag{4}$$

この式によるとランダムウォーク S_t が全ての t で収まるような領域は、$|S_t| < O(\sqrt{t \log\log(t)})$ であることが

示唆される。

3.2 最適な 1 変数 sequential test の提案

LIL の結果を考えると、sequential test の信頼区間列は $t \to +\infty$ で $\sqrt{2\log\log(t)/t}$ に近づけば最適と言える。sequential test の信頼区間列を作る試みは過去に行われているが、[3] では任意の $\epsilon > 0$ に対して $\sqrt{(2+\epsilon)t\log\log(t)}$ であり、[5] では $\sqrt{3t\log\log(t)}$ というように最適では無い。そこで、最適な信頼区間列を計算する方法を提案する。確率空間 $(\Omega, \mathcal{F}, P)$ と、離散時間フィルトレーション $\{\mathcal{F}_t\}_{t=0,1,\cdots}$ を用意する。$\{\Delta X_t\}_{t=1,2,\cdots}$ を各 ΔX_t が 1-subgaussian に従う平均 0 の確率変数とする。このとき $X_t = \sum_{k=1}^{t} \Delta X_k$ を考えると martingale になっている。ただし、$X_0 = 0$ とする。$\{\Delta X_t\}_{t=1,2,\cdots}$ が毎時刻得られる報酬に、$X_t = \sum_{k=1}^{t} \Delta X_k$ がその合計に対応する。まず、以下のような関数 $h(x)$ を定義する。

定義 1. 関数 $h(x)$ を以下の式で定義する。ただし、$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x exp(-t^2)$ を誤差関数とする。

$$h(x) = \frac{\sqrt{x}}{\sqrt{\pi}\,\mathrm{erf}(\sqrt{x})}\exp(-x) \tag{5}$$

次の定理 1 が subgaussian 確率変数列の平均に対する sequential test を行うための基本的な定理である。帰無仮説は全ての ΔX_t が 1-subgaussian でありかつ平均 0 であることとする。$\{\delta_t\}_{t=1,2,\cdots}$ というものを用意して、$\delta_t\ u_t\ p(u_t) \geq h\left(t\ u_t^2/2\right)$ という条件が満たされた時点で帰無仮説が棄却されたとして停止する。ただし、各 δ_t は時刻 $t-1$ までの情報によって決まるとする。このとき第一種エラー率、つまり停止した場合に帰無仮説が正しい確率は $\sup_{0<t}\{\delta_t\}$ 以下になる。

定理 1. $u_t = \frac{1}{t}X_t$ として、$p(x) > 0$ を $0 < x$ の範囲で定義された $\int_0^\infty p(x)dx = 1$ である凸関数とする。また、$\{\delta\}_{t=1,2,\cdots}$ を各 $\delta_t > 0$ が $\mathcal{F}_{t-1}$ 可測である確率変数列とする。このとき以下の式が成り立つ。

$$\mathbb{P}\left[\exists t > 0,\quad \delta_t\ u_t\ p(u_t) \geq h\left(\frac{1}{2}t\ u_t^2\right)\right] \leq \sup_{0<t}\{\delta_t\} \tag{6}$$

これを使って実際に sequential test を行うためには $p(x)$ を具体的に与えないといけない。$p(x)$ に対してどういう関数を設定するかだが、本稿では定義 2 のように $q(x)$ を用意してその積分が 1 になるように正規化したものを $p(x)$ として設定する。この式は、$p(x)$ が凸になるということと、後述するように信頼区間列が最適になるということの両方を満たすように、試行錯誤の上で選んだ。ただし、$p(x)$ が凸になることは数値計算的にしか確認していないので、この $p(x)$ を使った sequential test が正しいという数学的な保証は出来ていない。$p(x)$ を計算するためには $Q := \int_0^\infty q(x)dx$ を求める必要があるが、これは数値積分で求めた結果、$Q \approx 2.519$ であった。

定義 2. $p(x), q(x), Q$ を以下のように定義する。$p(x)$ は $q(x)$ を積分が 1 になるように正規化したものである。

$$\begin{aligned} p(x) &= \frac{q(x)}{Q} \\ q(x) &= \frac{1}{x(2.085x + \log(1+1/x)\log^2(1+\log(1+1/x)))} \\ Q &= \int_0^\infty q(x)dx \end{aligned} \tag{7}$$

次に信頼区間列が LIL から考えて最適であることを確認する。そのためにまずは、定理 1 に式 (7) を代入して計算される信頼区間列を、陽に計算出来る関数 $u(t)$ で評価した上で、それを LIL から考えられる最適な $\sqrt{2\log\log(t)/t}$ と比較する。それが次の定理 2 である。この定理によって、定理 1 に定義 2 の $p(x)$ を与えて行われる sequential test が LIL から考えて最適であることが分かる。

定理 2. 任意の $\delta \in (0, 1]$ に対して、以下のような関数 $u(t)$ を考える。ただし、$W_{-1}(x)$ はランベルトの W 関数である。

$$u(t) = \sqrt{\frac{2}{t}}\sqrt{-\frac{1}{2}W_{-1}\left(-\frac{8\delta^2\pi}{25Q^2\left(\log(1+\sqrt{\frac{t}{2}})\log^2(1+\log(1+\sqrt{\frac{t}{2}}))\right)^2}\right)} \tag{8}$$

このとき $T = \frac{25a^2Q^2}{2\delta^2e^2\pi}$ として、全ての整数 $t > T$ で以下の式が成り立つ。ただし、$p(x)$ は定義 2 のものを使う。

$$\delta u(t)p(u(t)) \geq h\left(\frac{1}{2}tu^2(t)\right) \tag{9}$$

また、$t \to \infty$ で次のような性質を持つ。

$$\limsup_{t\to\infty}\frac{u(t)}{\sqrt{t\log\log(t)}} = \sqrt{2} \tag{10}$$

従来の sequential test と信頼区間列のプロットを図 1 に示す。比較対象は、[5] の Theorem 2 で提案されているものと、lil'UCB[3] の Lemma 3 で提案されているものとする。公平な比較のため全て両側信頼区間に揃えた。そのため、提案手法では δ の代わりに $\delta/2$ を用いている。Theorem 2 は Initial Time というものがあり、$t \leq 173\log\left(\frac{4}{\delta}\right)$ では検定を行わない。lil'UCB の信頼区間列はパラメータ ϵ を持つが、[3] の中では $\epsilon = 0.01$ が推奨されているのでそれを用いる。

3.3 2 変数 sequential test の提案

最適アーム探索では、最終的に 2 つのアームの平均の差が 0 より大きいということを言いたい。これは、それ

ぞれのアームに対して Sequential Test から信頼区間列を計算して、その区間に被りが無いということで確認することが出来るが、実は 2 つのアームのプル回数が同じくらいの場合には、2 つのアームの平均の差に対する特別の sequential test を用意することによって、最大で必要プル回数を半分程度にすることが出来る。これを、本稿では 2 変数 sequential test と呼ぶ。次の定理 3 が基本となる定理である。

定理 3. 平均が 0 でスケールファクター 1 の 2 つの sub-gaussian 確率変数列 $X_1, X_2, \cdots, X_t$ と $Y_1, Y_2, \cdots, Y_t$ を考える。$u_t = \frac{1}{n(t)} \sum_{t_2=1}^{n}(t) X_{t_2}, v_t = \frac{1}{m(t)} \sum_{t_2=1}^{m}(t) Y_{t_2}$ として、$p(x) > 0$ を $0 \leq x$ の範囲で凸で $\int_0^\infty p(x)dx = 1$ である関数とする。このとき任意の $\delta \geq 0$ に対して以下の式が成り立つ。

$$\mathbb{P}\left[\exists t{>}0,\ \delta\ u_t\ p(u_t)\ v_t\ p(v_t) {\geq} h\left(\tfrac{1}{2}n(t)\ u_t^2\right) h\left(\tfrac{1}{2}m(t)\ v_t^2\right)\right] {\leq} \delta \quad (11)$$

また、以下のように $\delta = \delta_1 \cdot \delta_2$ と分解すれば、1 変数の場合の Sequential Test を組み合わせて 2 変数の Sequential Test が行えることがわかる。

$$\begin{cases} \delta_1\ u_t\ p(u_t) & \geq h\left(\frac{1}{2}n(t)\ u_t^2\right) \\ \delta_2\ v_t\ p(v_t) & \geq h\left(\frac{1}{2}m(t)\ v_t^2\right) \end{cases}$$
$$\Rightarrow \quad \delta\ u_t\ p(u_t)\ v_t\ p(v_t) {\geq} h\left(\tfrac{1}{2}n(t)\ u_t^2\right) h\left(\tfrac{1}{2}m(t)\ v_t^2\right) \quad (12)$$

これを用いて実際に 2 つのアーム間の平均の差を検定する方法を説明する。今、ある時刻 t で 2 つのアームの平均の差が Δ だったとする。u_t, v_t はそれぞれ 2 つのアームの真の平均からのずれを意味する。帰無仮説 H_0 は 2 つのアームの平均が等しいこととする。H_0 が正しいとすると、$u_t + v_t = \Delta$ が成り立つ。u_t, v_t は未知なので直接上の定理の条件を満たしているかは確認できないが、ありうる全ての $u_t + v_t = \Delta$ で上の定理の条件を満たすということが確認できれば、帰無仮説は正しくないということが結論付けられる。問題は、ありうる全ての (u_t, v_t) で定理の条件を満たしているかを確認することだが、$x < 0$ で $p(x) = 0$ なので、$u_t < 0$ または $v_t < 0$ で明らかに定理の条件を満たさない。つまり、この検定単独では検定が行えない。そこで、図 2 のように 1 変数の Sequential Test を 2 つ組み合わせて、、全部で 3 つの検定を組み合わせて検定を行うことを考える。第一種エラー率は均等に $\delta/3$ で割り振る。2 変数 Sequential Test では埋めきれない隙間を 1 変数 Sequential Test が補っていることが図からわかる。

3 つの検定を組み合わせているので実装が複雑になるように思えるが、実際は前述の定理で確認したとおり 2 変数の Sequential Test は、1 変数の Sequential Test に分解して行っても良いので簡単に実装できる。最終的に、実際に 2 つのアーム間の平均の差が 0 かどうかをエラー率 δ 以下で検定する方法は、以下の式で定義される ω を計算し、$\omega < \frac{\delta}{3}$ が成り立ったときに H_0 を棄却するというものである。

$$\omega{=}\max_{u+v=\Delta}\left\{\frac{h\left(\frac{1}{2}n\ u^2\right)h\left(\frac{1}{2}m\ v^2\right)}{u\ p(u)\ v\ p(v)} | \frac{h\left(\frac{1}{2}n\ u^2\right)}{u\ p(u)} {\leq} 1, \frac{h\left(\frac{1}{2}m\ v^2\right)}{v\ p(v)} {\leq} 1\right\} \quad (13)$$

1 変数 sequential test の必要サンプル数に対しての 2 変数 sequential test の必要サンプル数の比が、2 つのアームのプル回数の比 $n/(n+m)$ によってどう変わるかを調べた。必要サンプル数は、平均の差 $\Delta = 0.1$ をエラー率 $\delta = 0.1, 0.01$ で有意と言うために必要なサンプル数として計算した。結果を図 3 に示す。2 つのアームのプル回数が同じくらいのときに 2 変数 sequential test の方がサンプル数が少ないことが分かる。

4 最適アーム探索アルゴリズムの改良

4.1 LS(LIL Stopping) の改良

提案した 2 つの sequential test を用いて、[3] で提案されている LS の改良を行う。最大アーム $m(t)$ とその他全てのアーム i の間で、信頼区間に被りが無ければ $m(t) = 1$ と結論付けるのが、オリジナルの LS の原理である。このとき、各アームの信頼区間計算に用いるエラー率は δ/N とする。こうすることで全体のエラー率は δ 以下になる。しかし、オリジナルの LS では信頼区間計算に保守的な sequential test が用いられているので、提案した sequential test で置き換えれば sample complexity を減らすことが期待出来る。また、2 変数 sequential test では被りが無いということの代わりに、最大アーム $m(t)$ とその他全てのアームとの間で H_0 が棄却されたということを用いる。ただし、各アーム間の sequential test に用いるエラー率は $\delta/(N-1)$ とする。このようにオリジナルの LS に対して、1 変数 sequential test, 2 変数 sequential test で置き換えたものを本稿ではそれぞれ LS1, LS2 と呼ぶ。

4.2 Successive elimination の改良

successive elimination の原理はまずアーム集合 $A = \{1, 2, \cdots, N\}$ を用意する。LS と同様に最大アーム $m(t)$ とその他全てのアーム i の間で、信頼区間に被りが無いかを調べる。被りが無いアームがあったらそれをアーム集合から除いていって、最終的にアーム集合の要素数が 1 になったらそれを最適アームとして出力するアルゴリズムである。従って、LS と同様に sequential test で置き換えるという改良を行うことが出来る。オリジナルの

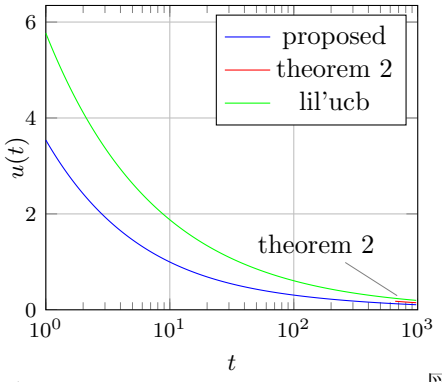

図 1: Sequential test bounds. ($\delta = 0.1$)

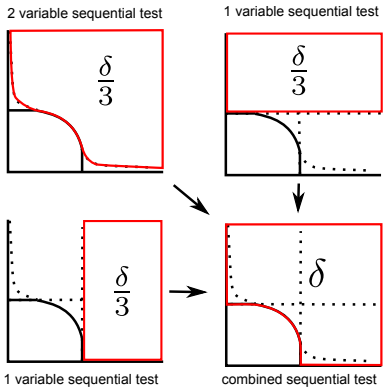

図 2: Sequential test combination. Regions surrounded by red lines represent events occurring with probabilities lower than δ or $\delta/3$. Axes represents u_t, v_t.

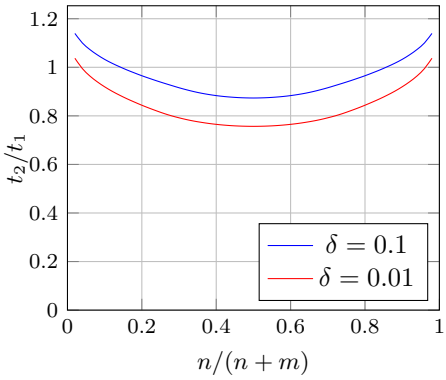

図 3: Ratio of required samples of 1 variable sequential test(t_1) and that of 2 variable sequential test(t_2) vs. two arms' pull count ratio $n/(n+m)$.

successive elimination に対してこの改良を施したものを、本稿では LS1 elimination, LS2 elimination と呼ぶ。

Algorithm 1 successive elimination

input: $N, \delta > 0$
intialize $t \leftarrow N, A \leftarrow \{1, 2, \cdots, N\}$
for $i = 1$ to N **do**
  pull arm i
end for
while $|A| > 1$ **do**
  $m \leftarrow \arg\max_{i \in A}\{\hat{\mu}_i(t)\}$
  $B \leftarrow \{i \in A \mid \text{sequential test}(m \text{ vs } i)\text{stopped}\}$
  $A \leftarrow A \setminus B$
  $I(t) \leftarrow \arg\min_{i \in A}\{T_i(t)\}$
  pull $I(t)$ arm
  $t \leftarrow t + 1$
end while
output: an arm in A (algorithm promise $|A| = 1$)

5 シミュレーションによる評価

5.1 実験条件

アームの期待値分布としては、[8] で提案されている、α で特徴付けされた分布を用いる (図 4)。報酬は分散 0.25 のガウス分布を用いる。アルゴリズムが停止した時刻の期待値を sample-complexity とし、それを式 (3) で表現される問題の難易度 H_1 を用いて正規化してプロットした。期待値は 100 回シミュレーションを行った平均値で求めた。その精度についてだが、期待値の信頼区間幅を相対値にしたもの (標準偏差 / 平均値 / $\sqrt{100}$) は最大で 5.2% であった。最大時刻を $T = 1000 \cdot H_1$ とし、T までに停止しなかった試行の停止時刻は T とみなして計算した。比較するアルゴリズムは、提案手法と従来手法の lil'UCB に加えて、LUCB とオリジナルの LS を組み合わせたもの (LUCB + LS(orig)) と、LUCB と LS2 を組み合わせたもの (LUCB + LS2) である。[3] によると、LUCB + LS(orig) が従来手法の中で最も sample-complexity が小さい。

5.2 アーム数への依存性

アーム数を変えたときにアルゴリズムの sample-complexity の変化を見る。具体的な実験条件と結果を表 1, 図 5 に示す。全体的に sample-complexity のアーム数への依存性が低いことと、提案手法 (LS2 elimination) は従来手法 (LUCB + LS(orig)) と比べて、3〜5 倍 sample-complexity が低いことが分かる。

表 1: Experiment settings 1

N	δ	distribution(α)
2, 3, 6, 12, 25, 50, 100	0.1	0.6

5.3 δ への依存性

エラー率 δ を変えたときの sample-complexity の変化を見る。比較的 sample-complexity の低いもののみに注目した。具体的な実験条件を表 2 に、結果を図 6 に示す。$\alpha = 0.6$ のみをプロットしたが、$\alpha = 0.3$ でも同じ直線傾向がある。表 3 は線形近似した結果である。線形近似の傾きは、式 (2) の係数 c に対応するが、LS2 elimination のデータから考えると $c \leq 2$ という予想が立つ。

表 2: Experiment settings 2

N	δ	distribution(α)
100	$10^{-1}, 10^{-2}, \cdots, 10^{-10}$	0.3, 0.6

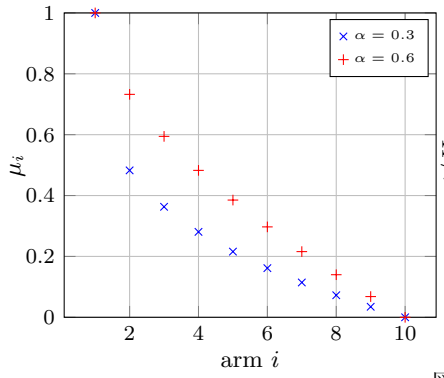

図 4: Distribution of the expectation of arm reward(μ_i). ($N = 10, \alpha = 0.3, 0.6$)

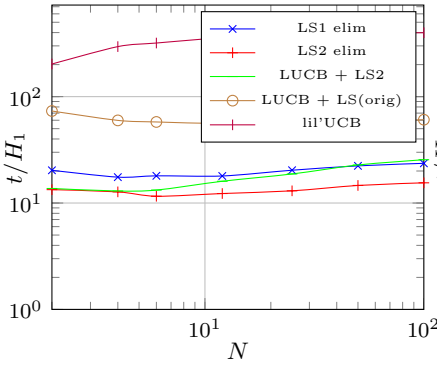

図 5: Sample-complexity vs. N. ($\delta = 0.1, \alpha = 0.3$)

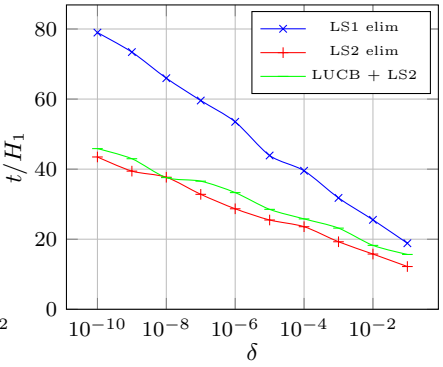

図 6: Sample-complexity vs. δ. ($N = 10, \alpha = 0.3$)

表 3: Linear fitting ($t/H_1 \approx a + b\log(1/\delta)$). ($N = 10$)

algorithm	fitting($\alpha = 0.3$)	fitting($\alpha = 0.6$)
LS1 elim	9.56 + 2.60 log(1/δ)	12.6 + 2.72 log(1/δ)
LS2 elim	8.79 + 1.26 log(1/δ)	8.88 + 1.44 log(1/δ)
LUCB + LS2	10.1 + 1.35 log(1/δ)	12.4 + 1.40 log(1/δ)
lil'UCB	38.6 + 278 log(1/δ)	35.4 + 262 log(1/δ)

6 結論

今回は、確率的マルチアームドバンディットに対する最適アーム探索の必要サンプル数を改善するために、sub-gaussian 確率変数の平均に関して従来よりもタイトに sequential test を行う方法を 2 つ提案した。うち 1 つは LIL の観点から考えて最適であることも示した。それら 2 つの sequential test を用いて successive elimination アルゴリズムの改良を行った。改良したアルゴリズムの sample-complexity はオーダー的に最適では無いが、実際には従来よりも小さくなるということをシミュレーションによって示した。これが例えば臨床試験に実際に応用されれば、被験者数、試験にかかる時間、コストが減ることが期待される。具体的には比較対象が 10 個程度で、エラー率 10% のときに、従来と比べて sample-complexity を 3 倍程度減らすことが出来る。

今後の研究の方向性としては、今まで提案された中で最も良いオーダーを持つ Exp-Gap[2] や lil'UCB[3] と同じオーダーを持ち、なおかつ提案手法と同じ程度の実性能を持つアルゴリズムを模索することや、紹介した手法は提案手法も含めて全てアルゴリズムが停止しない可能性がわずかにあるので、必ず停止することを保証したアルゴリズムを模索することが考えられる。

参考文献

[1] Eyal Even-dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *In Fifteenth Annual Conference on Computational Learning Theory (COLT)*, pp. 255–270, 2002.

[2] Zohar Shay Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *ICML (3)*, Vol. 28 of *JMLR Proceedings*, pp. 1238–1246. JMLR.org, 2013.

[3] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil' ucb : An optimal exploration algorithm for multi-armed bandits. In *Volume 35: Proceedings of The 27th Conference on Learning Theory*, pp. 423–439, 2014.

[4] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC subset selection in stochastic multi-armed bandits. In John Langford and Joelle Pineau, editors, *In proceedings of the 29th International Conference on Machine Learning (ICML)*, pp. 655–662, New York, NY, USA, June-July 2012. Omnipress.

[5] Akshay Balsubramani. Sharp uniform martingale concentration bounds. *CoRR*, 2014. arXiv:1405.2639v3 [math.PR].

[6] S Mannor and JN Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *The Journal of Machine Learning Research*, Vol. 5, pp. 623–648, 2004.

[7] J.-Y. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *Proceedings of the 23th annual conference on Computational Learning Theory*, Haífa (Israél), Jun 2010.

[8] Jamieson Kevin, Malloy Matthew, Nowak Robert, and Bubeck Sébastien. On finding the largest mean among many. *CoRR*, 2013. arXiv:1306.3917v1 [stat.ML].