Asymptotically optimal multi-armed bandit policies under a strict cost constraint

Hulst, S.

S. Hulst

Asymptotically optimal multi-armed bandit policies under a strict cost constraint

Bachelor thesis

17 Juli 2023

Thesis supervisor: dr. O. Kanavetas

Leiden University
Mathematical Institute

Contents

Introduction

The Multi-Armed Bandit problem serves as a model to study decision-making processes. In this model an agent must decide between multiple options, each having an unknown probability distribution that determines the reward. The agent's objective is to maximize the cumulative reward over a series of time periods where the agent chooses one option per period. This framework can be applied to various domains, including clinical trials, online advertising and resource allocation. The challenge of these decision-making processes is the balance between exploiting high rewards and exploring other options to discover potentially more lucrative decisions.

Simple policies for these types of problems are based on a greedy approach where a parameter, usually denoted by ε, decides the proportion of trials we dedicate to exploration and exploitation. These types of policies result in an approximate solution meaning that they will not approach the true optimal value for all instances of this problem. This is a common issue for greedy algorithms. The value of ε is set before applying the policy and it is not known beforehand whether this will be enough to find the true optimal solution.

In 1952 Robbins [1] considered an instance of the multi-armed bandit problem with 2 arms. The arms had expected values of α, β respectively. If it is known whether α or β is greater then an optimal policy would simply sample from the arm with the greater expected value. The performance of a policy on this problem was measured by the *regret*, the difference between the value achieved by the optimal policy, $\max(\alpha, \beta)$, and the expected value of the policy we follow. Robbins proposed a policy that in the long run achieved an average reward of $\max(\alpha, \beta)$. He did this by using a special type of sequence we will discuss later on.

This framework can be extended by adding a cost constraint. Considering the domain of online advertising, different types of advertisements cost different amounts of money and different websites charge different amounts. Depending on the cost, the website with the highest reward might not be the optimal choice. With the cost constraint, each arm now has an asssociated sampling cost and the agent is given a certain amount of budget he can use to sample. At the beginning of every period receives the same amount of budget again, adding it to however much budget he has left over after last period. The last consideration is how the budget is treated. Depending on the context, a lax constraint where the policy is allowed to exceed the budget for a certain period of time might be appropriate whilst in another a strict constraint where the policy is never allowed to exceed the budget models the problem best.

In 2012 Burnetas and Kanavetas [2] proposed a policy for the multi-armed bandit problem with a lax cost constraint. The agent has $k \in \mathbb{N}$ arms, each with their own distribution and associated cost. They proved that their policy converges to the optimal average reward in all cases under the constraint that the policy is not allowed to exceed the budget in the long run. In this thesis we will construct a policy for the multi-armed bandit problem with a stricter cost constraint where the agent is never allowed to exceed the budget.

In Chapter 1 the multi-armed bandit with sampling costs and its solutions will be defined and characterized. In Chapter 2 we will discuss and analyse the policy proposed by Burnetas and Kanavetas. In Chapter 3 we will construct the policy for the stricter cost constraint. In Chapter 4 we will prove that this policy is optimal and in Chapter 5 we will analyse its performance and characteristics under various circumstances.

1 Multi-Armed Bandit with sampling costs

1.1 Model description

Consider an instance of the multi-armed bandit problem where our agent has k options with $k \in \mathbb{N}$. These are the k arms of our model, which we will refer to as statistical populations from now on to emphasize their stochastic nature. Each population i is defined by a univariate distribution with density f_i. Each instance of this problem is uniquely determined by $f = (f_1, \ldots, f_k)$. Successive samples from a population i form a sequence $X_i = (X_{i,1}, X_{i,2}, \ldots)$ of independent and identically distributed random variables following f_i. Each population i also has an associated sampling cost c_i per sample. In each period the agent must sample from one population and receives a budget of C_0 every period to do so. This means that if the agent spends less than C_0 on sampling in one period, then the next period we will be able to spend C_0 plus the leftover budget from the previous period.

Without loss of generality, we make the following assumptions about the sampling costs $\{c_j\}_{j=1}^k$:

- $c_1 \leq c_2 \leq \ldots \leq c_k$ and $c_1 \neq c_k$;
- $c_1 \leq C_0 < c_k$.

The first assumption orders the populations according to cost and forces there to be at least two unique sampling costs. The second assumption ensures that we are considering a valid instance of the problem. If $c_1 > C_0$, then we would not be able to sample from any population in the first period and thus the problem is infeasible. On the other hand, if $c_k \geq C_0$, then the cost constraint is redundant because we can sample from any population at every point in time.

The objective of the agent is to maximize the expected average reward per period in the long run whilst f is unknown. This means that the agent has to learn the means, $\mu(f) = (\mu_1(f), \ldots, \mu_k(f))$ where $\mu_i(f) = \mathbb{E}^{f_i}[X_i]$, before he is able to find the optimal solution. Every time that we sample from a population, we add that new observation to our knowledge and adapt our policy. This is why this problem is also referred to as adaptive sampling and the policies that attempt to solve this are called adaptive policies.

1.2 Complete information

A sampling policy $\{x_j\}_{j=1}^k$ can be characterized as a probability distribution on the set of populations $\{1, \ldots, k\}$ which selects a population j with probability x_j. To analyse the performance of an adaptive policy we first need to characterize the optimal solution and the value it achieves. If we take the position of an agent with complete information about f, then we can compute the optimal solution via linear programming.

Let $\underline{\mu} = (\mu_1, \ldots, \mu_k)$ be the means of the populations. We want to maximize the expected reward, in other words

$$\text{maximize } \sum_{j=1}^{k} \mu_j x_j,$$

whilst keeping the expected costs below C_0:

$$\sum_{j=1}^{k} c_j x_j \leq C_0 \text{ and } \sum_{j=1}^{k} x_j = 1 \text{ and } x_i \geq 0, \forall i \in \{1, \ldots, k\}.$$

Combining this in a linear program, LP for short, gives us the average reward achieved in the long run by the optimal solution to:

$$z^*(\underline{\mu}) = z^* = \max \left\{ \sum_{j=1}^{k} \mu_j \cdot x_j \;\middle|\; \begin{array}{ccc} \sum_{j=1}^{k} c_j x_j + y & = & C_0 \\ \sum_{j=1}^{k} x_j & = & 1 \\ x_j \geq 0, & & \forall j \in \{1, \ldots, k\} \end{array} \right\}. \tag{1}$$

For every optimal average reward $z^*(\underline{\mu})$ we have an associated optimal sampling policy $\underline{x}^*$. Note that, given the same sampling costs, this linear program is only dependent on $\underline{\mu}$ and thus $z^*(\underline{\mu})$ is the same for all f with the same $\underline{\mu}$.

The optimal sampling policy takes two different forms, either the solution consists of two populations or the solution consists of one population. We are only interested in the populations which have a nonzero probability of being chosen, these are so-called basic variables in the linear program (Eq. 1). In the first case, we have basic variables x_i, x_j with $c_i \leq C_0 \leq c_j, c_i < c_j$ and

$$x_i = \frac{c_j - C_0}{c_j - c_i}, x_j = \frac{C_0 - c_i}{c_j - c_i} \text{ and } x_m = 0 \; \forall m \neq i, j \tag{2}$$

with

$$z^*(\underline{\mu}) = x_i \mu_i + x_j \mu_j. \tag{3}$$

For the second case, we have x_i, y as basic variables with $c_i \leq C_0$ and

$$x_i = 1, x_m = 0, \forall m \neq i, y = C_0 - c_i \tag{4}$$

with

$$z^*(\underline{\mu}) = \mu_i.$$

All the basic variables together form a basic feasible solution, BFS for short.

Definition 1 (Basic Feasible Solution) *A basic feasible solution is a solution b to a LP such that $b \subset \{x_1, \ldots, x_k\}$ and*

$$x_i \neq 0, \forall x_i \in b.$$

As we have discussed above, the solution consists either of one population with $c_i \leq C_0$ or two populations with $c_i \leq C_0 \leq c_j$. Let

$$d = \max\big\{i : c_i \leq C_0\big\}$$

then for a BFS b let

$$b = \big\{i : x_i > 0\big\}.$$

The set of all BFS can be defined by

$$K = \{b : b = \{i, j\}, i \leq d \leq j \text{ or } b = \{i\}, i \leq d\},$$

and finally we can define the set of optimal solutions, for the set of means $\underline{\mu}$, as follows

$$s(\underline{\mu}) = \{b \in K : b \text{ corresponds to an optimal BFS}\}.$$

1.3 Adaptive policy

Now that we know how to characterize the optimal solutions when the value of $\underline{\mu}$ is known, we will consider adaptive policies that are not dependent on knowledge of $\underline{\mu}$. We will further assume the following about our statistical populations:

Assumption 1 *The distributions of the statistical populations, $f = (f_1, \ldots, f_k)$, are independent and have finite expected values, $\mu_i = \mathbb{E}^{f_i}[X_i] < \infty, i = 1, \ldots, k$.*

This will be important later when constructing adaptive policies. For now, let F be the set of all $f = (f_1, \ldots, f_k)$ that satisfy Assumption 1. Class F will serve as the parameter set for the adaptive policies. Without knowledge of $\underline{\mu}$, these policies will have to depend on past observations of the selections and outcomes.

Let A_t, X_t denote the population selected and the observed outcome at period t, then let

$$h_t = (\alpha_1, x_1, \ldots, \alpha_{t-1}, x_{t-1}),$$

be the history of observed actions and selections up to period t. Now, an adaptive policy is defined as a sequence $\pi = (\pi_1, \pi_2, \ldots)$ of history dependent probability distributions on the set of populations $\{1, \ldots, k\}$. The probability that a population j is selected in period t, given history h_t, is

$$\pi_t(j, h_t) = \mathbb{P}(A_t = j | h_t).$$

To define the desirable properties of an adaptive policy, we need to first define a few other quantities. Given history h_n, let $T_n(\alpha)$ denote the number of times population α has been sampled in the first n periods

$$T_n(\alpha) = \sum_{t=1}^{n} \mathbb{1}\Big\{A_t = \alpha\Big\}.$$

Let S_n^π be the total reward up to period n under policy π:

$$S_n^\pi = \sum_{t=1}^{n} X_t,$$

and C_n^π be the total cost up to period n under policy π:

$$C_n^\pi = \sum_{t=1}^{n} c_{A_t}.$$

The main objective of an adaptive policy is to achieve the optimal average reward in the long run, this property is called consistency.

Definition 2 (Consistency) *A policy π is called consistent if it is feasible and*

$$\lim_{n\to\infty} \frac{S_n^\pi}{n} = z^*(\underline{\mu}), \ \ a.s, \ \forall f \in F.$$

This property ensures that a policy converges with probability 1 to the optimal expected value that could be achieved under complete information. The second property is also mentioned here, feasibility. The definition of feasibility determines how the cost constraint is handled. We will be discussing feasibility defined by two different types of cost constraint: an *asymptotic* cost constraint and a *strict* cost constraint.

2 Asymptotic cost constraint

An asymptotic cost constraint allows the agent to exceed the budget at certain periods as long as the long run expected average sampling cost is below the budget, in other words:

Definition 3 (Asymptotic feasibility) *A policy π is called feasible if*

$$\limsup_{n\to\infty} \frac{\mathbb{E}^{\pi}[C_n^{\pi}]}{n} \leq C_0, \forall f \in F.$$

It is very easy to show that these type of feasible policies exist because we know all the sampling costs. Any randomized policy that satisfies the constraints of the complete information LP (Eq. 1) is feasible for any f. The difficult part is constructing a consistent policy. There are two important parts to a consistent policy, obtaining estimates of the mean rewards of the populations and sampling from optimal populations. These two objectives clash with each other. To obtain estimates for all populations, we need to sample from nonoptimal populations. This sampling from nonoptimal populations lowers the average reward achieved. However, if we do not sample enough from all populations, then there is a chance that we do not obtain the true optimal solution. Burnetas and Kanavetas [2] solved these challenges using two techniques we will discuss now.

2.1 Sparse sequences

We will first tackle obtaining the estimates of the mean rewards of the populations. The problem is that beforehand we do not know how many times we need to sample from each population to get sufficient estimates to compute the optimal solution. What we need, is a system to decide when we sample from certain populations for the purpose of getting better estimates and when we sample from the populations we believe to be part of the optimal solution. The difficult part is that if we sample too little, then we might miss the true optimal solution, but if we sample too much then our average reward would be lower due to sampling a lot from nonoptimal solutions. The solution to achieving balance between these two aspects is using sparse sequences.

Definition 4 (Sparse sequence) *A sequence $\{\tau_n\}_{n\in\mathbb{N}}$ is called sparse if*

$$\lim_{n\to\infty} \frac{\sum_{k=1}^{n} \mathbb{1}\Big\{\tau_m = k, \text{for some } m \in \mathbb{N}\Big\}}{n} = 0$$

or equivalently

$$\lim_{n\to\infty} \frac{\tau_n}{n} = \infty.$$

The first part of Definition 4 is equivalent to requiring that the fraction of integers that the subsequence $\{\tau_k\}_{k=1}^{n}$ has in common with the set $\{1, 2, \ldots, n\}$ tends to 0 as $n \to \infty$. If we have a sequence that satisfies the second part of Definition 4, then it also satisfies the first definition. This follows from the fact that if a sequence $\{\tau_k\}_{k\in\mathbb{N}}$ grows faster than n as $n \to \infty$, then it grows sufficiently fast that the fraction of integers in common with $\{1, 2, \ldots, n\}$ tends to 0. These sequences will be used to balance between sampling too little and too much.

2.2 Certainty Equivalence Linear Program

Now that we have discussed a system for sampling from all populations to obtain good estimates, we need a method to determine what is the optimal solution and sample according to that solution. Here we take inspiration from the previously defined complete information LP (Eq. 1). There we used $\underline{\mu}$, but instead of the true value we use estimators, since the true values are unknown.

Definition 5 (Certainty Equivalence Linear Program, CELP) *For any $n \in \mathbb{N}$, let*

$$\underline{\hat{\mu}}_n = (\hat{\mu}_{j,T_n(j)}, j = 1, \ldots, k)$$

be the vector estimates of $\underline{\mu}$ based on the history up to period n. Let $\hat{z}_n$ be the solution to the Certainty Equivalence Linear Program using vector estimates up to period n:

$$\hat{z}_n = \max \left\{ \sum_{j=1}^{k} \hat{\mu}_{j,T_n(j)} \cdot \hat{x}_j \;\middle|\; \begin{array}{ccc} \sum_{j=1}^{k} c_j \hat{x}_j + y & = & C_0 \\ \sum_{j=1}^{k} \hat{x}_j & = & 1 \\ \hat{x}_j \geq 0, & \forall j \in \{1, \ldots, k\} & \end{array} \right\} \tag{5}$$

Just like before, every $\hat{z}_n$ has an associated sampling policy $\underline{\hat{x}}_n$ and these take the same form as Eq. 2 and Eq. 4. For every period n, we compute the solutions to the CELP which gives us $s(\underline{\hat{\mu}}_n)$. The goal is in the long run $s(\underline{\hat{\mu}}_n)$ will contain the same optimal solution as $s(\underline{\mu})$. Earlier we made Assumption 1 for this reason, since it allows us to define an estimator that can accomplish this.

Definition 6 (Strongly consistent) *We call $\hat{\mu}_n$ a strongly consistent estimator of μ if*

$$\lim_{n \to \infty} \hat{\mu}_n = \mu \ a.s.$$

Under Assumption 1, the well known sample mean

$$\overline{X}_{j,n} = \frac{1}{n} \sum_{t=1}^{n} X_t \cdot \mathbb{1}\Big\{\text{population } j \text{ is selected}\Big\},$$

is a strongly consistent estimator for μ_j, for all $j \in \{1, \ldots, k\}$. Now we have all the elements to construct a consistent policy.

2.3 Sparse Period Policy

A consistent policy has three goals:

1. Estimate the mean reward of all populations;
2. Sample from nonoptimal populations rarely enough to not affect the average reward achieved;
3. Sample from the optimal populations to approach the average reward achieved under complete information.

It has to accomplish this whilst staying feasible according to Definition 3. This is not a trivial task. Let's first consider k nonoverlapping sparse sequences of positive integers,

$$\tau_j = \{\tau_{j,m}, m = 1, 2, \ldots\}, \ \ j \in \{1, \ldots, k\},$$

such that

$$\lim_{m\to\infty}\frac{\tau_{j,m}}{m}=\infty \text{ and } \tau_{j,1}=j \text{ for } j=1,\dots,k.$$

These sequences help us with the first two goals. Whenever a period n coincides with one of the k sequences, we will sample from that population. This forces us to periodically sample from all the populations and will allow the estimates to converge to the true mean in the long run whilst also sampling rarely enough in the long run to not affect the average reward.

The last goal will be accomplished with the CELP (Eq. 5). Every period that does not coincide with any of the k sequences, we will compute the CELP with the current estimates and perform the sampling policy obtained. Combining these two techniques gives us the policy π^{AC} which I will refer to as the "Sparse Period Policy" (SPP): π^{AC} selects populations j with a probability equal to:

$$\pi^{\mathrm{AC}}(j|h_n)=\begin{cases}1 & \text{if } \tau_{j,m}=n \text{ for some } m\geq 1,\\ \hat{x}_{n,j} & \text{otherwise.}\end{cases} \tag{6}$$

Burnetas and Kanavetas [2] proved that this policy is consistent for the asymptotic cost constraint.

3 Strict cost constraint

At this stage, we want to study a more difficult version of this problem. The asymptotic cost constraint allows us to freely sample in the beginning as long as we stay under budget in the long run. This would not be broadly applicable to real world situations. If a budget is given per time period for a project, it is expected that you obey that budget at every point in time. This inspires our next feasibility criterion with a stricter cost constraint:

Definition 7 (Strict feasibility) *A policy π is called feasible if*

$$\forall n\in\mathbb{N}:\frac{\mathbb{E}^{\pi}[C_n^{\pi}]}{n}\leq C_0, \forall f\in F.$$

For a policy to be feasible under this definition, it will have to never exceed the budget. Considering the policy π^{AC} (Eq. 6) through this lens, we can see that it would violate this criterion within the first k steps. From our assumptions, there is at least one population m such that $c_m > C_0$. Therefore when the sparse sequence forces us to sample from this population, we will violate the feasibility criterion. Due to the forced samplings from the sparse sequences, we will exceed the budget in some periods.

The sparse sequences solved the important challenge of sampling enough to obtain sufficient estimates, so we should look at how we can adapt them to this stricter cost constraint. The main problem is that some populations are too expensive to sample in one period. However, if we sample from cheaper populations in a few periods, we expand our budget for the next period and might now be able to sample from the expensive population. So we can sample from all populations to obtain estimates if we consider our actions for multiple periods at a time. Therefore, we will look at policies defined by blocks of periods instead of singular periods.

3.1 Blocks of periods

A block is a sequence of periods. Before the start of the block, the length and populations to be selected are determined based on currently known information. In the previous period-based policy, we used two important techniques: forced sampling by sparse sequences and linear programming periods. Burnetas, et al. [3] adapted both of these techniques to block-based policies.

3.1.1 Sampling block

To use cheaper populations to subsidize the expensive populations, we need to know how much extra budget we gain after sampling from cheap populations and how much we need for the expensive populations. To this end, we define the *difference* for population j as

$$\delta_j = C_0 - c_j.$$

Each δ_j corresponds to the net effect of sampling from population j on the budget, either it represents a net cost if $\delta_j < 0$ or a net saving if $\delta_j > 0$. This gives us an alternative way to determine whether a sequence of sampling does or does not exceed the budget. Let $\{m_j\}_{j=1}^k$ be a sequence such that population j is sampled m_j times, then if

- $\sum_{j=1}^k m_j\delta_j > 0$: this sequence exceeds the budget;
- $\sum_{j=1}^k m_j\delta_j \leq 0$: this sequence does not exceed the budget.

Now, if we find a sequence $\{m_j^*\}_{j=1}^k$ such that $\sum_{j=1}^k m_j^*\delta_j \leq 0$, then we can follow this sequence to obtain estimates for every population. This sequence can be computed by an integer linear program.

$$\{m_j^*\}_{j=1}^k = \operatorname{argmin}\left\{\sum_{j=1}^k m_j : \sum_{j=1}^k m_j\delta_j \leq 0 \ \ \& \ \ m_j \in \mathbb{N}, \forall j \in \{1,\ldots,k\}\right\}.$$

This LP is only dependent on constants, namely the costs $\{c_j\}_{j=1}^k$ and the budget C_0, so it can be computed once and the solution can be continually used afterwards. The sampling block is then defined as:

for $j \in \{1,\ldots,k\}$, sample m_j^* times from population j.

3.1.2 Linear programming block

After obtaining initial estimates for the mean rewards of the populations, we want to compute the CELP (Eq. 5) and sample according to its optimal solution. In the beginning, we characterized a sampling policy with randomization that follows a probability distribution $\{x_j\}_{j=1}^k$ on $\{1,\ldots,k\}$. Another way to implement a sampling policy is to sample periodically from all populations such that the proportion of samples from each population j is equal to the randomization probability x_j. These two characterizations are equivalent if the randomization probabilities $\{x_j\}_{j=1}^k$ are rational.

Before we begin a linear programming block, we compute the optimal solution. Let b be the optimal BFS of the CELP at this period. The LP block is then defined as follows:

- if $b = \{i\}$, we sample from population i once. The length of the LP block is then equal to $m_i^b = 1$;
- if $b = \{i, j\}$, we sample from the least cost population first and such that the frequencies are equal to the randomization probabilities

  $$x_i = \frac{|\delta_j|}{|\delta_i| + |\delta_j|}, x_j = \frac{|\delta_i|}{|\delta_i| + |\delta_j|}.$$

  The length of the LP block is then equal to $m_i^b + m_j^b = |\delta_j| + |\delta_i|$.

When the CELP (Eq. 5) is computed using $\hat{\mu}_n$, then at the end of the LP block, the expected average reward will be equal to $\hat{z}_n$ in the first case and equal to $(m_i^b + m_j^b) \cdot \hat{z}_n$ in the second case.

3.2 Sparse Block Policy

An important difference between a block-based policy and period-based policy is that the next step of the policy will be described in terms of blocks instead of periods. Here we are faced with the same three challenges as the SPP (Eq. 6) whilst having to conform to the stricter definition of feasibility. In my block-based policy, which I will refer to as the Sparse Block Policy (SBP) we will use sampling blocks and linear programming blocks to construct a consistent policy.

Similar to before, a sparse sequence will help us with obtaining estimates for the mean rewards of the populations whilst not affecting the average reward in the long run. Consider a sparse sequence of positive integers:

$$\{\tau_m\}_{m \in \mathbb{N}},$$

such that

$$\lim_{m \to \infty} \frac{\tau_m}{m} = \infty \text{ and } \tau_1 = 1.$$

This sparse sequence will determine when a sampling block $\{m_j^*\}_{j=1}^k$ will be performed and when the current block ℓ does not coincide with $\{\tau_m\}_{m \in \mathbb{N}}$, a linear programming block is performed. The Sparse Block Policy, π^{SC} can thus be described as follows

$$\pi^{\text{SC}}(\ell) = \begin{cases} \text{Sampling block} & \text{if } \tau_m = \ell \text{ for some } m \in \mathbb{N}, \\ \text{Linear Programming block} & \text{otherwise.} \end{cases} \tag{7}$$

This policy is consistent which I will prove in the next chapter.

4 Consistency of SBP

Consistency consists of two parts, proving feasibility (Def. 7) and proving that the policy converges with probability 1 to the optimal average reward that could be achieved under complete information (Def. 2).

4.1 Feasibility

It is trivial to prove that π^{SC} is a feasible policy according to the strict cost constraint. Our policy is defined as a sequence of blocks, determined by the sparse sequence $\{\tau_m\}_{m\in\mathbb{N}}$. By the definitions of the block in general and more specifically of the sampling block and the linear programming block, we know that at every point in the blocks we will stay under budget and thus remain feasible. Therefore π^{SC} is feasible, or equivalently

$$\forall n \in \mathbb{N} : \frac{\mathbb{E}^{\pi^{\mathrm{SC}}}[C_n^{\pi^{\mathrm{SC}}}]}{n} \leq C_0, \forall f \in F.$$

4.2 Convergence to optimum

If the policy has completed $\ell - 1$ blocks, then there are three possibilities: π^{SC} will perform a

- sampling block, or
- LP block using a nonoptimal BFS b, or
- LP block using an optimal BFS b.

Now, let $SB(\ell)$ be the amount of sampling blocks that have been performed after ℓ blocks, or equivalently

$$SB(\ell) = \sum_{t=1}^{\ell} \mathbb{1}\Big\{\tau_m = t, \text{ for some } m \in \mathbb{N}\Big\},$$

and let $L^b(\ell)$ be the amount of LP blocks using BFS b that have been performed after ℓ blocks, or equivalently

$$L^b(\ell) = \sum_{t=1}^{\ell} \mathbb{1}\Big\{b \in s(\underline{\hat{\mu}}_t), b \text{ is used in block } t\Big\}.$$

Combining both quantities provides us the following equation that captures all the possibilities in ℓ blocks:

$$\ell = SB(\ell) + \sum_{b \notin s(\underline{\mu})} L^b(\ell) + \sum_{b \in s(\underline{\mu})} L^b(\ell).$$

Dividing both sides by ℓ gives us the total block equation:

$$1 = \frac{SB(\ell)}{\ell} + \sum_{b \notin s(\underline{\mu})} \frac{L^b(\ell)}{\ell} + \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} \tag{8}$$

4.2.1 Sampling blocks

The first goal is to perform less and less sampling blocks in the long run. Eventually we will have learned the optimal solution and any sampling block after that point will drag down our average reward. Equivalently, we want

$$\lim_{\ell\to\infty} \frac{SB(\ell)}{\ell} = 0.$$

Proving this limit is quite trivial if we use the definition of $SB(\ell)$:

$$\lim_{\ell\to\infty} \frac{SB(\ell)}{\ell} = \lim_{\ell\to\infty} \frac{\sum_{t=1}^{\ell} \mathbb{1}\Big\{\tau_m = t, \text{ for some } m \in \mathbb{N}\Big\}}{\ell} = 0, \text{from (Def. 4)}.$$

4.2.2 Nonoptimal LP blocks

The second goal is to eventually perform as few as possible LP blocks using a nonoptimal BFS b. Performing these types of LP blocks will drag down our average reward. This goal is equivalent to the following limit:

$$\lim_{\ell \to \infty} \sum_{b \notin s(\underline{\mu})} \frac{L^b(\ell)}{\ell} = 0.$$

Using the definition of $L^b(\ell)$, we can note the following inequality:

$$L^b(\ell) = \sum_{t=1}^{\ell} \mathbb{1}\Big\{ b \in s(\underline{\hat{\mu}}_t), b \text{ is used in block } t \Big\} \leq \sum_{t=1}^{\ell} \mathbb{1}\Big\{ b \in s(\underline{\hat{\mu}}_t) \Big\}.$$

This follows from the fact that the solution set of the CELP (Eq. 5) can contain multiple optimal BFS b. In these cases, there are different $b_1, \dots, b_k \in s(\underline{\hat{\mu}}_t)$ that achieve the same average reward $\hat{z}_t$. If there are multiple optimal BFS, then one will be chosen from the set $s(\underline{\hat{\mu}}_t)$ where each $b_i \in s(\underline{\hat{\mu}}_t)$ has equal probability of being chosen. This means that if a BFS b is in $s(\underline{\hat{\mu}}_t)$, it will not necessarily be used in block t. In this limit we are only considering nonoptimal LP blocks, so for a period $t \in \mathbb{N}$ we know that for BFS b, $b \in s(\underline{\hat{\mu}}_t)$ and $b \notin s(\underline{\mu})$. The fact that b is not the true optimal BFS means that at least one estimate of the mean rewards of the populations is different enough from the true value that we get a different optimal BFS in period t than the true optimal BFS. Formalizing this gives us the following lemma, by Burnetas and Kanavetas [2].

Lemma 1 *For any $\underline{\mu}$, there exists $\varepsilon > 0$ such that for any $t = 1, 2, \dots$ if $b \in s(\underline{\hat{\mu}}_t)$ and $b \notin s(\underline{\mu})$ for some $b \in K$, then*

$$||\underline{\mu} - \underline{\hat{\mu}}_t|| \geq \varepsilon$$

Note that $||\cdot||$ is the supremum norm here.

From this lemma it follows that

$$\sum_{t=1}^{\ell} \mathbb{1}\Big\{ b \in s(\underline{\hat{\mu}}_t) \Big\} \leq \sum_{t=1}^{\ell} \mathbb{1}\Big\{ ||\underline{\mu} - \underline{\hat{\mu}}_t|| \geq \varepsilon \Big\}.$$

Earlier, we required $\underline{\hat{\mu}}_t$ to be a strongly consistent estimator, meaning $\underline{\hat{\mu}}_t \to \underline{\mu}$ a.s. We will converge to $\underline{\mu}$, so

$$\lim_{\ell \to \infty} \frac{1}{\ell} \sum_{t=1}^{\ell} \mathbb{1}\Big\{ ||\underline{\mu} - \underline{\hat{\mu}}_t|| \geq \varepsilon \Big\} = 0, \text{ a.s.}$$

and thus

$$\lim_{\ell \to \infty} \frac{L^b(\ell)}{\ell} \leq \lim_{\ell \to \infty} \frac{1}{\ell} \sum_{t=1}^{\ell} \mathbb{1}\Big\{ ||\underline{\mu} - \underline{\hat{\mu}}_t|| \geq \varepsilon \Big\} = 0, \text{ a.s.}$$

meaning that we will eventually never perform a nonoptimal LP block.

4.2.3 Optimal LP blocks

Combining both of these results with Eq. 8, we can see that

$$1 = \frac{SB(\ell)}{\ell} + \sum_{b \notin s(\underline{\mu})} \frac{L^b(\ell)}{\ell} + \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} \Rightarrow \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} = 1 - \frac{SB(\ell)}{\ell} - \sum_{b \notin s(\underline{\mu})} \frac{L^b(\ell)}{\ell}$$

$$\sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} = 1, \ell \to \infty, \text{ a.s,}$$

we will only perform optimal LP blocks in the long run. The last step is to prove that the average reward achieved by performing optimal LP blocks converges to the optimal reward under complete information. Let W_ℓ be the sum of outcomes from blocks where the optimal BFS b is used, then we want to prove that

$$\lim_{\ell \to \infty} \frac{W_\ell}{\ell} = z^*(\underline{\mu}) \text{ a.s.}$$

First we need some notation to define this limit. Let $s_\pi(\ell)$ be the total length of the first ℓ blocks, then let Y_l be the reward gained from block ℓ

$$Y_\ell = \sum_{t=s_\pi(\ell-1)}^{s_\pi(\ell)} X_t,$$

and then let $Y_{j,\ell}$ be the reward gained from population j in block ℓ

$$Y_{j,\ell} = \sum_{t=s_\pi(\ell-1)}^{s_\pi(\ell)} X_t \cdot \mathbb{1}\Big\{j \text{ is sampled from, in period } t\Big\}.$$

Now we can formulate W_ℓ as

$$W_\ell = \sum_{b \in s(\underline{\mu})} \sum_{t=1}^{\ell} \frac{Y_t \cdot \mathbb{1}\Big\{b \text{ is used in block } t\Big\}}{\sum_{j \in b} m_j^b}.$$

To prove this limit we will rewrite W_ℓ in terms of quantities that have known properties,

$$\begin{aligned} \frac{W_\ell}{\ell} &= \frac{1}{\ell} \sum_{b \in s(\underline{\mu})} \sum_{t=1}^{\ell} \frac{Y_t \cdot \mathbb{1}\Big\{b \text{ is used in block } t\Big\}}{\sum_{j \in b} m_j^b} \\ &= \frac{1}{\ell} \sum_{b \in s(\underline{\mu})} \frac{1}{\sum_{j \in b} m_j^b} \sum_{j \in b} \sum_{t=1}^{\ell} Y_{j,t} \cdot \mathbb{1}\Big\{b \text{ is used in block } t \text{ and } j \text{ is sampled from}\Big\}. \end{aligned}$$

Here we have split up the reward achieved per block into the populations that make up the optimal LP block. Recall that $L^b(\ell)$ denotes the amount of LP blocks under BFS b in the first ℓ blocks and m_j^b denotes how many times population j has been sampled under BFS b in the first ℓ blocks, then let $L_j^b(\ell)$ be the amount of times that population j has been sampled under BFS b in the first ℓ blocks

$$L_j^b(\ell) = m_j^b \cdot L^b(\ell).$$

Now, let $\overline{Y_{j,l}^b}$ be the sample mean of population j under BFS b after block ℓ

$$\overline{Y_{j,l}^b} = \frac{1}{L_j^b(\ell)} \sum_{t=1}^{\ell} Y_{j,t} \cdot \mathbb{1}\Big\{b \text{ is used in block } t\Big\}.$$

Multiplying $\overline{Y_{j,l}^b}$ by $L_j^b(\ell)$ allows us to replace $Y_{j,t}$:

$$\begin{aligned}\frac{W_\ell}{\ell} &= \frac{1}{\ell} \sum_{b \in s(\underline{\mu})} \frac{1}{\sum_{j \in b} m_j^b} \sum_{j \in b} \sum_{t=1}^{\ell} Y_{j,t} \cdot \mathbb{1}\Big\{b \text{ is used in block } t \text{ and } j \text{ is sampled from}\Big\} \\ &= \frac{1}{\ell} \sum_{b \in s(\underline{\mu})} \frac{1}{\sum_{j \in b} m_j^b} \sum_{j \in b} L_j^b(\ell) \cdot \overline{Y_{j,l}^b}\ .\end{aligned}$$

The last step of rewriting will be to multiply by $\frac{L^b(\ell)}{L^b(\ell)}$, which gives us

$$\begin{aligned}\frac{W_\ell}{\ell} &= \frac{1}{\ell} \cdot \frac{L^b(\ell)}{L^b(\ell)} \sum_{b \in s(\underline{\mu})} \frac{1}{\sum_{j \in b} m_j^b} \sum_{j \in b} L_j^b(\ell) \cdot \overline{Y_{j,l}^b} \\ &= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} \sum_{j \in b} \frac{L_j^b(\ell)}{L^b(\ell) \cdot \sum_{j \in b} m_j^b} \cdot \overline{Y_{j,l}^b}\ .\end{aligned}$$

Therefore, we get an expression that is similar to Eq. 3:

$$\frac{L_j^b(\ell)}{L^b(\ell) \cdot \sum_{j \in b} m_j^b} \sim x_j \text{ and } \overline{Y_{j,l}^b} \sim \mu_j, \ell \to \infty \text{ a.s.}$$

The denominator of the fraction is how many times population j is sampled while the numerator is the total amount of times both populations in BFS b are sampled and thus together they perform like the sampling frequency x_j. Now, let

$$z_\ell^b = \sum_{j \in b} \frac{L_j^b(\ell)}{L^b(\ell) \cdot \sum_{j \in b} m_j^b} \cdot \overline{Y_{j,l}^b}\ ,$$

then we get the following expression

$$\begin{aligned}\frac{W_\ell}{\ell} - z^* &= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} \sum_{j \in b} \frac{L_j^b(\ell)}{L^b(\ell) \cdot \sum_{j \in b} m_j^b} \cdot \overline{Y_{j,l}^b} - z^* \\ &= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} z_\ell^b - z^*.\end{aligned}$$

Now, let $L(\ell)$ be the amount of LP blocks that use an optimal BFS b in the first ℓ blocks,

$$L(\ell) = \sum_{b \in s(\underline{\mu})} L^b(\ell),$$

then we can rewrite the expression in order to get

$$\begin{aligned}
\frac{W_\ell}{\ell} - z^* &= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} z^b_\ell - z^* \\
&= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} z^b_\ell - z^* + \frac{L(\ell)}{\ell} z^* - \frac{L(\ell)}{\ell} z^* \\
&= \sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} (z^b_\ell - z^*) - (1 - \frac{L(\ell)}{\ell}) z^*.
\end{aligned}$$

Earlier we proved that in the long run we only perform optimal LP blocks so

$$\lim_{\ell \to \infty} \frac{L(\ell)}{\ell} = 1 \text{ and thus } \lim_{\ell \to \infty} (1 - \frac{L(\ell)}{\ell}) z^* = 0 \text{ a.s.}$$

To complete the proof we need so show that

$$\sum_{b \in s(\underline{\mu})} \frac{L^b(\ell)}{\ell} (z^b_\ell - z^*) \to 0 \text{ a.s } \forall b \in s(\underline{\mu})$$

$L^b(\ell)$ is a random variable that is increasing in ℓ and $0 \leq L^b(\ell) \leq \ell$ because in ℓ blocks we can perform at most ℓ LP blocks. Due to this, either $L^b(\ell) \to \infty$ or $L^b(\ell) \to M$ for some $M < \infty$. Now let D and D^c represent these two events respectively,

$$D = \Big\{ L^b(\ell) \to \infty \Big\} \text{ and } D^c = \Big\{ L^b(\ell) \to M \Big\},$$

and let $\mathbb{P}(D) = p$ and $\mathbb{P}(D^c) = 1 - p$ with $p \in \mathbb{R}_{>0}$.
Also let

$$A = \Big\{ \lim_{\ell \to \infty} \frac{L^b(\ell)}{\ell} (z^b_\ell - z^*) = 0 \Big\},$$

then

$$\mathbb{P}(A) = p \cdot \mathbb{P}(A|D) + (1 - p) \cdot \mathbb{P}(A|D^c),$$

by the law of total probability. Now, we know that $\frac{L^b(\ell)}{\ell} \leq 1$ for all $\ell \in \mathbb{N}$, so

$$\begin{aligned}
\mathbb{P}(A|D) &= \mathbb{P}\bigg(\lim_{\ell \to \infty} \frac{L^b(\ell)}{\ell} (z^b_\ell - z^*) = 0 \mid \lim_{\ell \to \infty} L^b(\ell) = \infty \bigg) \\
&\geq \mathbb{P}\bigg(\lim_{\ell \to \infty} (z^b_\ell - z^*) = 0 \mid \lim_{\ell \to \infty} L^b(\ell) = \infty \bigg).
\end{aligned}$$

Given that $L^b(\ell) \to \infty$ as $\ell \to \infty$, we know that we will keep performing these optimal LP blocks and we can thus apply the strong law of large numbers,

$$\begin{aligned}\mathbb{P}(A|D) &\geq \mathbb{P}\left(\lim_{\ell\to\infty}(z_\ell^b - z^*) = 0 \mid \lim_{\ell\to\infty} L^b(\ell) = \infty\right)\\ &= 1.\end{aligned}$$

In the second event D^c, the difference $z_\ell^b - z^*$ is bounded for any $\ell \in \mathbb{N}$, so

$$\mathbb{P}(A|D^c) = \mathbb{P}\left(\lim_{\ell\to\infty}\frac{L^b(\ell)}{\ell}(z_\ell^b - z^*) = 0 \mid \lim_{\ell\to\infty} L^b(\ell) = M\right) = 1.$$

Therefore $\mathbb{P}(A) = 1$ and thus with

$$\lim_{\ell\to\infty}\frac{L^b(\ell)}{\ell}(z_\ell^b - z^*) = 0 \text{ a.s, } \forall b \in s(\underline{\mu}),$$

the proof is complete. When performing optimal LP blocks, the Sparse Block Policy will converge to the average reward achieved under complete information, or equivalently

$$\lim_{\ell\to\infty}\frac{W_\ell}{\ell} = z^*(\underline{\mu}),$$

and thus the Sparse Block Policy is consistent.

5 Simulations

Now that we have proven that SBP (Eq. 7) possesses the theoretical properties we desire, we can study its behaviour in practice via simulations. Whilst convergence in the long run is a great property, other considerations like rate of convergence compared to other solutions, growth of regret and performance in different mean configurations are also important.

5.1 Optimal sparse sequence

There is a lot of freedom in the choice of sparse sequence. Any sequence that conforms to Definition 4 can be used in an implementation of SBP. This does not mean however that every sparse sequence performs the same when used in the SBP. We will be looking at sparse sequences of a power function form:

$$\{\tau_m^b = m^b, m = 1, 2, \ldots\}.$$

For $b \in (1, \infty)$, $\{\tau_m^b\}_{m\in\mathbb{N}}$ is a sparse sequence. Let $b_1, b_2 \in (1, \infty)$ such that $b_1 < b_2$. If we perform the SBP with both $\{\tau_m^{b_1}\}$ and $\{\tau_m^{b_2}\}$ then we would see that we perform more sampling blocks with the first sequence. This could be both advantageous and disadvantageous. If we perform many sampling blocks, then the mean estimates would become accurate very soon and thus we find the optimal solution sooner. However, performing many sampling blocks also means sampling a lot from nonoptimal populations and this can mean that the average reward will deviate longer from the average reward achieved under complete information. The sparser sequence $\{\tau_m^{b_2}\}$ can have the inverse of this problem. If we quickly find the optimal solution, then performing few sampling blocks is great and allows us to converge faster to the optimal reward. Conversely, if we do not have accurate mean estimates, then we could be sampling from a nonoptimal solution for many blocks. Burnetas and Kanavetas [2] found that for the SPP (Eq. 6) $b = 2$ was optimal out of $(1.2, 1.5, 2, 3, 5)$.

In a sampling block we sample from all populations at least once. In the SPP, whenever the period coincides with a sparse sequence one population is sampled once. This means that in the SBP if the block count coincides with the sparse sequence, we learn more about the populations then when the same happens in the SPP. Therefore, we expect that a larger b will be optimal for the SBP. To study this, we will simulate an instance of the multi-armed bandit problem with $k = 4$ arms. The outcomes of arm i follow a binomial distribution with the parameters (N, p_i) where $N = 5$ and $p_1 = 0.3, p_2 = 0.5, p_3 = 0.9, p_4 = 0.8$. Therefore the mean reward vector is equal to $\underline{\mu} = (1.5, 2.5, 4.5, 4)$. The cost vector is $c = (3, 4, 8, 10)$ and our budget is equal to $C_0 = 5$. Given this set of parameters, the optimal policy is $\underline{x} = (0, \frac{3}{4}, \frac{1}{4}, 0)$ and the optimal average reward is $z^*(\underline{\mu}) = 3$. We will compare the performance of the SBP using a sparse sequence of power function form with $b \in \{1.2, 2, 3, 5, 7\}$.

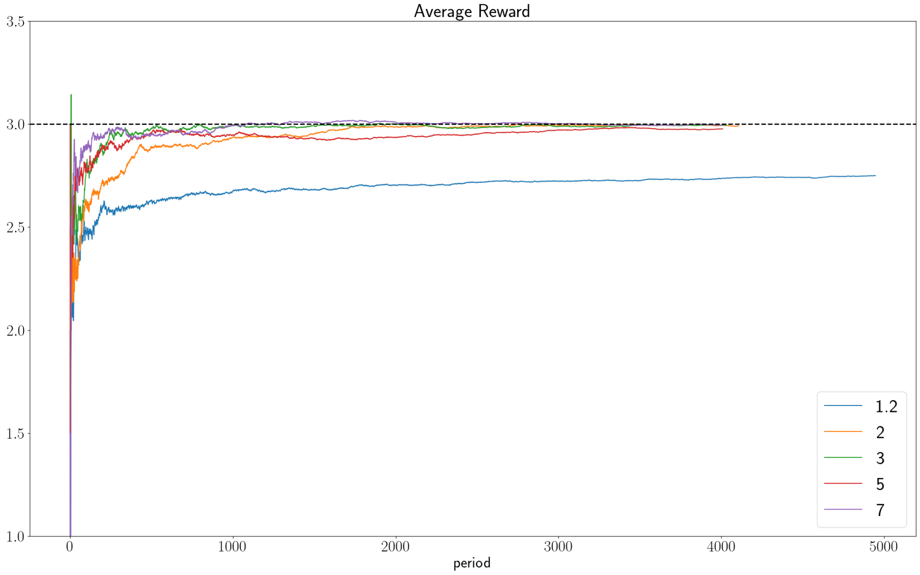


Figure 1: Performance when optimal solution is found early on

In Figure 1 we observe a case where each instance of SBP found the optimal solution quite fast. The sequences with $b \in \{2, 3, 5, 7\}$ are close together and deviate very little from the optimal value. The sequence with $b = 1.2$ however deviates a lot more despite having found the optimal solution too. As discussed above, the relatively large amount of sampling blocks drags down the average reward achieved by this instance of the SBP.

The cases where the solution is not immediately found are more interesting. Sequences where $b = 5$ and $b = 7$ are a lot more volatile in these circumstances. They are very sparse, so if they do not find the optimal solution then they will perform a lot of LP blocks with a nonoptimal solution. In Figure 2 we observe instances of the SBP with $b = 5$ and $b = 7$ performing a lot worse, similar to the instance with $b = 1.2$. In rare cases, instances of SBP with $b \in \{2, 3\}$ can also perform like the sequences with $b \in \{5, 7\}$ in Figure 2 but this is quite rare due to their higher number of sampling blocks.

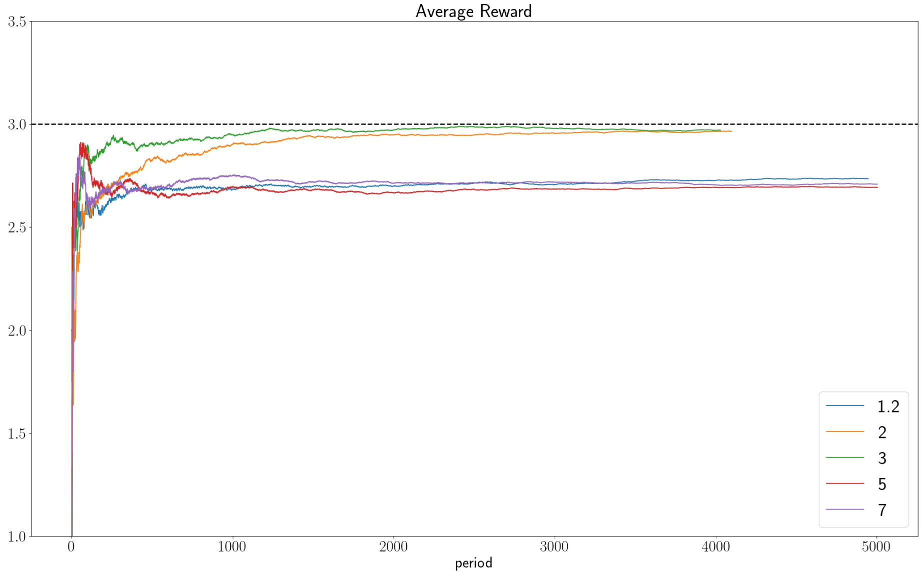


Figure 2: Large deviation for sequences with $b \in \{5, 7\}$

An important thing to note is the structure of the sampling block. In this configuration the sampling block $\{m_j^*\}_{j=1}^4 = \{4, 1, 1, 1\}$. The first population is sampled 4 times and the others only once. The first population is not part of the optimal solution. Whenever an instance of the SBP does not find the optimal solution $\underline{x}^* = (0, \frac{3}{4}, \frac{1}{4}, 0)$, it finds $\underline{x} = (\frac{3}{5}, 0, 0, \frac{2}{5})$. This could be due to sampling 4 times from the first population. To test this, we will change the cost vector to $c = (8, 3, 6, 10)$. As a consequence, the sampling block becomes $\{m_j^*\}_{j=1}^4 = \{1, 5, 1, 1\}$ and the optimal sampling policy becomes $\underline{x} = (0, \frac{1}{3}, \frac{2}{3}, 0)$. Therefore the new optimal average reward achieved under complete information is $z^*(\underline{\mu}) = 3.8\overline{3}$. In Figure 3 we see a similar picture to Figure 1. Here again all the sequences found the optimal solution early on. The sequence with $b = 1.2$ still has the largest deviation compared to the rest. However, whilst not as large as the instance with $b = 1.2$, when using the sequence with $b = 2$ there is also quite a deviation from the optimal average reward compared to sequences with $b \in \{3, 5, 7\}$. It could be that the average reward gained from the sampling block is now lower compared to the optimal average reward and therefore when performing more sampling blocks, the average reward will deviate more.

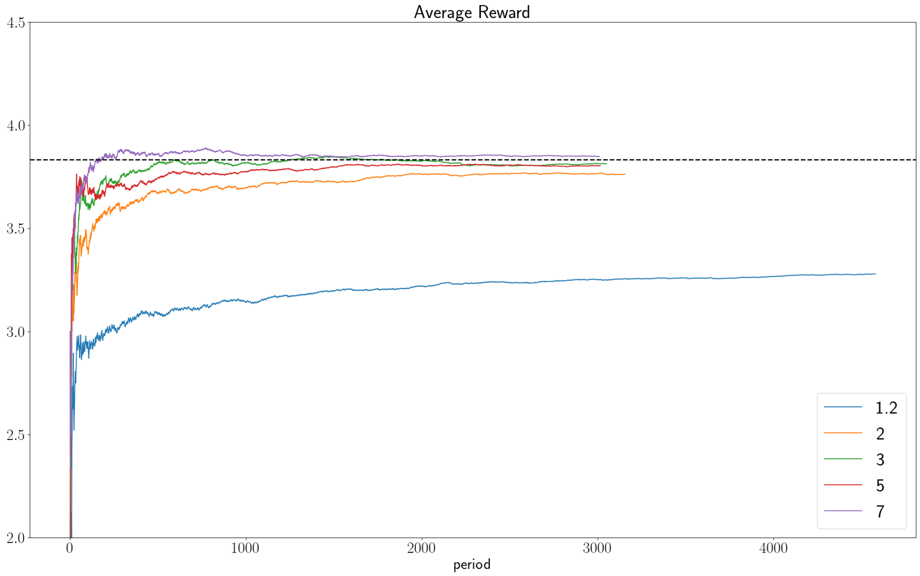


Figure 3: General performance in the modified configuration

Taking both of these configurations into account, for the SBP a sparse sequence of the power function form with $b = 3$ performs the best compared to $\{1.2, 2, 5, 7\}$. The SBP with $b = 1.2$ performs the worst due to the large number of sampling blocks causing it to deviate a large amount from the optimal average reward achieved under complete information. Instances with $b \in \{5, 7\}$ can perform well but their relatively low amount of sampling blocks makes their performance dependent on whether they find the optimal solution early on. If they do, the performance is great. If they do not, then they have a large deviation for a long time because they will perform a large amount of nonoptimal LP blocks before the next sampling blocks. The SBP with $b = 2$ performs well for one type of sampling block, but not for another type. Due to the fact that an instance of SBP with $b = 3$ performs well in both scenarios, we consider it the best, for this type of sparse sequence.

5.2 Comparison of Greedy, SPP and SBP

In the process of constructing the SBP, we took inspiration from the SPP. They are very similar in their approach to solving the multi-armed bandit problem with the only difference being in the way they obtain estimates for all populations without violating their cost constraints. This difference could lead to a difference in performance. To investigate this, we will simulate the same multi-armed bandit problem instance we used at the start of this chapter. For the SBP, we will use a sparse sequence of power function form with $b = 3$ whilst the SPP will use sparse sequences of the form:

$$\{\tau^b_{j,m} = \ell_j + m^b, m = 1, 2, \ldots\}, j = 1, \ldots, k,$$

where ℓ_j are defined constants per population such that the sequences are nonoverlapping. Burnetas and Kanavetas [2] found that for the SPP these sparse sequences with the parameter $b = 2$ are optimal. We will also add in a greedy policy to highlight the importance of the sparse sequences. The greedy policy will perform one sampling block at the start and afterwards only perform LP blocks.

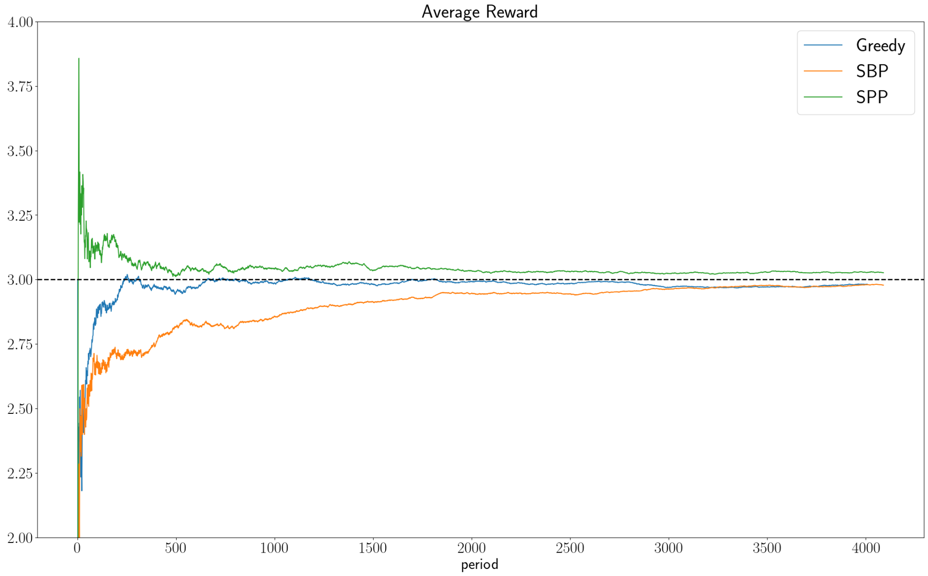


Figure 4: Similar performance a cross all three policies

In Figure 4 we can see that all the policies can perform well. The greedy policy performs well here because after its sampling block it found the optimal solution. However in Figure 5 we see that the greedy policy can also perform quite badly. It did not find the optimal solution after its sampling block and thus only performed nonoptimal LP blocks. The difference between the greedy policy and the adaptive policies SBP, SPP is that the greedy policy is not guaranteed to achieve the optimal average reward. The SBP and SPP do not necessarily find the optimal solution in the beginning but we have proven that in the long run they will find the optimal solution and achieve the optimal average reward. The greedy policy can get stuck in a local optimum which is what has happened in Figure 5. The performance between the SBP and SPP is largely similar but there is a difference. In Figure 5 the SPP line has a spike above the optimal average reward in the beginning. This is due to the forced samplings of the SPP. In the first 4 periods each population is sampled and due to their stochastic nature, we can receive very high rewards from these samplings, causing the spike. This happens less with the SBP because we sample more from the inexpensive population with a lower mean reward and afterwards once from the others. The average reward is less sensitive to a spike in the beginning due to the extra samplings in the sampling block.

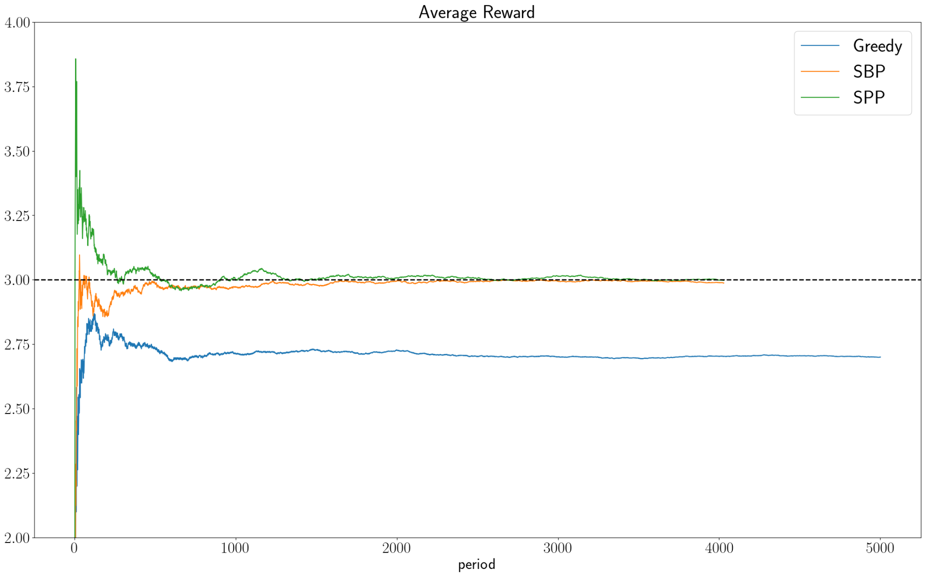


Figure 5: Large deviation by the greedy policy

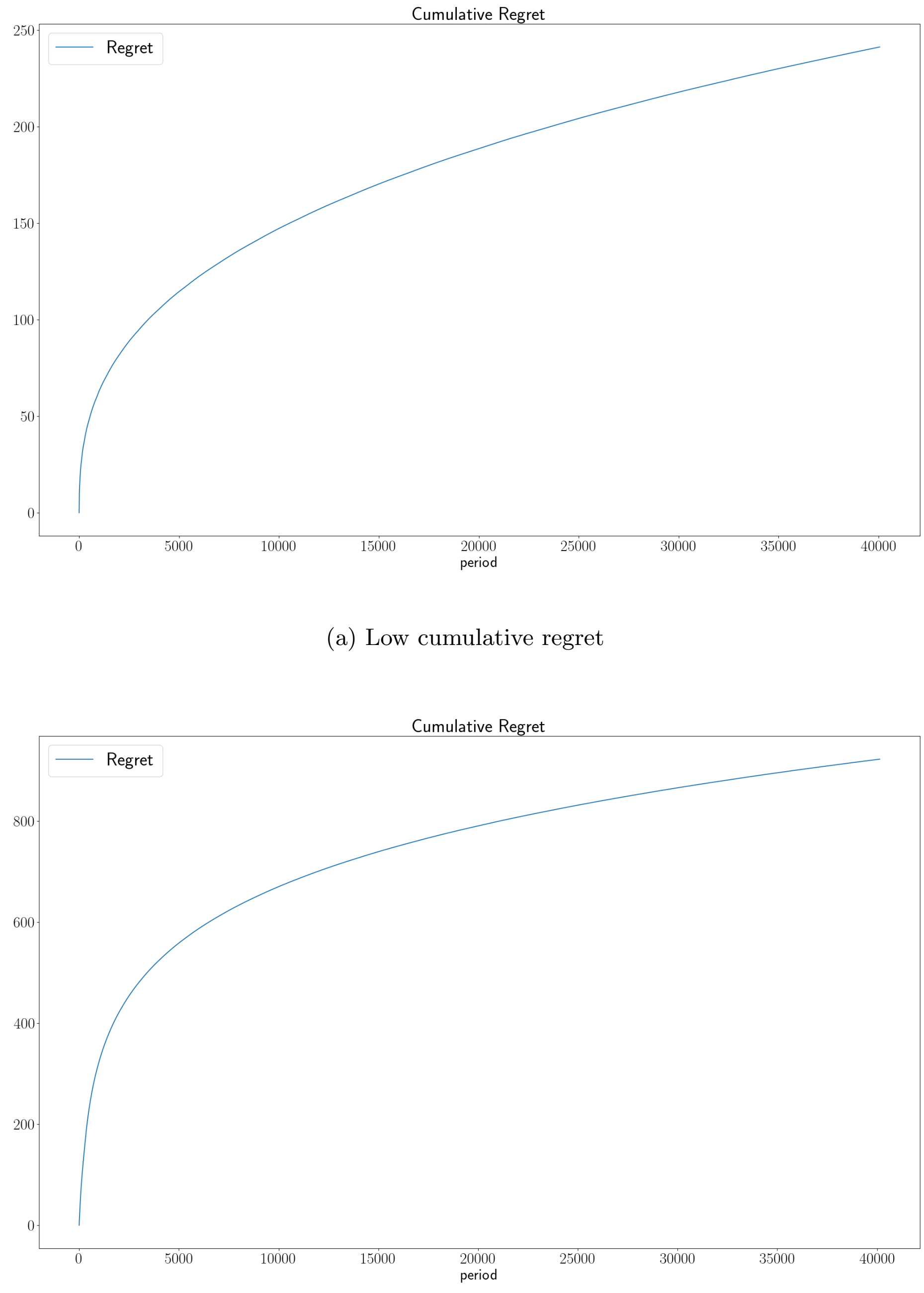
5.3 Growth of regret

Another important aspect to consider is the growth of the cumulative regret. To analyse this, we will be looking at the expected difference between the expected average reward and the optimal average reward.

Definition 8 (Expected regret) *The expected difference between the average reward and the optimal average reward under a consistent policy π is given by:*

$$R_n^\pi(\underline{\mu}) = \mathbb{E}^\pi\left(\frac{W_n}{n}\right) - z^*(\underline{\mu}).$$

We know that SBP is a consistent policy and thus $\frac{W_n}{n}$ converges almost surely to $z^*(\underline{\mu})$. However this does not imply the same for convergence in expectation without further assumptions on the population distributions. To investigate to growth of the cumulative growth, we will consider the summation of R_n^π from $n = 1$ to the current period. In Figure 6a and 6b we can see a clear logarithmic growth curve in both figures. The difference can be found in the scale of the y-axis. In Figure 6b the optimal solution was not found early on and that is why the cumulative regret is for larger here than in Figure 6a.



(a) Low cumulative regret

(b) High cumulative regret

5.4 Influence of means and variances

5.4.1 Small differences between means

An important consideration in the construction of our consistent policy is the strongly consistent estimator. For the SBP, the sample mean was chosen under Assumption 1. The adaptive policy relies on the samples to accurately distinguish each mean. Depending on the distributions, accurately estimating the mean rewards can be more or less difficult. If we consider an instance of the multi-armed bandit problem with $k = 4$ arms, but change the parameters of the binomial distribution to $N = 5$ and $p_1 = 0.6, p_2 = 0.65, p_3 = 0.63, p_4 = 0.62$ then we get the new mean vector $\underline{\mu} = (3.0, 3.25, 3.15, 3.1)$. The cost vector stays the same, $c = (3, 4, 8, 6)$. The optimal sampling policy now becomes simply $\underline{x} = (0, 1, 0, 0)$. In Figure 7 we can see that the SBP can find the optimal solution quickly despite the close means. However, this fast convergence to the optimal solution is less likely then in the earlier configuration with greater differences between the means. In Figure 8a we can see that in 1000 blocks the optimal solution was not found and this can even continue whilst performing 5000 blocks. The SBP has found a certain solution that seems optimal and due to the small differences between the means, it takes more time to find the actual optimal solution. At the same time, due to these small differences between the means the difference between the optimal average reward and the average reward obtained by a nonoptimal solution is also small.

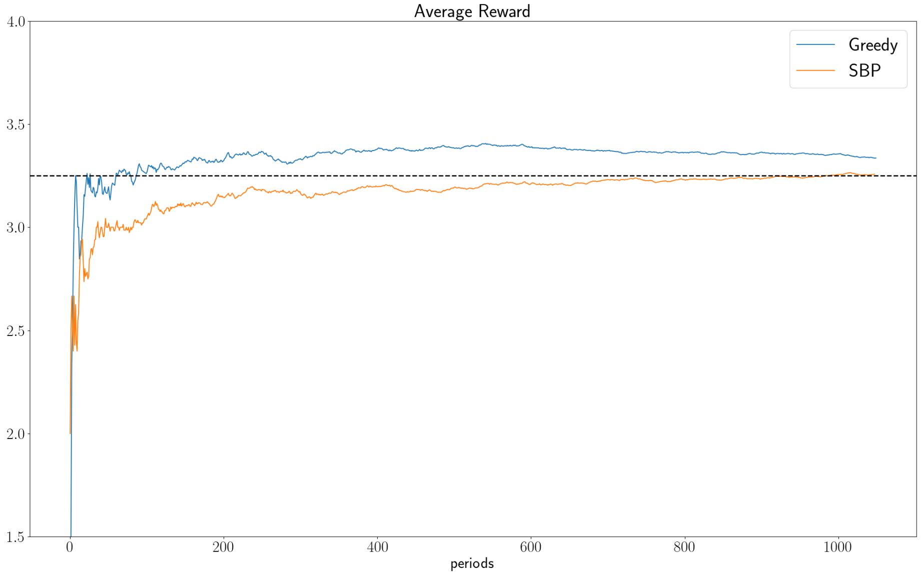
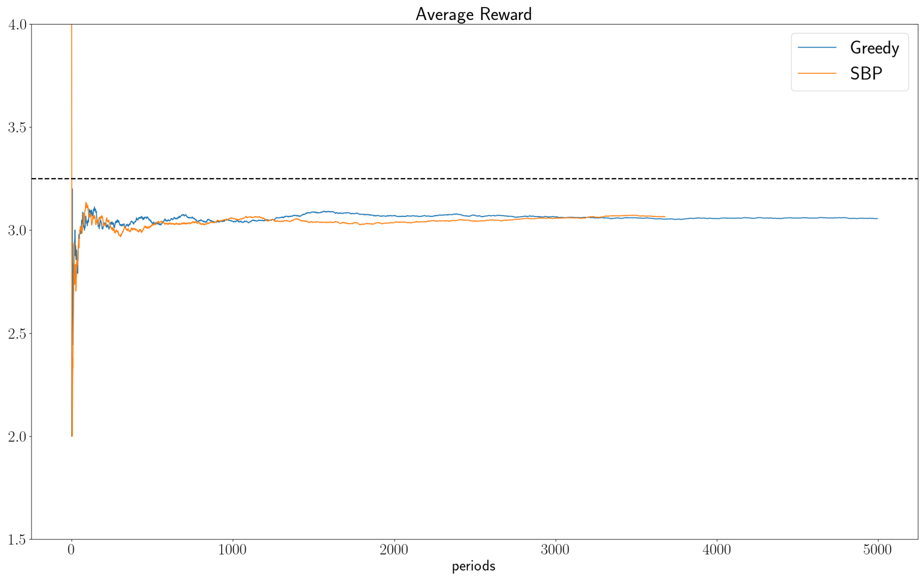
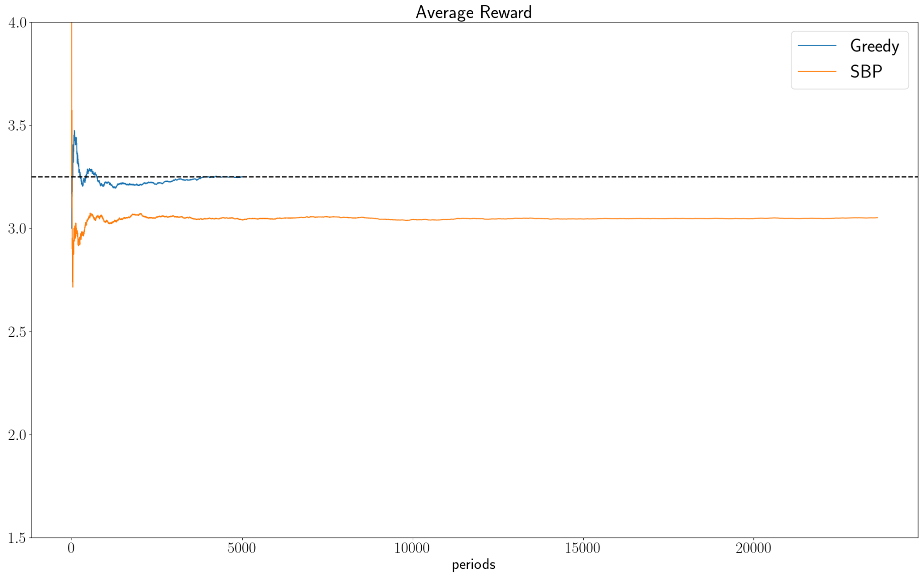


Figure 7: Success in accurately distinguishing close means



(a) Failure in accurately distinguishing close means



(b) Failure in 5000 blocks

5.4.2 High and low variance

Another aspect of the distributions that can influence the accuracy of the estimators is the variance. Two populations with different means can be difficult to distinguish if their variance causes their samples to overlap. To study this, we will consider an instance of the multi-armed bandit problem with $k = 4$ arms. These populations will be distributed following a normal distribution to allow us to directly control the variances. The distribution of arm i is given by μ_i and σ^2 where $\mu_1 = 7, \mu_2 = 5, \mu_3 = 10, \mu_4 = 8$. We will perform the SBP four times, each time with a different variance, where $\sigma^2 \in \{1, 2, 5, 7\}$. In Figure 9 we can see that most of the impact of the higher variance is in the beginning. In the long run the influence of higher variance is dampened by the high amount of samples taken. Sometimes a higher variance takes longer than the others to converge to the optimal average reward but eventually it will become stable and achieve the optimal average reward.

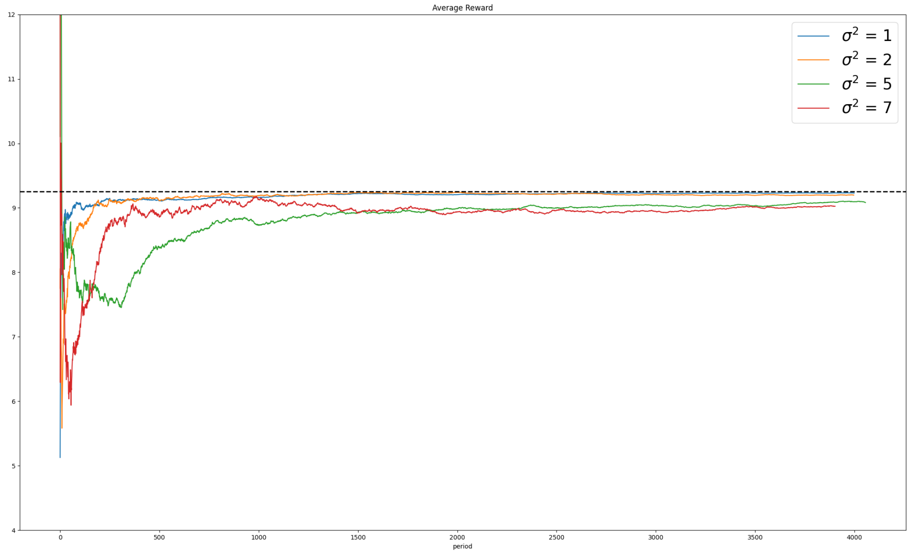


Figure 9: Variance has most impact in the beginning

6 Conclusion

After taking inspiration from the work of Burnetas and Kanavetas [2], we have constructed a consistent policy for the multi-armed bandit problem with a strict cost constraint. We proved that it converges to the optimal average reward achieved under complete information in all cases. Afterwards, we performed some simulations that demonstrate the important qualities of this policy. First, we determined the optimal sparse sequence of power function form. After, we saw that it can outperform the greedy policy by avoiding getting stuck in local optima and it performs similarly to the SPP (Eq. 6) without ever exceeding the budget. We have also shown that the growth of the regret is logarithmic. Lastly, the SBP can perform well in challenging scenarios with small differences between means or high variance. Usually, it is able to find the optimal solution quickly and even it is stuck with a nonoptimal solution, eventually it will find the optimal solution and achieve the optimal average reward by our proof in Chapter 4.

References

[1] Herbert E. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58:527–535, 1952.

[2] A. N Burnetas and O. A Kanavetas. *Applications of Mathematics and Informatics in Military Science*, volume 71. Springer, 2012.

[3] A. N Burnetas, O. A Kanavetas, and M. N Katehakis. Asymptotically optimal multi-armed bandit policies under a cost constraint. *Probability in the Engineering and Informational Sciences*, 31(3):284–310, 2017.