A. Roodenburg

Let bandits tell you when to maintain your machines

A study of the maintenance planning problem

Bachelor thesis

June 2025

Thesis supervisor: Dr. D. van der Hoeven

Leiden University
Mathematical Institute

Abstract

In this thesis algorithms are developed that learn when to plan maintenance. In essence, the main challenge is optimizing the planning problem wherewith cost can be saved by maintaining the part until just before it breaks. The core of the challenge lies in the fact that we do not know when the part will break. The different algorithms that are developed in this paper have the choice between exploring: gaining more knowledge about the failure time by letting the part break and paying for the repair, or exploiting: to plan maintenance early and thereby save cost but not knowing how long the part would have functioned. Notice that letting the part break is expensive, but planning maintenance too early, and thereby too often, is expensive as well.

Three algorithms were designed, which all use a different strategy. The first algorithm has an exploration phase and then an exploitation phase. Which means that this algorithm lets the part break for a certain amount of times, and after that it plans maintenance on the day that would have given the smallest loss in the explore phase. The second algorithm has an 'explore while exploiting' strategy. It does so by being optimistic about the expected loss of the possible actions: not choosing the best known option, but choosing what could be the best option. And the third algorithm exploits only: every round it chooses the best known option.

These algorithms are then compared by mathematical analysis and simulations of the pseudo regret. From this is concluded what the strong and weak points of every algorithm are. The discussion contains suggestions for further analysis.

Contents

1 Introduction

When working with machines, one knows that if no maintenance is done, after a certain period of time the machine will break down. Machines break down when parts of them fail. To make the machine work again, the particular part needs to be repaired against a certain cost. This can be expensive when a part needs to be ordered, a mechanic needs to install it, and because the machine does not work for a period of time. However, it is also possible to maintain the part before it fails. If it costs more to maintain a part than to repair a part, it is trivial that the optimal strategy is to let the part break. Therefore, this thesis treats the case where costs can be saved by maintaining the part before it breaks. But planning maintenance is no easy task: when a part is maintained too early money could be saved by maintaining later, but when maintenance is planned too late the part will break and needs to be repaired.

The problem is first defined, and the type of algorithm that will be used is introduced. To design a good maintenance strategy, some key insights are given: about what information we gain after choosing an action, that the best day to maintain a part is the day before it breaks, and that choosing the best day to maintain is based on the probability distribution of the failure time of the part. Using these key insights, a formula for the expected loss of every action is formulated. The algorithms will later use this formula to try to find the optimal action.

After fully understanding the problem, three algorithms are designed and compared by mathematical analysis and simulations. The simulations allow us to observe the behavior of the algorithms and to see when and why the algorithms do or do not work. The analysis shows how good the algorithms perform for every given set of parameters. Finally, the strong and weak points of the three algorithms are highlighted and discussed.

1.1 Problem definition

At the start two constants are given: the cost of repair $C_r > 0$ and the cost of maintenance $C_m > 0$. As discussed in the introduction, this thesis focuses on the case where costs can be saved by maintaining the part before it breaks, therefore $C_r > C_m$. Then T rounds are played. Each round we choose to maintain the part on a certain day, and the part will break on a certain day. In other words: each round $t \in \{1, \ldots, T\} =: [T]$ we choose action $a_t \in [K]$ and the part will break on day $f_t \in [K]$, where K is chosen such that $f_t \in [K]$ for all t. If the part is still functioning when maintenance is planned, we pay C_m/a_t. If the part breaks on day f_t before maintenance is planned, we pay C_r/f_t. This gives the following loss function for action a in round t

$$\ell_t(a) = \frac{C_m}{a}\mathbb{I}\{a < f_t\} + \frac{C_r}{f_t}\mathbb{I}\{a \geq f_t\}.$$

To decide which algorithm works best, we define the pseudo regret: the sum over all rounds of the expected loss of the action chosen by the algorithm minus the the expected loss of the optimal action. As we will find out in Section 2.3, the expected loss of an action depends on the probability distribution of the failure time of the part. The optimal action is defined as the action that has the minimal expected loss based on the true probability distribution of the failure time. Thus the pseudo regret is

$$R_T = \mathbb{E}\left[\sum_{t=1}^{T} \left(\ell_t(a_t) - \ell_t(a^*)\right)\right],$$

where a_t is the action chosen in round t and $a^* = \arg\min_{a \in [K]} \mathbb{E}[\ell_t(a)]$ is the optimal action. Notice that a^* is the same every round since the probability distribution of the failure time is the same every round. We define the algorithm that works best as the algorithm with the smallest pseudo regret. In Chapter 3 we will try to bound the pseudo regret of the algorithms.

1.2 Bandit algorithms

In Chapter 3 three algorithms will be designed to decide when to plan maintenance. Decision making algorithms which make a new decision every round based on the knowledge gained in the rounds before, are called Bandit algorithms. This name comes from the classic problem of the multi-armed bandits, where a gambler has the choice to play K different slot machines. These slot machines are also known as one-armed bandits, hence the name for the type of algorithm. The gambler wants to play the slot machine that has the highest payout distribution, but the only way to figure out their payout distribution is to play them. Therefore the gambler needs to choose between exploring all the possible options and exploiting his best known option.

An example of a bandit is the Epsilon Greedy algorithm. A gambler using this strategy will start with a 'explore only' phase and after that an 'exploit only' phase. So first he plays each slot machine a certain amount of rounds, and after that he will only play on the machine that gave him the highest payout. The Epsilon Greedy algorithm is introduced in Auer et al., 2002.

Another bandit algorithm is the Upper Confidence Bound algorithm. This algorithm does not explore and then exploit, but rather explores while exploiting. A gambler using this strategy is optimistic about the payout of the slot machines. So if he knows that one machine will give him a payout between 9$ and 11$ and the other will give him a payout between 4$ and 12$, he will not choose the first machine based on the average payout but he will choose the second machine because he is optimistic and wants 12$. The Upper Confidence Bound algorithm is also introduced in Auer et al., 2002.

Notice that this gambling problem is different from the maintenance planning problem. The gambler wants to maximize payout while we want to minimize costs, the gambler needs to estimate K different probability distributions while we only need to estimate the probability distribution of the failure time (which will be shown in Section 2.3), and the gambler only gains knowledge from a slot machines when he plays it, while we also gain knowledge about the days before maintenance is done. What knowledge is exactly gained after a certain action is chosen is discussed in Section 2.1. Although the gambling problem is different from the maintenance planning problem, the concepts of the two algorithms introduced above will be used to design algorithms to solve the maintenance planning problem in Chapter 3.

2 Understanding the problem

To develop a good maintenance strategy, it is important to first fully understand the problem. In this chapter we highlight the key concepts of our setting; the feedback model, the loss function and since the goal is to minimize the expected regret; the expected loss.

2.1 Feedback model

At the start of each round is decided on which day to do maintenance based on the knowledge gained in the rounds before. Therefore we need to look carefully at what feedback is given when choosing an action, and what information is useful when planning maintenance.

In round t, when a part is maintained on day a_t it is possible to calculate the loss for days $a \leq a_t$, but not for days $a > a_t$ as the failure time f_t is not observed (recall the loss function given in Section 1.1). This can also be seen in the first graph of Figure 1. Where the arrows represent for which days feedback is received after choosing action a. When maintenance is planned on day a_t, but the part breaks down on day $f_t < a_t$, the failure time is observed and therefore it is possible to calculate the loss for all actions $a \in [K]$. This can be seen in the second graph of Figure 1. Where this time the arrows also go to all nodes on the right of a. This might seem like a rich feedback model because it either gives feedback for actions $a \in [a_t]$ or for every action, but don't be fooled: not all information is useful. Notice that $f_t = a$ is never observed for $a > a_t$. This means that the observations for $a > a_t$ in the case of $f_t \leq a_t$ are biased, since it tells us that we have not seen the part break on day a while we did not have the opportunity to actually see it break. Therefore, if this information is used to estimate the probability that the part breaks on day a, this probability is underestimated for large a. Thus in the analysis, we will only use the unbiased feedback. We define the number of times were feedback is received for day a, as the number of times when we could have seen the part break, which is

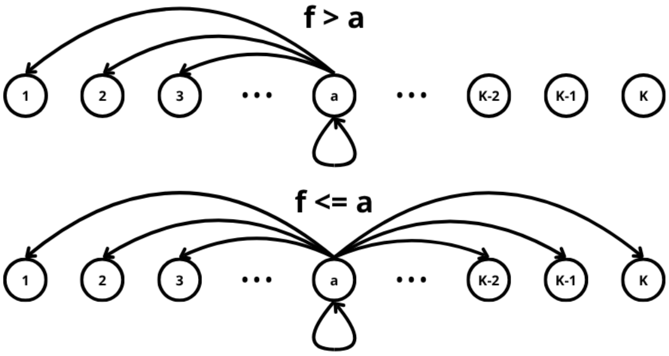

Figure 1: graph representation of the feedback model

$$n_t(a) = \sum_{s=1}^{t} \mathbb{I}\{a \leq a_t\}.$$

This also means that the only way to receive feedback for every action a, is to select $a_t = K$. We define receiving feedback for every action as receiving full feedback.

2.2 Behavior of the loss function

Since we are looking for the action that gives us the smallest loss, it is good to understand the behavior of the loss function. In Figure 2 we can see that the best day to maintain the part is the day before it breaks. But the problem is that we do not know when the part will fail. In this figure we can also see that the loss is constant after the part has broken. This means that if the part breaks, we would rather be far to late with planning maintenance than just to late, because $\ell(f_t) = \ell(K)$. In other words: $a_t = f_t$ gives you the same loss as $a_t = K$, but the second option gives you more unbiased feedback.

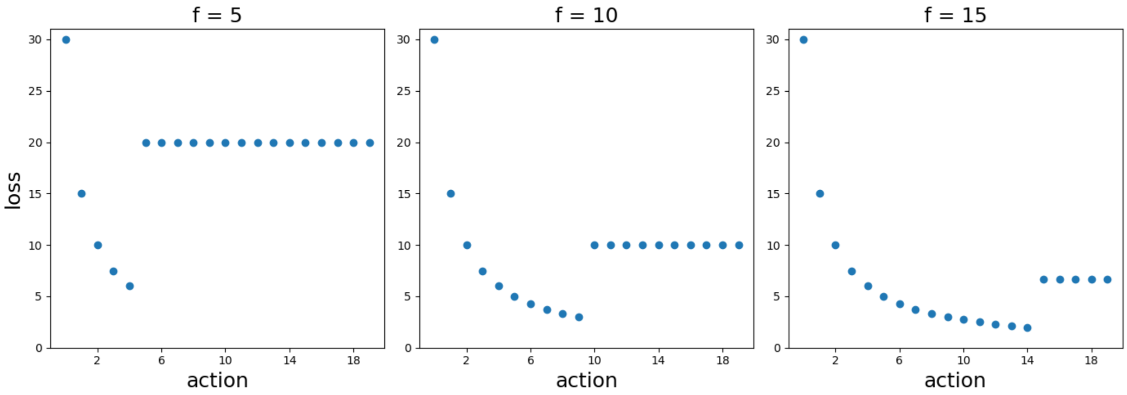

Figure 2: loss functions where $C_r = 100, C_m = 30$ and the failure time is varied

2.3 Expected loss

As mentioned earlier the goal is to minimize the pseudo regret, which is the expected loss of the actions chosen by the algorithm minus the expected loss of the optimal choice. So to bound the pseudo regret, it is necessary to work out the expected value of the loss function. And the algorithms will need a function that they can use to determine which action will give them the minimum loss based on their observations.

Theorem 1 (Expected loss)**.** *Let $C_r, C_m \in \mathbb{R}_{>0}$ such that $C_r > C_m$, and $K, T \in \mathbb{N}_{>0}$ such that $f_t \in [K]$ for all $t \in [T]$. Then the expected loss of action a is equal to*

$$\mathbb{E}[\ell_t(a)] = \frac{C_m}{a} + \sum_{i=1}^{a-1} \left(\frac{C_r}{i} - \frac{C_m}{a} \right) \mathbb{P}(f_t = i).$$

Proof. To calculate the expected loss per action a first recall the loss function introduced in Section 1.1

$$\ell_t(a) = \frac{C_m}{a} \mathbb{I}\{a < f_t\} + \frac{C_r}{f_t} \mathbb{I}\{a \geq f_t\}.$$

To determine the expected value of this function, Lemma 1 from Weiss et al., 2005 is introduced.

Lemma 1 (Tower Rule)**.** *Let $A_1, \ldots, A_N$ be a partition of the sample space, and let X be an integrable random variable. Then*

$$\mathbb{E}[X] = \sum_{i=1}^{N} \mathbb{E}[X \mid A_i] \cdot \mathbb{P}(A_i).$$

Therefore

$$\begin{aligned} \mathbb{E}[\ell_t(a)] &= \sum_{i=1}^{K} \mathbb{E}[\ell_t(a) | f_t = i] \mathbb{P}(f_t = i) && (1) \\ &= \sum_{i=1}^{a-1} \mathbb{E}[\ell_t(a) | f_t = i] \mathbb{P}(f_t = i) + \sum_{j=a}^{K} \mathbb{E}[\ell_t(a) | f_t = j] \mathbb{P}(f_t = j) && (2) \\ &= \sum_{i=1}^{a-1} \frac{C_r}{i} \mathbb{P}(f_t = i) + \sum_{j=a}^{K} \frac{C_m}{a} \mathbb{P}(f_t = j) && (3) \\ &= C_r \sum_{i=1}^{a-1} \frac{\mathbb{P}(f_t = i)}{i} + \frac{C_m}{a} \sum_{j=a}^{K} \mathbb{P}(f_t = j). && (4) \end{aligned}$$

Equality (1) is the implementation of the tower rule. In the next step the summation is split by action a. This is done because the loss is equal to C_m/a for $a < f_t$ and C_r/f_t for $a \geq f_t$, which is used in (3). In (4) the terms are rearranged so that the constants are in front of the summations.

In section 2.1 we found out that we have more unbiased feedback for small a than for large a (recall $n_t(a) = \sum_{s=1}^{t} \mathbb{I}\{a \leq a_t\}$). To use this observation the equation above will be rewritten using

$\sum_{i=1}^{K} \mathbb{P}(f_t = i) = 1$. Which is done below. Notice that for this step it seems that the assumption $f_t \in [K]$ is necessary, but this can be resolved by defining $\mathbb{P}(f_t = K) := \mathbb{P}(f_t \geq K)$.

$$\mathbb{E}[\ell_t(a)] = C_r \sum_{i=1}^{a-1} \frac{\mathbb{P}(f_t = i)}{i} + \frac{C_m}{a}\left(1 - \sum_{i=1}^{a-1} \mathbb{P}(f_t = i)\right) \tag{5}$$

$$= \sum_{i=1}^{a-1} \frac{C_r}{i}\mathbb{P}(f_t = i) + \frac{C_m}{a} - \sum_{i=1}^{a-1} \frac{C_m}{a}\mathbb{P}(f_t = i) \tag{6}$$

$$= \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\mathbb{P}(f_t = i). \tag{7}$$

In step (6) we get rid of the brackets, and in the final step the summations are put together. Notice that the second term is always bigger than zero since $\frac{C_r}{a-1} - \frac{C_m}{a} > \frac{C_r - C_m}{a} > 0$ because $C_r > C_m$. □

Notice that from Theorem 1 we know the expected loss of action a only depends on the constants C_r, C_m and the probability distribution of the failure time f_t.

With the formula above the expected loss can be calculated given the probability density function of f_t, the cost of repair and the cost of maintenance. Which is done in Figure 3.

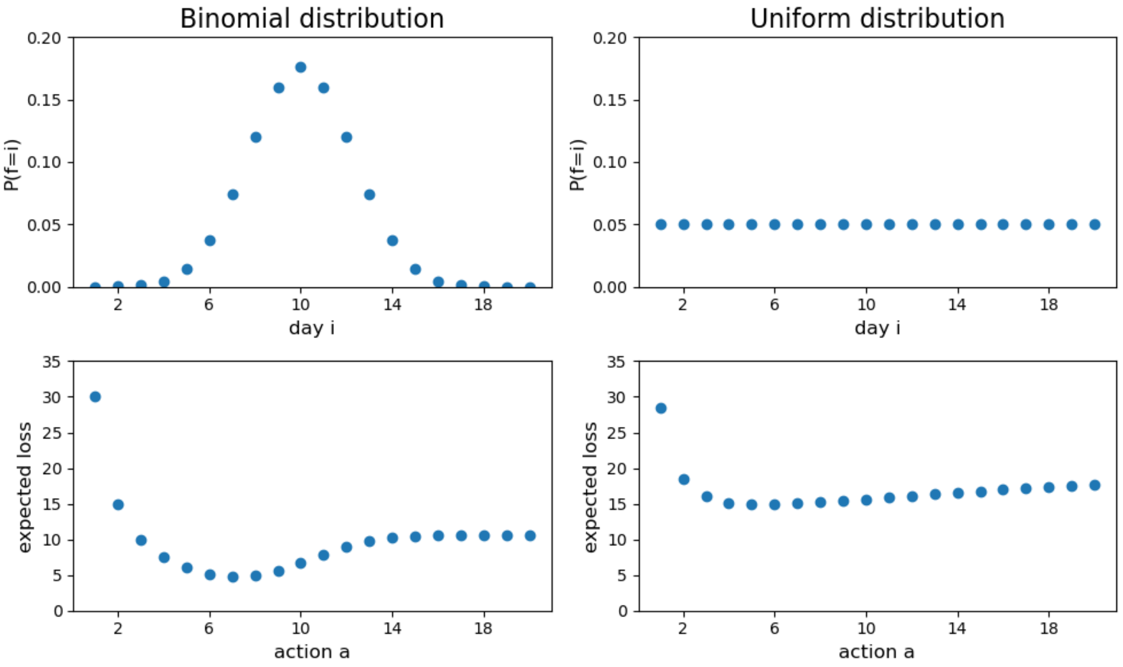

Figure 3: expected loss function for binomial and uniform distribution with parameters $C_r = 100$ and $C_m = 30$

In this figure the failure time is first given the binomial distribution, and then the uniform distribution. The two lower plots show the resulting expected loss functions. In these two lower plots we can see that for this parameters we would rather maintain too late ($a > a^*$) than too early ($a < a^*$). Notice that the optimal action a^* is the lowest point of this function. And we can see that the regret for actions $a > a^*$ is relatively small when the failure time has a uniform distribution, as the expected loss function increases slowly after its lowest point.

3 Algorithms

The algorithms will have to make a choice between exploring and exploiting: gaining knowledge about the failure time by letting the part break, or exploiting the best known option. Exploring is expensive, but exploiting has the risk that the best known option is not the best option. This chapter will cover three algorithms that all have a different strategy in terms of exploring and exploiting. The first has an explore and then exploit strategy, the second has an explore while exploiting strategy, and the last is an exploit-only strategy. The ideas behind the first two were introduced in Section 1.1: the Epsilon Greedy algorithm and the Upper Confidence Bound algorithm. The third algorithm was designed specifically for this problem, which is named the Empirical Bandit.

As seen in Section 2.3, the key challenge is estimating $\mathbb{P}(f = i) := p_i$. To estimate p_i the algorithms will use the empirical probability of a part breaking on day a. Which says that in round t the chance that the part breaks on day i is equal to the number of times the part has broken on day i, divided by the number of times we could have seen it break on day i. Denote by $b_t(i) := \sum_{s=1}^{t} \mathbb{I}\{f_s = i\}\mathbb{I}\{i \leq a_t\}$ the number of times the part has broken on day i. Here the second indicator function makes sure that the failure is actually observable. And we use $n_t(i)$ which was introduced in Section 2.1. Then the empirical probability is defined as

$$\hat{p}_{t,i} = \frac{b_t(i)}{n_t(i)} = \frac{\sum_{s=1}^{t} \mathbb{I}\{f_s = i\}\mathbb{I}\{i \leq a_t\}}{\sum_{s=1}^{t} \mathbb{I}\{i \leq a_t\}}.$$

Now define $\mu(a)$ the expected loss of action a based on p_i. And $\hat{\mu}_t(a)$ the estimated expected loss for action a based on the empirical probability $\hat{p}_t$ after t rounds. Using Theorem 1 gives

$$\mu(a) = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)p_i,$$

$$\hat{\mu}_t(a) = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_{t,i}.$$

3.1 Epsilon Greedy

This algorithm first explores only for the first T_0 rounds. By Section 2.1 we know that the only way to get unbiased feedback for every action is by choosing action K. Thus $a_t = K$ for $t \in [T_0]$. After this explore phase the algorithm starts to exploit the best known option. It does so by selecting the action that has the minimal loss based on the first T_0 observations. This strategy is described in Algorithm 1.

Algorithm 1 Epsilon Greedy

```
EG($K, T, C_r, C_m$)
for $t$ in range($T_0$) do                                        ▷ exploration phase
    $a_t = K$                                ▷ let the part break and therefore observe $f_t$
    $b(f_t) += 1$                         ▷ update number of times part has broken on day $f_t$
end for
$n(a) = T_0$ for $a \in [K]$                     ▷ for every action feedback is received $T_0$ times
$\hat{p}_i = b(i)/n(i)$                                       ▷ calculate empirical probability
$\hat{\mu}(a) = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_i$ for $a \in [K]$    ▷ calculate the expected loss for every action
for $t$ in range($T_0, T$) do                                     ▷ exploitation phase
    $a_t = \arg\min_{a \in [K]} \hat{\mu}(a)$                              ▷ choose best known action
end for
    return $a_1, \ldots, a_T$
```

The pseudo regret bound of the algorithm tells us how good the algorithm works for every given set of parameters, and is given in Theorem 2.

Theorem 2. *The pseudo regret of the Epsilon Greedy algorithm can be bounded by*

$$T_0 Cr + 2(T - T_0)\sum_{a:\Delta_a > 0}\Delta_a \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\exp\left(-\frac{T_0\Delta_a^2}{8}\right),$$

with probability at least $1 - \delta$*, where* $\delta = \exp(-\frac{T_0 \Delta_a^2}{8})$.

Proof. This proof follows the same structure of Luo, 2017. As mentioned in the introduction, the goal of this problem is to minimize the pseudo regret

$$R_T = \sum_{t=1}^{T} \mathbb{E}\left[\ell(a_t) - \ell(a^*)\right].$$

Define $\Delta_a = \mu(a) - \mu(a^*)$. Then the pseudo regret can be rewritten as

$$R_t = \sum_{t=1}^{T} \mathbb{E}\left[\Delta_{a_t}\right] = \sum_{t=1}^{T} \mathbb{E}\left[\sum_{a=1}^{K} \mathbb{I}\{a_t = a\}\Delta_a\right] = \sum_{t=1}^{T}\sum_{i=a}^{K} \mathbb{E}\left[\mathbb{I}\{a_t = a\}\right]\Delta_a = \sum_{a:\Delta_a>0} \Delta_a \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{a_t = a\}\right].$$

Since epsilon greedy explores for the first T_0 rounds against the cost $\sum_{t=1}^{T_0} \frac{C_r}{f_t} \leq T_0 C_r$, the pseudo regret is bounded by

$$R_t = T_0 Cr + \sum_{a:\Delta_a>0} \Delta_a \mathbb{E}\left[\sum_{t=T_0+1}^{T} \mathbb{I}\{a_t = a\}\right].$$

To find a bound for the expected value it is rewritten

$$\mathbb{E}\left[\sum_{t=T_0+1}^{T} \mathbb{I}\{a_t = a\}\right] = (T - T_0)\mathbb{P}\left(a_t = a\right).$$

Note that

$$\mathbb{P}(a_t = a) \leq \mathbb{P}\left(\hat{\mu}_{T_0}(a) \leq \hat{\mu}_{T_0}(a^*)\right),$$

since the algorithm chooses action $a_t = a$ if $a = \arg\min_{a\in[K]} \hat{\mu}_{T_0}(a)$, thus if $\hat{\mu}_{T_0}(a) \leq \hat{\mu}_{T_0}(i)$ for all $i \in [K]$ and therefore also for a^*. And notice that if $\hat{\mu}_{T_0}(a) \leq \hat{\mu}_{T_0}(a^*)$ than one of the two following rare events must happen

$$\hat{\mu}_{T_0}(a) \leq \mu(a) - \Delta_a/2,$$
$$\hat{\mu}_{T_0}(a^*) \geq \mu(a^*) + \Delta_a/2,$$

since otherwise $\hat{\mu}_{T_0}(a) > \mu(a) - \Delta_a/2 = \mu(a^*) + \Delta_a/2 > \hat{\mu}_{T_0}(a^*)$. This can also be seen in figure 4.

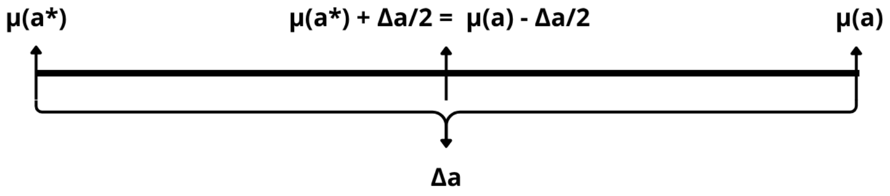

Figure 4: Visualization of the argument. If $\hat{\mu}(a)$ is on the left side of the center and $\hat{\mu}_{T_0}(a^*)$ is on the right side, then $\hat{\mu}_{T_0}(a) < \hat{\mu}_{T_0}(a^*)$. Also notice that $\mu(a^*) < \mu(a)$ for all actions $a \in [K]$ by definition of a^*.

Because both events imply that $|\mu(a) - \hat{\mu}_{T_0}(a)| > \Delta_a/2$, the probability that one of these events happens is equal to the following

$$2\mathbb{P}\left(|\mu(a) - \hat{\mu}_{T_0}(a)| > \Delta_a/2\right) = 2\mathbb{P}\left(\left|\frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)p_i - \frac{C_m}{a} - \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_{T_0,i}\right| > \Delta_a/2\right) \tag{8}$$

$$= 2\mathbb{P}\left(\left|\sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)p_i - \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_{T_0,i}\right| > \Delta_a/2\right) \tag{9}$$

$$= 2\mathbb{P}\left(\sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right) \cdot |p_i - \hat{p}_{T_0,i}| > \Delta_a/2\right) \tag{10}$$

$$= 2\sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\mathbb{P}\left(|p_i - \hat{p}_{T_0,i}| > \Delta_a/2\right). \tag{11}$$

Equality (8) uses the definition of $\mu(a)$, and the formula from Theorem 1. In the next step the terms C_m/a cancel each other out. Then the summations are put together, and in the last step the constants are taken out of the probability. To determine a bound for this probability, Lemma 2 from Hoeffding, 1963 is introduced.

Lemma 2 (Hoeffding's inequality)**.** *Let $X_1, \dots, X_T \in [-B, B]$ for some $B > 0$ be independent random variables such that $\mathbb{E}[X_t] = 0$ for all $t \in [T]$, then we have for all $\delta \in (0,1)$,*

$$\mathbb{P}\left(\sum_{t=1}^{T} X_t \geq B\sqrt{2T \ln 1/\delta}\right) \leq \delta.$$

By definition of $n_t(i)$ and the fact that the algorithm chooses $a_t = K$ for the first T_0 rounds we know that $n_{T_0}(i) = T_0$ for all $i \in [K]$. Therefore $\hat{p}_{T_0,i} = \frac{1}{T_0}\sum_{s=1}^{T_0} \mathbb{I}\{f_s = i\}$, so the properties of Lemma 2 are satisfied. Now using Lemma 2 and rewriting $T_0\Delta_a/2 = \sqrt{2T_0 \ln 1/\delta}$ to $\delta = \exp(-\frac{T_0\Delta_a^2}{8})$ gives the following bound

$$\mathbb{P}\left(|p_i - \hat{p}_{T_0,i}| > \Delta_a/2\right) \leq \exp\left(-\frac{T_0\Delta_a^2}{8}\right),$$

therefore

$$\mathbb{P}\left(|\mu(a) - \hat{\mu}_{T_0}(a)| > \Delta_a/2\right) \leq 2\sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\exp\left(-\frac{T_0\Delta_a^2}{8}\right).$$

Putting everything together then gives the bound for the pseudo regret of the Epsilon Greedy algorithm given in the theorem. □

The bound of Theorem 2 shows that the algorithm suffers from linear regret: $\mathcal{O}(T)$.

Now we will simulate the maintenance planning problem to see how the Epsilon Greedy algorithm (:EG) behaves. When T increases, relatively less observations are necessary to make a good guess of the optimal action. Therefore in the simulations below the exploration phase is of length $T_0 = \sqrt{T}$.

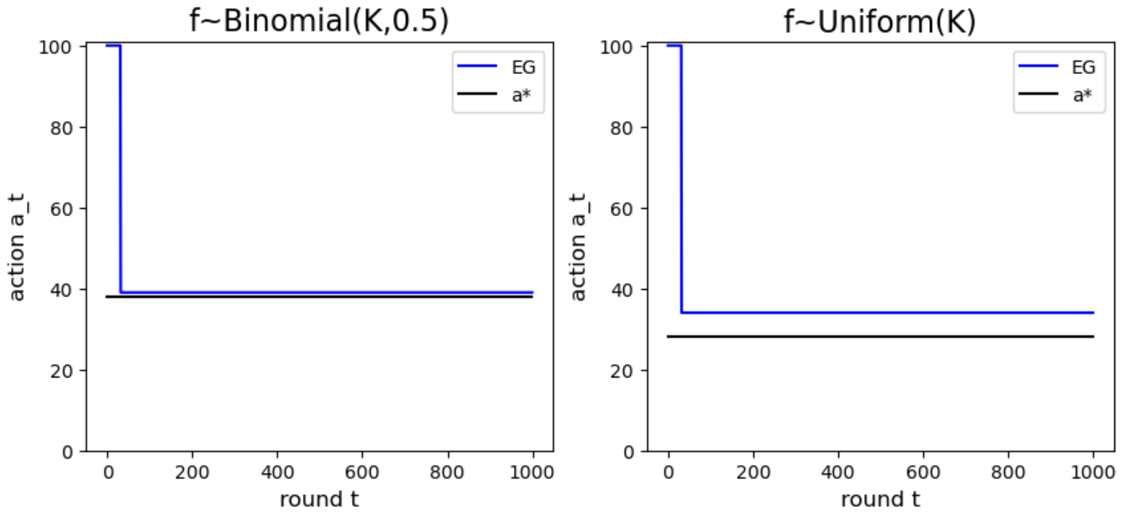

Figure 5: Behavior of EG with parameters $K = 100$, $T = 1000$, $C_b = 100$, $C_m = 30$ where different probability distributions are used for the failure time.

In Figure 5 the behavior of EG can be seen. First the part breaks $\sqrt{T}$ times and after that an action is chosen. In the left plot can be seen that an action is chosen that is close to the optimal action a^*. This means that after an expensive exploration phase, the regret will grow relatively slow (but linear). In the right plot the algorithm selected an action that is further from the optimal action. Therefore the regret will grow faster (also linear) over time. This is the flaw of this algorithm: after the exploration phase the algorithm stops learning. So if an action is chosen that is not the optimal action, EG will suffer linear regret.

3.2 Optimistic Bandit

Where the Epsilon Greedy algorithm had an explore phase and an exploit phase, the Optimistic Bandit algorithm exploits while exploring. It does so by not choosing the best know option, but by being optimistic about the random variable and therefore choosing what could be the best option. This strategy was briefly discussed in the introduction as Upper Confidence Bound. There the upper bound of the payout of the slot machines was taken because the gambler wants to maximize payout. In the maintenance strategy problem we need to take the lower bound of the empirical probabilities since we want to minimize costs. But first a confidence bound needs to be calculated. Therefore, Lemma 3 from Maurer and Pontil, 2009 is introduced.

Lemma 3 (data dependent Bennett's inequality)**.** *Let $Z_1, ..., Z_n$ be i.i.d. random variables with values in $[0,1]$ and let $\delta > 0$. Then with probability at least $1-\delta$ we have*

$$\mathbb{E}Z - \frac{1}{n}\sum_{i=1}^{n} Z_i \leq \sqrt{\frac{2V_n(Z)\ln(2/\delta)}{n}} + \frac{7\ln(2/\delta)}{3(n-1)},$$

where $V_n(Z) = \frac{1}{n(n-1)}\sum_{1\leq i<j\leq n}(Z_i - Z_j)^2$ is the sample variance.

In the maintenance strategy problem, the failure time is the random variable. Therefore this theorem allows us to bound the probability $\mathbb{P}(f_t = i) = p_i$. Thus with probability at least $1-\delta$ we have $p_i \in [\hat{p}_{t,i} - \beta_{t,i}, \hat{p}_{t,i} + \beta_{t,i}]$, where

$$\beta_{t,i} = \sqrt{\frac{2\hat{p}_{t,i}(1-\hat{p}_{t,i})\ln(2/\delta)}{n_t(i)}} + \frac{7\ln(2/\delta)}{3(n_t(i)-1)}.$$

Using this confidence bound, the strategy works as follows. The first two rounds the part is broken, and therefore the algorithm gets full feedback twice. After that the algorithm every round calculates the empirical probabilities and their confidence bounds. It then uses the lower bound of the probabilities to optimistically determine the expected loss for every action. After that the action with the minimum expected loss is chosen and $n_t(i)$ and $b_t(i)$ (introduced at the beginning of this chapter) are updated if necessary. This is also described in algorithm 2.

Algorithm 2 Optimistic Bandit

```
OB($K, T, C_r, C_m$)
$a_1 = K$, $b(f_1) = 1$, $n+ = 1$                                   ▷ let the part break and update $b$ and $n$
for $t$ in range$(2, T)$ do
    $\hat{p}_{t,i} = b(i)/n(i)$                                       ▷ empirical probability
    $\beta_{t,i} = \sqrt{\frac{2\hat{p}_{t,i}(1-\hat{p}_{t,i})\ln(2/\delta)}{n(i)}} + \frac{7\ln(2/\delta)}{3(n(i)-1)}$     ▷ calculate confidence bound
    $p^-_{t,i} = \max(\hat{p}_{t,i} - \beta_{t,i}, 0)$                  ▷ take the lower bound
    for $a$ in range$(K)$ do
        $\mu^-_t(a) = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)p^-_{t,i}$    ▷ calculate expected loss for each action
    end for
    $a_t = \arg\min_{a\in[K]} \mu^-_t(a)$                                ▷ choose action with least expected loss
    $n(i)+ = 1$ for $i \leq a_t$                                        ▷ update $n$
    if $f_t < a_t$ then                                                 ▷ the part breaks
        $b(f_t)+ = 1$                                                   ▷ update $f$
    end if
end for
    return $a_1, \ldots, a_T$
```

To determine how good the Optimistic Bandit algorithm works for all sets of parameters, we look at the bound of the pseudo regret of this algorithm, which is given by Theorem 3.

Theorem 3. *The pseudo regret of the Optimistic Bandit algorithm can be bounded by*

$$4\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) + 8KC_r\sqrt{\frac{2\ln^2(T^2)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6KC_r}}+\frac{3\Delta_{\min}}{4KC_r}}} + \frac{4KC_r}{3}\ln(T^2)\ln\left(\frac{\ln(T^2)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6KC_r}}+\frac{3\Delta_{\min}}{4KC_r}}\right),$$

with probability at least $1-\delta$ where $\delta = 1/T^2$.

Proof. In this proof the data dependent bound for the empirical probability of Lemma 3 is not used to simplify calculations. Instead we use the bound of Lemma 4, which is also introduced in Maurer and Pontil, 2009. The bound of Lemma 4 uses the true probabilities p_i instead of the estimations $\hat{p}_i$. The algorithm can not use this bound since the true probabilities are not known beforehand.

Lemma 4 (Bennett's inequality)**.** *Let $Z, Z_1, ..., Z_n$ be i.i.d. random variables with values in $[0,1]$ and let $\delta \in (0,1)$. Then with probability at least $1-\delta$ we have that*

$$\mathbb{E}Z - \frac{1}{n}\sum_{i=1}^{n} Z_i \leq \sqrt{\frac{2\mathbb{V}Z \ln 1/\delta}{n}} + \frac{\ln 1/\delta}{3n},$$

where $\mathbb{V}Z = \mathbb{E}(Z - \mathbb{E}Z)^2$ is the variance.

So now define

$$\beta_{t,i} = \sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n_t(i)}} + \frac{\ln(1/\delta)}{3n_t(i)}.$$

From Lemma 4 it then follows that after t rounds, p_i is bounded from below by $p^-_{t,i} = \hat{p}_{t,i} - \beta_{t,i}$ with high probability. Therefore, with high probability the expected loss of action a is bounded from below by

$$\mathbb{E}\left[\ell^-_t(a)\right] = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)p^-_{t,i}.$$

Denote by $\lambda_{t,i}$ the event that for all $i \in [K]$ we have that $|p_i - \hat{p}_{t,i}| > \beta_{t,i}$. This event will be used in the calculation of the bound of the pseudo regret, and therefore we want to bound the probability of this event happening. Notice that we can not use Lemma 4 directly as $n_t(i)$ is a random variable. The probability that the event $\lambda_{t,i}$ happens is bounded by

$$\begin{aligned}
\mathbb{P}\left(\lambda_{t,i}\right) &= \mathbb{P}\left(|p_i - \hat{p}_{t,i}| > \beta_{t,i}\right) &(12)\\
&= \mathbb{P}\left(|p_i - \hat{p}_{t,i}| > \sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n_t(i)}} + \frac{\ln(1/\delta)}{3n_t(i)}\right) &(13)\\
&\leq \mathbb{P}\left(\exists n \in [T] \text{ such that } \left|p_i - \frac{1}{n}\sum_{s=1}^{n}\mathbb{I}\{f_s = i\}\right| > \sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n}} + \frac{\ln(1/\delta)}{3n}\right) &(14)\\
&\leq \sum_{n=1}^{T}\mathbb{P}\left(\left|p_i - \frac{1}{n}\sum_{s=1}^{n}\mathbb{I}\{f_s = i\}\right| > \sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n}} + \frac{\ln(1/\delta)}{3n}\right) &(15)\\
&\leq \sum_{n=1}^{T}\delta = T\delta. &(16)
\end{aligned}$$

Equality (12) follows from the definition of $\lambda_{t,i}$ and the next equality follows from the definition of $\beta_{t,i}$. After that we use the definition of $\hat{p}_{t,i}$ to rewrite the term so that the probability can be bounded in a way that Lemma 4 can be applied. Inequality (15) uses that the probability that there is such a $n \in [T]$ is less or equal than the probability of the sum over all $n \in [T]$, and the last inequality uses Lemma 4.

To calculate a bound for the pseudo regret of the Optimistic Bandit, first the expected regret for action a_t is calculated as follows

$$\mathbb{E}\left[\ell_t(a_t) - \ell_t(a^*)\right] = \mathbb{E}\left[\ell_t(a_t)\right] - \mathbb{E}\left[\ell^-_t(a_t)\right] + \mathbb{E}\left[\ell^-_t(a_t)\right] - \mathbb{E}\left[\ell_t(a^*)\right].$$

To make calculations clearer, the first two terms will be bounded first and then the second two terms. The first two terms can be bounded as follows

$$
\begin{aligned}
\mathbb{E}\left[\ell_t(a_t)\right] - \mathbb{E}\left[\ell_t^-(a_t)\right] &= \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \mathbb{E}\left[p_i - p_{t,i}^-\right] && (17)\\
&= \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\} \cdot (p_i - p_{t,i}^-) + \mathbb{I}\{\lambda_{t,i}^c\} \cdot (p_i - p_{t,i}^-)\right] && (18)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \left(\mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\}\right] + \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}^c\} \cdot (p_i - \hat{p}_{t,i} + \beta_{i,t})\right]\right) && (19)\\
&= \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[p_i - \hat{p}_{t,i} + \beta_{i,t} | \lambda_{t,i}^c\right]\right) && (20)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[2\beta_{i,t}\right]\right) && (21)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \left(T\delta + 2\mathbb{E}\left[\beta_{t,i}\right]\right) && (22)\\
&= T\delta\mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] + 2\mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\beta_{t,i}\right] && (23)\\
&\leq T\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) + 2\mathbb{E}\left[\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i}\right]. && (24)
\end{aligned}
$$

Equality (17) comes from using the formula of the expected loss of Theorem 1. Equality (18) uses $1 = \mathbb{I}\{\lambda_{t,i}\} + \mathbb{I}\{\lambda_{t,i}^c\}$. Inequality (19) bounds $p_i - p_{t,i}^- \leq 1$ and uses the definition of the lower bound of the empirical probability $p_{t,i}^-$. Equality (20) comes from $\mathbb{E}[\mathbb{I}_A] = \mathbb{P}(A)$ and $\mathbb{E}[\mathbb{I}_A \cdot B] = \mathbb{P}(A)\mathbb{E}(B|A)$. Inequality (21) then uses the definition of the event $\lambda_{t,i}$, which means that if $\lambda_{t,i}^c$ then $p_i - \hat{p}_{t,i} \leq \beta_{t,i}$. (22) uses the earlier proven bound for the event $\lambda_{t,i}$ happening and bounds $\mathbb{P}(\lambda_{t,i}^c) \leq 1$. Equality (25) works out the brackets, and the last inequality uses that $a_t \leq K$ and $-1/a_t \leq -1/K$ to get rid of the expected value of the first term, and the summation $\sum^{a_t-1}$ is rewritten as $\sum^{K-1}\mathbb{I}\{i < a_t\}$ so that both summations sum up to $K-1$.

The second two terms of the pseudo regret can be bounded by

$$
\begin{aligned}
\mathbb{E}\left[\ell_t^-(a_t)\right] - \mathbb{E}\left[\ell_t(a^*)\right] &\leq \mathbb{E}\left[\ell_t^-(a^*)\right] - \mathbb{E}\left[\ell_t(a^*)\right] && (25)\\
&= \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \mathbb{E}\left[p_{t,i}^- - p_i\right] && (26)\\
&= \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\} \cdot (p_{t,i}^- - p_i) + \mathbb{I}\{\lambda_{t,i}^c\} \cdot (p_{t,i}^- - p_i)\right] && (27)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \left(\mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\}\right] + \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}^c\} \cdot (\hat{p}_{t,i} - p_i - \beta_{t,i})\right]\right) && (28)\\
&= \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[\hat{p}_{t,i} - p_i - \beta_{t,i} | \lambda_{t,i}^c\right]\right) && (29)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[\beta_{t,i} - \beta_{t,i}\right]\right) && (30)\\
&\leq \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot T\delta && (31)\\
&\leq T\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right). && (32)
\end{aligned}
$$

Inequality (25) uses that $\mathbb{E}[\ell_t^-(a_t)] \leq \mathbb{E}[\ell_t^-(a^*)]$ since $a_t = \arg\min_{a \in [K]} \mathbb{E}[\ell_t^-(a)]$. (26)-(30) then do exactly the same as (17)-(21), but notice that this time the $\beta_{t,i}$'s do not add up but cancel each other out. (31) uses the earlier proven bound for the event $\lambda_{t,i}$ happening, and the last inequality uses that $a_t \leq K$ and $-1/a_t \leq -1/K$.

Putting the above together gives the following pseudo regret bound for action a_t

$$2T\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + 2\mathbb{E}\left[\sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right]. \tag{33}$$

As the pseudo regret of the algorithm is calculated by adding the regret of all actions a_t together, we can find a bound for the pseudo regret by summing the bound above from $t = 1$ to $t = T$ which gives

$$\begin{aligned} R_t &= \mathbb{E}\left[\sum_{t=1}^{T} \ell_t(a_t) - \ell_t(a^*) \right] && (34) \\ &= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{a_t \neq a^*\} \cdot (\ell_t(a_t) - \ell_t(a^*)) \right] && (35) \\ &\leq 2T^2\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + 2\mathbb{E}\left[\sum_{t=1}^{T} \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right]. && (36) \end{aligned}$$

Where in the first step the indicator function is added because $l_t(a_t) - l_t(a^*) = 0$ when $a_t = a^*$. To calculate the second term, Lemma 5 from Gaillard et al., 2014 is introduced.

Lemma 5. *Let $a_0 > 0$ and $a_1, \dots, a_m \in [0, 1]$ and let $f : (0, +\infty) \to [0, +\infty)$ be a non increasing function. Then*

$$\sum_{j=1}^{m} a_i f(a_0 + a_{i-1}) \leq f(a_0) + \int_{a_0}^{\sum_{n=0}^{m} a_i} f(u) du.$$

Define the following function

$$g(x) = \sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{x}} + \frac{\ln(1/\delta)}{3x},$$

and notice that $g(n_t(i)) = \beta_{t,i}$. The integral of this function will be used in following calculations and is equal to

$$\int g(x)dx = 2\sqrt{2p_i(1-p_i)\ln(1/\delta)x} + \frac{1}{3}\ln(1/\delta)\ln(x) + C.$$

Now the second term of the pseudo regret can be bounded by

$$\begin{aligned} &2\mathbb{E}\left[\sum_{t=1}^{T} \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right] && (37) \\ &\leq 2C_r \sum_{i=1}^{K-1} \mathbb{E}\left[\sum_{t:n_t(i) < n^*(i)} \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} + \sum_{t:n_t(i) \geq n^*(i)} \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right] && (38) \\ &\leq 2C_r \sum_{i=1}^{K-1} \mathbb{E}\left[2\sqrt{2p_i(1-p_i)\ln(1/\delta)n^*(i)} + \frac{1}{3}\ln(1/\delta)\ln(n^*(i)) + \sum_{t:n_t(i) \geq n^*(i)} \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right]. && (39) \end{aligned}$$

In the first step the summation $\sum^T$ is split at the value $t : n_t(i) = n^*(i)$ and $\frac{C_r}{i} - \frac{C_m}{K}$ is bounded by C_r for all i. Inequality (39) uses the function g and Lemma 5 where $a_i = \mathbb{I}\{i < a_t\}$ and $\sum_{n=0}^{m} a_i = n^*(i)$.

Now recall $\Delta_a = \mathbb{E}[l_t(a) - l_t(a^*)]$ and $\Delta_{\min} = \min_{a \neq a^*} \Delta_a$. Then the pseudo regret is equal to the expected value of $\sum_{t=1}^{T} \Delta_{a_t}$ and therefore

$$\begin{aligned} R_t &= 2R_t - \mathbb{E}\left[\sum_{t=1}^{T} \Delta_{a_t} \right] && (40) \\ &\leq 2R_t - \mathbb{E}\left[\sum_{t=1}^{T} \Delta_{\min} \right] && (41) \\ &\leq 4T^2\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + 4C_r \sum_{i=1}^{K-1} \mathbb{E}\left[2\sqrt{2p_i(1-p_i)\ln(1/\delta)n^*(i)} + \frac{1}{3}\ln(1/\delta)\ln(n^*(i)) \right] && (42) \\ &+ 4C_r \sum_{i=1}^{K-1} \mathbb{E}\left[\sum_{t:n_t(i) \geq n^*(i)} \mathbb{I}\{a_t \neq a^*\} \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right] - \mathbb{E}\left[\sum_{t:n_t(i) \geq n^*(i)} \Delta_{\min} \right]. && (43) \end{aligned}$$

Equality (40) uses that $x = 2x - x$. After that $\Delta_a \geq \Delta_{\min}$ is used. The R_T in (41) is replaced for the last found bound for the pseudo regret in (42)-(43). (42) is an inequality because $\sum^T \Delta_{\min} \geq \sum_{t:n_t(i)\geq n^*(i)} \Delta_{\min}$. Now combine and bound the two terms of (43) by

$$4C_r \sum_{i=1}^{K} \mathbb{E}\left[\sum_{t:n_t(i)\geq n^*(i)} \left(\sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n_t(i)}} + \frac{\ln(1/\delta)}{3n_t(i)} - \frac{\Delta_{\min}}{4KC_r} \right) \right] \tag{44}$$

$$\leq 4C_r \sum_{i=1}^{K} \mathbb{E}\left[\sum_{t:n_t(i)\geq n^*(i)} \left(\sqrt{\frac{2\ln(1/\delta)}{n^*(i)}} + \frac{\ln(1/\delta)}{3n^*(i)} - \frac{\Delta_{\min}}{4KC_r} \right) \right], \tag{45}$$

where we bound $p_i \leq 1$ and use that $n_t(i) \geq n^*(i)$. To minimize this bound of the pseudo regret, we want to choose $n^*(i)$ such that

$$\sqrt{\frac{2\ln(1/\delta)}{n^*(i)}} + \frac{\ln(1/\delta)}{3n^*(i)} - \frac{\Delta_{\min}}{4KC_r} = 0.$$

Let $x = \sqrt{\frac{\ln(1/\delta)}{n^*(i)}}$, then we need to solve

$$\frac{1}{3}x^2 + \sqrt{2}x - \frac{\Delta_{\min}}{4KC_r} = 0,$$

thus use the abc-formula with $a = 1/3$, $b = \sqrt{2}$, $c = -\frac{\Delta_{\min}}{4KC_r}$ to find

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = -\frac{3}{2}\left(\sqrt{2} \mp \sqrt{2 + \frac{\Delta_{\min}}{3KC_r}} \right).$$

So now substitute x to get

$$\sqrt{\frac{\ln(1/\delta)}{n^*(i)}} = -\frac{3}{2}\left(\sqrt{2} + \sqrt{2 + \frac{\Delta_{\min}}{3KC_r}} \right).$$

Then square both sides and simplify the right term

$$\begin{aligned}
\frac{\ln(1/\delta)}{n^*(i)} &= \frac{9}{4}\left(2 + 2\sqrt{2}\sqrt{2 + \frac{\Delta_{\min}}{3KC_r}} + 2 + \frac{\Delta_{\min}}{3KC_r} \right) \\
&= \frac{9}{4}\left(4 + \frac{4}{\sqrt{2}}\sqrt{2 + \frac{\Delta_{\min}}{3KC_r}} + \frac{\Delta_{\min}}{3KC_r} \right) \\
&= 9\left(1 + \frac{1}{\sqrt{2}}\sqrt{2 + \frac{\Delta_{\min}}{3KC_r}} + \frac{\Delta_{\min}}{12KC_r} \right) \\
&= 9 + 9\sqrt{1 + \frac{\Delta_{\min}}{6KC_r}} + \frac{9\Delta_{\min}}{12KC_r} \\
&= 9 + 9\sqrt{1 + \frac{\Delta_{\min}}{6KC_r}} + \frac{3\Delta_{\min}}{4KC_r}.
\end{aligned}$$

And therefore choose

$$n^*(i) = \frac{\ln(1/\delta)}{9 + 9\sqrt{1 + \frac{\Delta_{\min}}{6KC_r}} + \frac{3\Delta_{\min}}{4KC_r}},$$

so that (45) is equal to zero. Notice that $K > 0$, $C_r > 0$ and $\Delta_{\min} > 0$ by definition, so we have no division by zero. Thus the pseudo regret is bounded by

$$
\begin{aligned}
&4T^2\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i}-\frac{C_m}{K}\right)+4C_r\sum_{i=1}^{K-1}\mathbb{E}\left[2\sqrt{2p_i(1-p_i)\ln(1/\delta)n^*(i)}+\frac{1}{3}\ln(1/\delta)\ln(n^*(i))\right] && (46)\\
&\leq 4T^2\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i}-\frac{C_m}{K}\right) && (47)\\
&+4C_r\sum_{i=1}^{K-1}\left(2\sqrt{2\ln(1/\delta)\frac{\ln(1/\delta)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6KC_r}}+\frac{3\Delta_{\min}}{4KC_r}}}+\frac{1}{3}\ln(1/\delta)\ln\left(\frac{\ln(1/\delta)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6KC_r}}+\frac{3\Delta_{\min}}{4KC_r}}\right)\right) && (48)\\
&\leq 4T^2\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i}-\frac{C_m}{K}\right) && (49)\\
&+8KC_r\sqrt{\frac{2\ln^2(1/\delta)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6K}}+\frac{3\Delta_{\min}}{4K}}}+\frac{4KC_r}{3}\ln(1/\delta)\ln\left(\frac{\ln(1/\delta)}{9+9\sqrt{1+\frac{\Delta_{\min}}{6K}}+\frac{3\Delta_{\min}}{4K}}\right). && (50)
\end{aligned}
$$

In the first inequality the chosen value for $n^*(i)$ is substituted and $p_i \leq 1$ is used. The second inequality uses that $K-1<K$. Substituting $\delta = 1/T^2$ then finalizes the proof.

□

Theorem 3 shows us that the pseudo regret of the Optimistic Bandit algorithm is $\mathcal{O}(\ln T^2)$. Now we will do four simulations of this algorithm to see how it behaves.

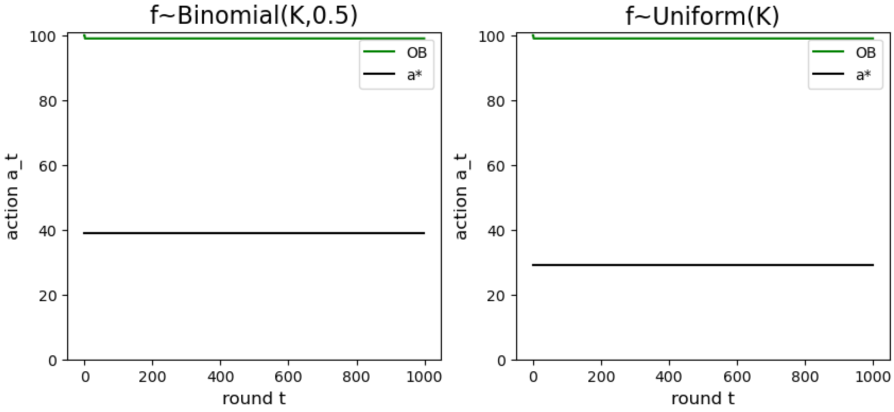

Figure 6: Behavior of OB with parameters $K = 100$, $T = 1000$, $C_r = 100$, $C_m = 30$ where different probability distributions are used for the failure time.

From the simulations of figure 6 can be seen that for those parameters the algorithm chooses $a_t = K-1$ nearly all the time, and that for this time span the Optimistic Bandit works pretty bad. This also shows that the algorithm takes a relatively long time to learn before it starts to make its guess for the optimal action.

Figures 7, 9 and 8 show that there are instances in which the algorithm functions better: if K gets smaller (Figure 7), if C_m/C_r gets smaller (Figure 8), or if T gets bigger (Figure 9). In Figure 7 can be seen that the Optimal Bandit indeed works better for smaller values of K, but that it still takes relatively long before the algorithm gets close to the optimal action. Figure 8 shows that the action chosen by the algorithm gets closer to the optimal action quicker if C_m/C_r is smaller (notice that $T = 5000$ in Figure 7 and $T = 1000$ in Figure 8). This seems to have a stronger effect in the case where the failure time has a uniform distribution. Figure 9 shows that also for the parameters of Figure 6 the algorithm seems to eventually choose the optimal action (notice that $T = 500000$).

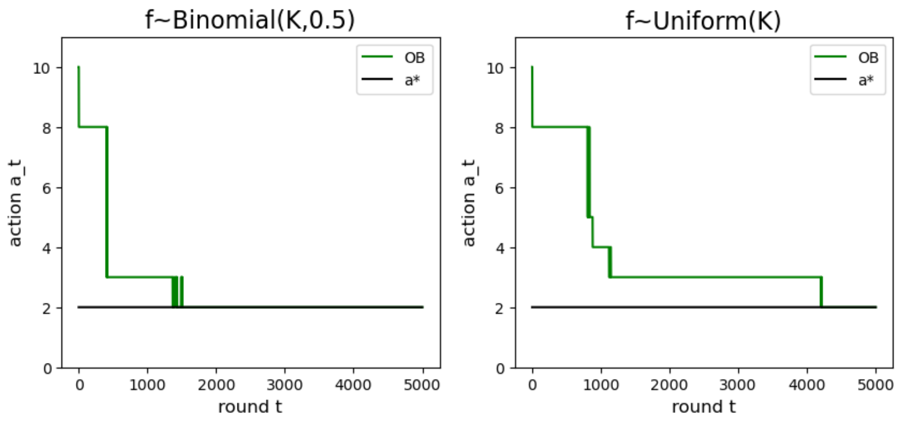

Figure 7: Behavior of OB with parameters $K = 10$, $T = 5000$, $C_r = 100$, $C_m = 30$ where different probability distributions are used for the failure time.

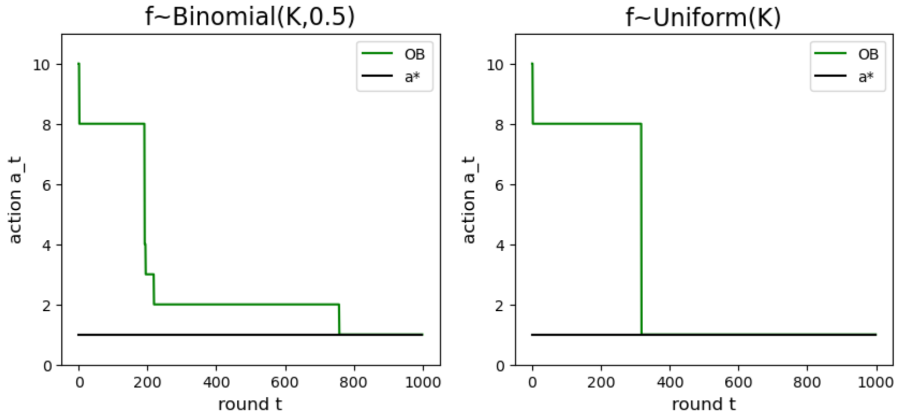

Figure 8: Behavior of OB with parameters $K = 10$, $T = 1000$, $C_r = 100$, $C_m = 1$ where different probability distributions are used for the failure time.

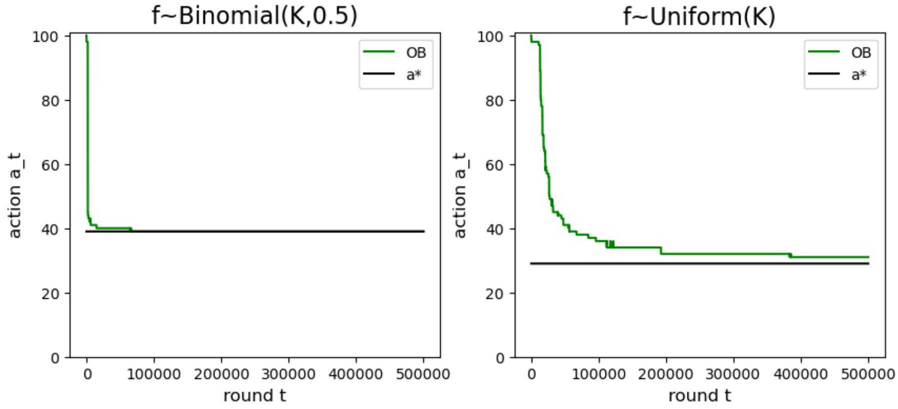

Figure 9: Behavior of OB with parameters $K = 100$, $T = 500000$, $C_r = 100$, $C_m = 30$ where different probability distributions are used for the failure time.

3.3 Empirical Bandit

This algorithm has an exploit only strategy: every round it chooses the best known option. But notice that by exploiting it also keeps learning every round since we get feedback for all days a for which $a \leq a_t$. As the name says this algorithm focuses purely on the empirical probability and not on any confidence bounds and it works as follows. The first round action $a_1 = K$ is chosen to receive full feedback and observe f_1. Then for the following rounds the empirical probability is used to calculate the expected loss for every action. The action that gives the smallest loss is then chosen. After that n is updated and if the part breaks b is also updated. This can also be seen algorithm 3

Algorithm 3 Empirical Bandit

```
**EB**($K, T, C_r, C_m$)
$a_1 = K$, $b(f_1) = 1$, $n(i) = 1$ for all $i \in [K]$        ▷ let the part break and update $b$ and $n$
**for** $t$ in range($2, T$) **do**
    $\hat{p}_{t,i} = b(i)/n(i)$        ▷ empirical probability
    **for** $a$ in range($K$) **do**
        $\hat{\mu}_t(a) = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_{t,i}$        ▷ calculate expected loss for each action
    **end for**
    $a_t = \arg\min_{a\in[K]} \hat{\mu}_t(a)$        ▷ choose action with least expected loss
    $n(i) += 1$ for $i \leq a_t$        ▷ update $n$
    **if** $f_t < a_t$ **then**        ▷ the part breaks
        $b(f_t) += 1$        ▷ update $f$
    **end if**
**end for**
    **return** $a_1, \dots, a_T$
```

Theorem 4. *The pseudo regret of the Empirical Bandit algorithm can be bounded by*

$$\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right)\cdot\left(1 + T\sqrt{2\ln(T^2)} + \frac{1}{3}T\ln(T^2)\right),$$

with probability at least $1 - \delta$ *where* $\delta = 1/T^2$.

Proof. In this proof we again use that $\mathbb{P}(\lambda_{t,i}) \leq T\delta$ from the proof of the bound for the Optimist Bandit (see (12)-(16)). First note that the expected loss of action a after t rounds is equal to

$$\mathbb{E}\left[\hat{\ell}_t(a)\right] = \frac{C_m}{a} + \sum_{i=1}^{a-1}\left(\frac{C_r}{i} - \frac{C_m}{a}\right)\hat{p}_{t,i}.$$

To calculate a bound for the pseudo regret of the Empirical Bandit, first the expected regret for action a_t is calculated as follows

$$\begin{align}
\mathbb{E}\left[\ell_t(a_t) - \ell_t(a^*)\right] &= \mathbb{E}\left[\ell_t(a_t)\right] - \mathbb{E}\left[\hat{\ell}_t(a_t)\right] + \mathbb{E}\left[\hat{\ell}_t(a_t)\right] - \mathbb{E}\left[\ell_t(a^*)\right] \tag{51}\\
&\leq \mathbb{E}\left[\ell_t(a_t)\right] - \mathbb{E}\left[\hat{\ell}_t(a_t)\right] + \mathbb{E}\left[\hat{\ell}_t(a^*)\right] - \mathbb{E}\left[\ell_t(a^*)\right] \tag{52}\\
&= \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right] \cdot \mathbb{E}\left[p_i - \hat{p}_{t,i}\right] + \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right] \cdot \mathbb{E}\left[\hat{p}_{t,i} - p_i\right] \tag{53}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \mathbb{E}\left[|p_i - \hat{p}_{t,i}|\right] \tag{54}\\
&= \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\} \cdot |p_i - \hat{p}_{t,i}| + \mathbb{I}\{\lambda_{t,i}^c\} \cdot |p_i - \hat{p}_{t,i}|\right] \tag{55}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(\mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\}\right] + \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}^c\} \cdot |p_i - \hat{p}_{t,i}|\right]\right) \tag{56}\\
&= \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[|p_i - \hat{p}_{t,i}||\lambda_{t,i}^c\right]\right) \tag{57}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right) \cdot \mathbb{E}\left[\beta_{i,t}\right]\right) \tag{58}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(T\delta + \mathbb{E}\left[\sqrt{\frac{2p_i(1-p_i)\ln(1/\delta)}{n_t(i)}} + \frac{\ln(1/\delta)}{3n_t(i)}\right]\right) \tag{59}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(T\delta + \mathbb{E}\left[\sqrt{2p_i(1-p_i)\ln(1/\delta)} + \frac{1}{3}\ln(1/\delta)\right]\right) \tag{60}\\
&\leq \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(T\delta + \sqrt{2\ln(1/\delta)} + \frac{1}{3}\ln(1/\delta)\right). \tag{61}
\end{align}$$

Equality (51) comes from adding and subtracting the same term. Inequality (52) uses that $\mathbb{E}[\hat{\ell}_t(a_t)] \leq \mathbb{E}[\hat{\ell}_t(a^*)]$ since $a_t = \arg\min_{a\in[K]} \mathbb{E}[\hat{\ell}_t(a)]$. After that we use Theorem 1. Inequality (54) is a rough bound: we use that either $\mathbb{E}[p_i - \hat{p}_{t,i}]$ or $\mathbb{E}[p_i - \hat{p}_{t,i}]$ is negative, so we bound the negative term with zero. We do not know if this is the right or left term, so therefore we bound the sum by $\sum^K$ since for both cases we have $\sum^{a_t-1} \leq \sum^K$ and $\sum^{a^*-1} \leq \sum^K$. Then we do the same as in (17)-(21). Inequality (59) then uses the bound $\mathbb{P}(\lambda_{t,i}) \leq T\delta$ and the definition of $\beta_{t,i}$. Inequality (60) is the next rough bound: we use that $n_t(i) \geq 1$. Notice that we can not use the same trick as in the proof of Theorem 3 because we do not have the indicator function $\mathbb{I}\{i \leq a_t\}$. In the last step we bound $p_i \leq 1$.

As the pseudo regret of the algorithm is calculated by adding the regret of all actions at together, we can find a bound for the pseudo regret by summing the bound above from $t = 1$ to $t = T$ which gives

$$\begin{align}
R_t &= \sum_{t=1}^{T} \mathbb{E}\left[\ell_t(a_t) - \ell_t(a^*)\right] \tag{62}\\
&\leq \sum_{t=1}^{T}\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(T\delta + \sqrt{2\ln(1/\delta)} + \frac{1}{3}\ln(1/\delta)\right) \tag{63}\\
&= \sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) \cdot \left(T^2\delta + T\sqrt{2\ln(1/\delta)} + \frac{1}{3}T\ln(1/\delta)\right). \tag{64}
\end{align}$$

Substitute $\delta = 1/T^2$ to finalize the proof. □

Theorem 4 shows that the pseudo regret of the Empirical Bandit algorithm is $\mathcal{O}(T\ln T^2)$. Notice that this is pretty bad and that here improvements can be made.

In Figure 10 the following behavior can be seen. The algorithm quickly chooses a_t near the optimal action, and over time it seems to get close to the optimal action.

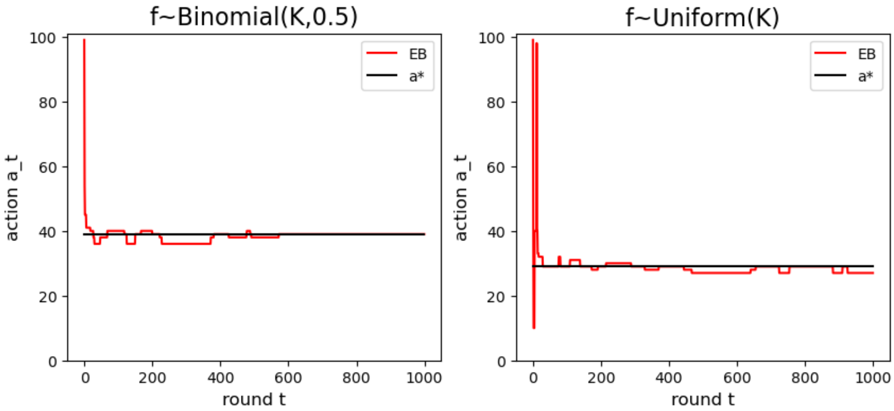

Figure 10: Behavior of EB with parameters $K = 100$, $T = 1000$, $C_b = 100$, $C_m = 30$ where different probability distributions are used for the failure time.

When looking at this figure we could ask ourselves why this 'exploit-only' algorithm seems to be exploring sometimes. To understand this behavior of the Empirical Bandit, observe the proof of the following claim.

Claim 1. *The 'exploit-only' algorithm the Empirical Bandit also explores when necessary.*

Proof. It is necessary to explore when the algorithm chooses to maintain before the optimal day of maintenance. In that case we would expect to repeatedly not see the part break. When the algorithm chooses a_t, we know that $\mathbb{E}[\hat{\ell}(a_t)] \leq \mathbb{E}[\hat{\ell}(a_t + 1)]$. Now use Theorem 1 to find the following inequality

$$\frac{C_m}{a_t} + \sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\hat{p}_{t,i} \leq \frac{C_m}{a_t+1} + \sum_{i=1}^{a_t}\left(\frac{C_r}{i} - \frac{C_m}{a_t+1}\right)\hat{p}_{t,i}.$$

Multiply the first term on the left with $\frac{a_t+1}{a_t+1}$ and the first term on the right with $\frac{a_t}{a_t}$. Also work out the brackets to get the following inequality

$$\frac{C_m(a_t+1)}{a_t(a_t+1)} + C_r\sum_{i=1}^{a_t-1}\frac{\hat{p}_{t,i}}{i} - \frac{C_m}{a_t}\sum_{i=1}^{a_t-1}\hat{p}_{t,i} \leq \frac{C_m a_t}{a_t(a_t+1)} + C_r\sum_{i=1}^{a_t}\frac{\hat{p}_{t,i}}{i} - \frac{C_m}{a_t+1}\sum_{i=1}^{a_t}\hat{p}_{t,i}.$$

Now we can subtract the first term on the right of both sides. Also subtract the second term on the left from both sides to get the following inequality

$$\frac{C_m}{a_t(a_t+1)} - \frac{C_m}{a_t}\sum_{i=1}^{a_t-1}\hat{p}_{t,i} \leq C_r\frac{\hat{p}_{t,a_t}}{a_t} - \frac{C_m}{a_t+1}\sum_{i=1}^{a_t}\hat{p}_{t,i}.$$

Then multiply the second term on the left with $\frac{a_t+1}{a_t+1}$ and take the last term out the summation on the right. That gives the following inequality

$$\frac{C_m}{a_t(a_t+1)} - \frac{C_m(a_t+1)}{a_t(a_t+1)}\sum_{i=1}^{a_t-1}\hat{p}_{t,i} \leq C_r\frac{\hat{p}_{t,a_t}}{a_t} - \frac{C_m a_t}{a_t(a_t+1)}\left(\sum_{i=1}^{a_t-1}\hat{p}_{t,i} + \hat{p}_{t,a_t}\right).$$

Now we can add $\frac{C_m a_t}{a_t(a_t+1)}\sum_{i=1}^{a_t-1}\hat{p}_{t,i}$ to both sides to get the following inequality

$$\frac{C_m}{a_t(a_t+1)} - \frac{C_m}{a_t(a_t+1)}\sum_{i=1}^{a_t-1}\hat{p}_{t,i}\left(a_t + 1 - a_t\right) \leq C_r\frac{\hat{p}_{t,a_t}}{a_t} - \frac{C_m a_t}{a_t(a_t+1)}\left(\hat{p}_{t,a_t}\right).$$

Clean up the terms and multiply the first term on the right with $\frac{a_t+1}{a_t+1}$ so that every term is divided by $a_t(a_t+1)$. This gives us the following inequality

$$\frac{C_m}{a_t(a_t+1)} - \frac{C_m}{a_t(a_t+1)}\sum_{i=1}^{a_t-1}\hat{p}_{t,i} \leq C_r\frac{\hat{p}_{t,a_t}(a_t+1)}{a_t(a_t+1)} - \frac{C_m a_t\hat{p}_{t,a_t}}{a_t(a_t+1)}.$$

And finally multiply every term with $a_t(a_t+1)$ and take C_m out of the brackets in the left and $\hat{p}_{t,a_t}$ out of the brackets on the right to get the following inequality that holds whenever a_t is chosen in round t

$$C_m\left(1-\sum_{i=1}^{a_t-1}\hat{p}_{t,i}\right)\leq\hat{p}_{t,a_t}\left(C_r(a_t+1)-C_m a_t\right).$$

Notice that if $\hat{p}_{t,a_t}=0$ then $\sum_{i=1}^{a_t-1}\hat{p}_{t,i}$ needs to be equal to 1 for the inequality to hold. Which is only possible if $f_1,\dots,f_t=1$ and $a_1,\dots,a_t=1$. And if this rare event happens there is no better option for the algorithm to plan maintenance as every action gives loss C_r. (Note that if you have a part that breaks every first day you should probably look for an other supplier for the parts of your machine.) In other words, the algorithm only chooses a_t if $\hat{p}_{t,a_t}\geq 0$, which means that we have seen the part break on day a_t at least once.

Now look at the case where it is necessary that the algorithm explores higher values of a. As stated earlier it is necessary to explore when we repeatedly have not seen the part. If action a_t is chosen and the part does not break ($a_t<f_t$) in round t then $n_{t+1}(i)=n_t(i)+1$ and $b_{t+1}(i)=b_t(i)$ for all $i\leq a_t$. This means that

$$\sum_{i=1}^{a_t-1}\hat{p}_{t+1,i}=\sum_{i=1}^{a_t-1}\frac{b_{t+1}(i)}{n_{t+1}(i)}=\sum_{i=1}^{a_t-1}\frac{b_t(i)}{n_t(i)+1}=\sum_{i=1}^{a_t-1}\frac{b_t(i)}{n_t(i)}\frac{n_t(i)}{n_t(i)+1}=\sum_{i=1}^{a_t-1}\hat{p}_{t,i}\cdot\frac{n_t(i)}{n_t(i)+1}.$$

Thus the sum on the left side of the inequality will shrink when we do not observe the part braking. This causes the left side of the inequality to grow. So if we repeatedly do not see the part break, sooner or later the inequality will not hold, so then $\mathbb{E}[\hat{\ell}_{t+1}(a_t)]>\mathbb{E}[\hat{\ell}_{t+1}(a_t+1)]$. As the algorithm chooses $a_{t+1}=\arg\min_{a\in[K]}\mathbb{E}[\hat{\ell}_{t+1}(a)]$, the algorithm will then choose a higher value a and therefore explore higher values of a.

Also notice that when C_m grows that the left side of the inequality grows, and the right side of the inequality shrinks. This means that when the cost of maintenance gets closer to the cost of repair, the algorithm will explore earlier. And when C_m gets smaller the opposite happens. So when the cost of maintenance gets smaller the algorithm is less likely to explore. This is exactly the behavior we would want to see as a rational person would take more risk if $C_m\to C_r$ and take less risk when $C_m<<C_r$. □

4 Simulations of the pseudo regret

In this chapter the algorithms of Chapter 3 are compared by simulations of the pseudo regret. The plots will also show the actions chosen by the algorithms since the actions are a good way to understand the behavior of the pseudo regret of the algorithms. The simulations are done for different sets of parameters to see in which cases which algorithm performs best. The figures below show the simulations and they all have the same structure: the left plots show the simulations where the failure time has the binomial distribution, and in the right plots the failure time has the uniform distribution. The top plots show the average actions chosen by the algorithms. The bottom plots show the average expected regret for the chosen actions. All plots show the average action/regret of 100 simulations.

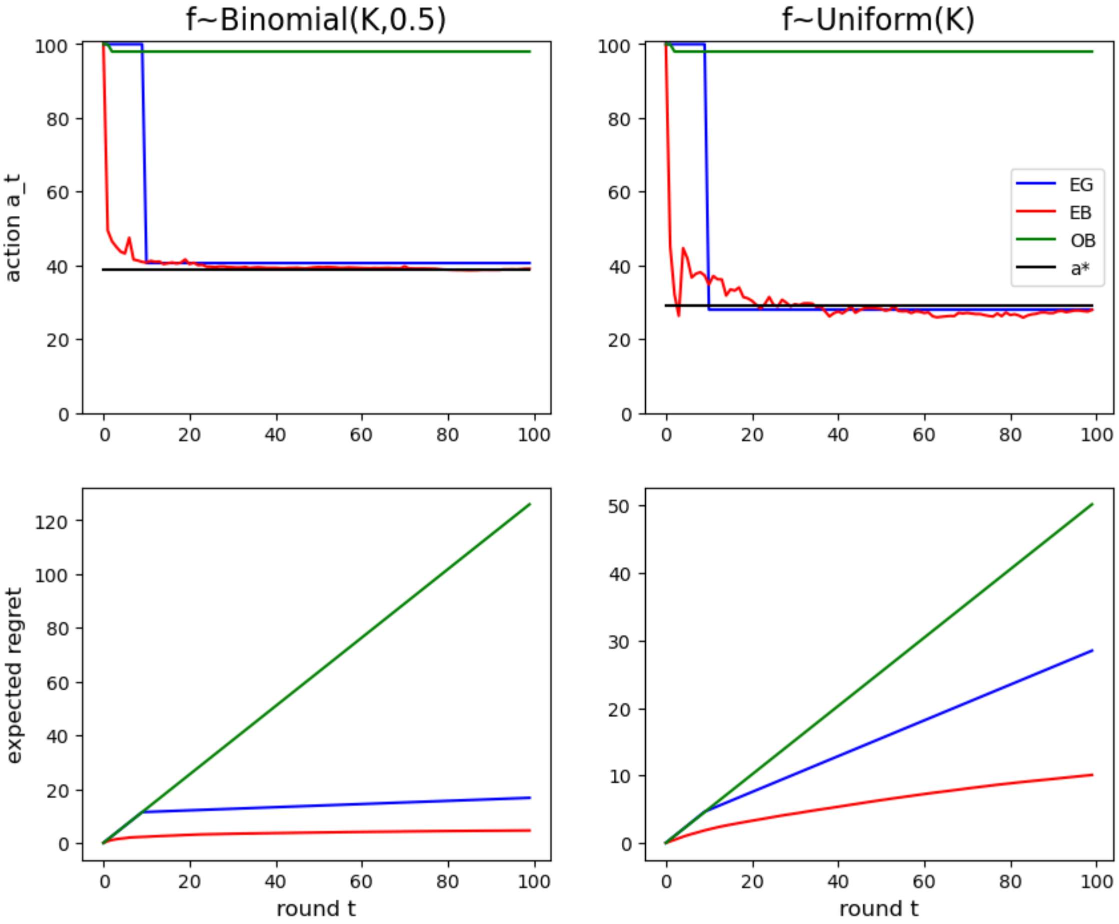

Figure 11: The average actions chosen by EG, EB and OB and their average expected regret over 100 simulations with parameters $K = 100,\ T = 100,\ C_b = 100,\ C_m = 30$ where different probability distributions are used for the failure time.

Figure 11 shows the short term performance of the algorithms. The first thing that stands out is that the Optimistic Bandit performs bad for these parameters. We can see that the Optimistic Bandit gets a lot of information by choosing $a_t = K-1$ but that it does not (yet) use this information to make a good guess for the optimal action. Also observe that Epsilon Greedy makes a relatively good choice when f_t has the binomial distribution, but that it has a harder time when f_t has the binomial distribution (in the lower left plot the blue line increases slower than in the lower right plot). In both cases the Empirical Bandit performs best as the pseudo regret is the smallest. Note that the choice a_t of the Empirical Bandit gets better over time, as we can see that in the lower plots the regret increases slower over time.

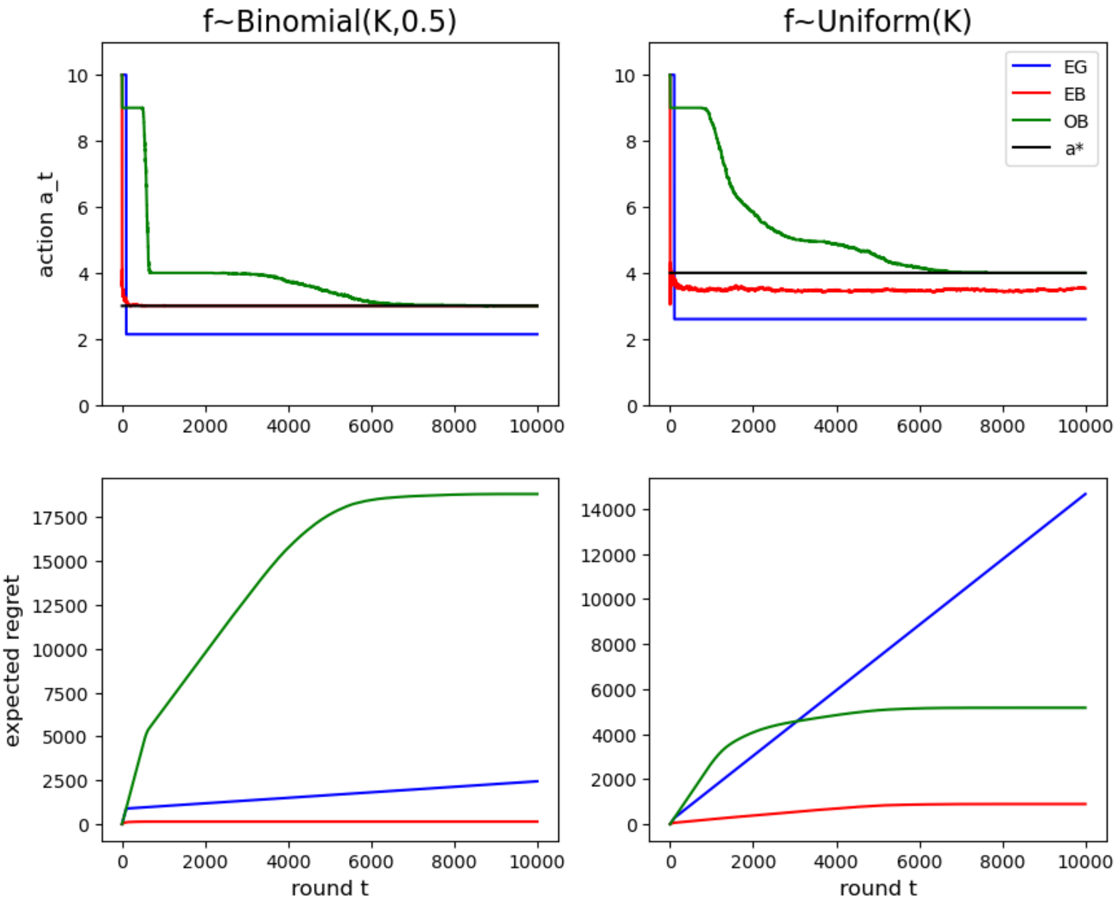

Figure 12: The average actions chosen by EG, EB and OB and their average expected regret over 100 simulations with parameters $K = 10,\ T = 10000,\ C_b = 100,\ C_m = 40$ where different probability distributions are used for the failure time.

In Figure 12 we can again see that Epsilon Greedy performs better when f_t has the binomial distribution, then when f_t has the uniform distribution. We can also see that when f_t has the binomial distribution, both the Optimistic Bandit and the Empirical Bandit find the optimal solution after a certain period of time, but that the Empirical Bandit needs less time and therefore has least regret.

Interesting to see in the right plots is that the Empirical Bandit does not choose the optimal solution every time, but that the regret nevertheless appears minimal. The average action chosen by the Empirical Bandit is around 3.5, so approximately half of the time the algorithm chooses $a_t = 4$ and half of the time $a_t = 3$ over time. Since the regret of the Empirical Bandit does not seem to grow after $T = 5000$ and since $a_t = 4$ is the optimal action, this should mean that the regret of $a_t = 3$ is close to zero for these parameters. So let us calculate the regret of $a = 3$ to see if this is really the case. Use Theorem 1 and $K = 10, C_m = 40, C_r = 100$ to find

$$\begin{aligned}\mathbb{E}[\ell(3)] &= \frac{40}{3} + \left((100 - \frac{40}{3}) + (50 - \frac{40}{3})\right)\frac{1}{10} \\ &= \frac{40}{3} + \left(\frac{260}{3} + \frac{110}{3}\right)\frac{1}{10} \\ &= \frac{40}{3} + \frac{37}{3} = \frac{77}{3}\end{aligned}$$

$$\begin{aligned}\mathbb{E}[\ell(4)] &= 10 + \left((100 - 10) + (50 - 10) + (\frac{100}{3} - 10)\right)\frac{1}{10} \\ &= 10 + \left(90 + 40 + \frac{70}{3}\right)\frac{1}{10} \\ &= \frac{30}{3} + \frac{1}{3}(27 + 12 + 7) = \frac{76}{3}.\end{aligned}$$

Therefore the regret of action $a = 3$ is indeed relatively small:1/3, and looking at the values on the y-axis one can understand that this increase of regret is not noticeable in this plot.

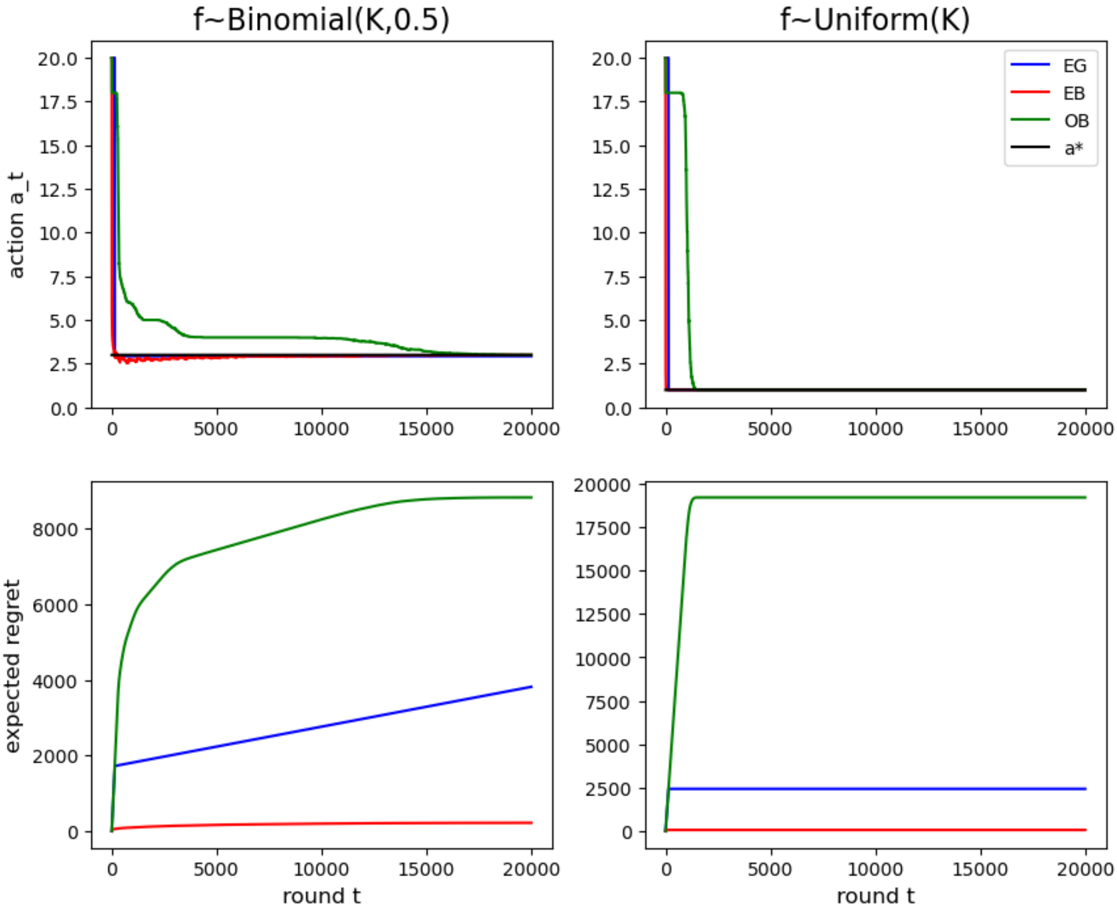

Figure 13: The average actions chosen by EG, EB and OB and their average expected regret over 100 simulations with parameters $K = 20,\ T = 20000,\ C_b = 100,\ C_m = 1$ where different probability distributions are used for the failure time.

Figure 13 shows the case where it is relatively cheap to do maintenance. From the plots of the pseudo regret we can see that Epsilon Greedy has a decent guess of the optimal action when f_t has the binomial distribution, and that it chooses the optimal action every time when f_t has the uniform distribution. We can also see that both the Optimistic Bandit and the Empirical bandit find the optimal solution after a certain amount of time, but that the Empirical Bandit needs less time and therefore has least pseudo regret.

Notice the behavior of the Optimistic Bandit in the top left plot: the algorithm appears to explore in steps. This observation will be used in the following page.

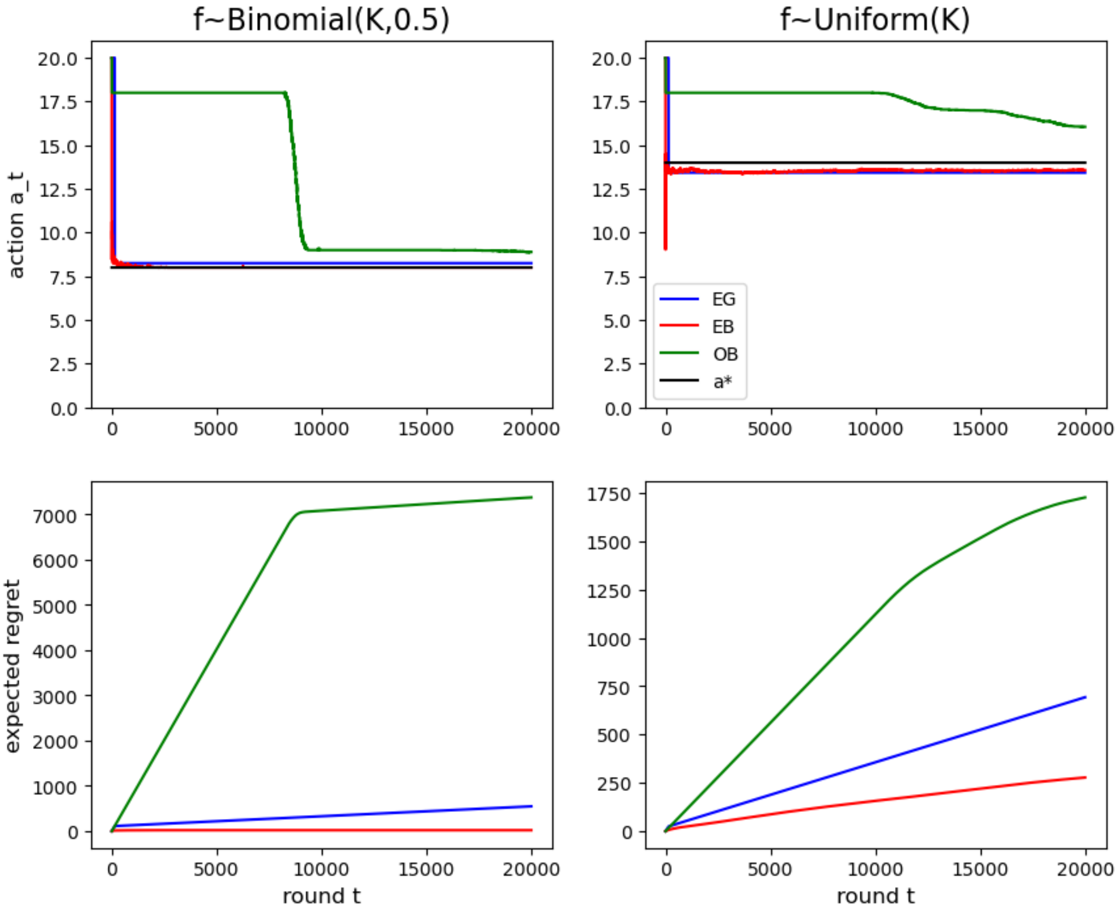

Figure 14: The average actions chosen by EG, EB and OB and their average expected regret over 100 simulations with parameters $K = 20,\ T = 20000,\ C_b = 100,\ C_m = 75$ where different probability distributions are used for the failure time.

Figure 14 shows the case where it is relatively expensive to do maintenance. We can see that the Optimistic Bandit needs a lot of time to learn in both cases. But because of the bound of Theorem 3 and the behavior of OB in the top left plot of Figure 13 we would expect that this algorithm will find the optimal action over time.

In the simulations where f_t has the binomial distribution the pseudo regret of the Empirical Bandit appears to be optimal, and in the top left plot we also see that the algorithm finds the optimal action relatively quick. When f_t has the uniform distribution the Empirical Bandit also performs best, but it is not clear if it will find the optimal action over time.

5 Conclusion

We start with the conclusions that can be made about the maintenance planning problem. Section 2.1 showed us that not all information gained from the rounds before is useful: in the case where the part breaks before maintenance is planned we can calculate what would have been the loss for every action. But because $f_t = a$ is never observed for $a > a_t$, the observations for $a > a_t$ in the case of $f_t \leq a_t$ are biased, since it tells us that we have not seen the part break on day a while we did not have the opportunity to actually see it break. Therefore, if this information is used to estimate the probability that the part breaks on day a, this probability is underestimated for large a. From this follows that selecting $a_t = K$ is the only way to receive unbiased feedback for every action. From Section 2.2 we learned that the best day to plan maintenance is the day before the part breaks. And that if the part breaks, we would rather be far to late with planning maintenance than just to late, as we get more feedback when maintenance is planned far to late. Section 2.3 showed us that the expected loss for every action can be calculated in a way that makes use of the feedback model, and that the expected loss depends on the probability distribution of the failure time. So from Chapter 2 we conclude that the challenge of the maintenance planning problem is to estimate the probability distribution of the failure time. Which means that the algorithms need to make a choice of letting the part break to learn more about the failure time, or to try to guess the optimal action to save the repair cost.

Now we move on to the conclusions about the performance of the three algorithms that where designed to solve the maintenance planning problem. We start with the Epsilon Greedy algorithm. The bound of Theorem 2 proved that the algorithm suffers from linear pseudo regret. In the simulations we saw that the regret is indeed linear, but that it can take the Optimistic quite some time before it outperforms Epsilon Greedy. When f_t has the uniform distribution we see that most of the time the Epsilon Greedy algorithm chooses an action $a_t < a^*$, and from Figure 3 we know that that gets punished with a bigger expected loss, and therefore the pseudo regret grows harder in this case.

Secondly we observe the Optimistic Bandit algorithm. Theorem 3 gave us a promising pseudo regret bound, but in the simulations we saw the flaw of this algorithm: it indeed finds the optimal action over time, but it needs a relatively long time to do so. We observed in Section 3.2 that the Optimistic Bandit algorithm needs more time to explore when K is big, or when it is relatively expensive to do maintenance (C_m/C_b is big). Notice that when it is relatively cheap to do maintenance, it is relatively expensive to explore. Therefore there is an incentive to exploit earlier.

And last but not least we observe the Empirical Bandit algorithm. The bound that was found in Theorem 4 was by far the worst, but this algorithm has performed best for all simulations. The reason that this algorithm works so well might be because of the feedback model and Theorem 1: when action a_t is chosen we receive feedback for days $i \leq a_t$, and when we want to calculate the expected loss for a_t we do not need the observations for all days but only for days $i \leq a_t$. In the simulations can be seen that the actions chosen by the Empirical Bandit algorithm get close to the optimal action impressively quick, and then over time stay close to the optimal action. The simulations however do not imply that the algorithm will always find the optimal action when T gets bigger.

To conclude this thesis; let the Empirical Bandit tell you when to maintain your machines. Since this algorithm seems to work best for most sets of parameters.

6 Discussion

Below are suggestions for further analysis on the three algorithms that were designed to solve the maintenance planning problem.

To improve the Epsilon Greedy algorithm, further analysis could be done on how to choose T_0. For short term use $T_0 = \sqrt{T}$ seems to work well, but it is not proven why. For long term use we need to choose T_0 such that we can say with high probability after T_0 rounds we know the optimal solution. Then with high probability the regret will not grow after the exploration phase.

As concluded in the previous chapter, the Optimistic Bandit is good at finding the optimal action, but the algorithm needs a long time to do so. Therefore we could try to modify the algorithm in a way such that it starts to exploit earlier. Notice that the Epsilon Greedy algorithm is given a certain time to explore, while the Optimistic Bandit algorithm decides by itself when it is time to exploit. So we could try to use the same principle of Epsilon greedy to let the algorithm exploit earlier. However we should be careful with the modifications to make sure that the bound of Theorem 3 still holds.

For the Empirical Bandit algorithm, further analysis could be done to improve the bound of the pseudo regret. To improve this bound we could start at inequality (54). Instead of this rough estimation we could do the following

$$\begin{aligned}
\mathbb{E}\left[\ell_t(a_t) - \ell_t(a^*)\right] &\le \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right]\cdot\mathbb{E}\left[p_i - \hat{p}_{t,i}\right] + \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right]\cdot\mathbb{E}\left[\hat{p}_{t,i} - p_i\right] \quad (65)\\
&= \mathbb{E}\left[\sum_{i=1}^{a_t-1}\left(\frac{C_r}{i} - \frac{C_m}{a_t}\right)\right]\cdot\mathbb{E}\left[p_i - \hat{p}_{t,i}\right] - \mathbb{E}\left[\sum_{i=1}^{a^*-1}\left(\frac{C_r}{i} - \frac{C_m}{a^*}\right)\right]\cdot\mathbb{E}\left[p_i - \hat{p}_{t,i}\right] \quad (66)\\
&\le \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\mathbb{E}\left[p_i - \hat{p}_{t,i}\right] \quad (67)\\
&= \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\}\cdot(p_i - \hat{p}_{t,i}) + \mathbb{I}\{\lambda_{t,i}^c\}\cdot(p_i - \hat{p}_{t,i})\right] \quad (68)\\
&\le \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\left(\mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}\}\right] + \mathbb{E}\left[\mathbb{I}\{\lambda_{t,i}^c\}\cdot(p_i - \hat{p}_{t,i})\right]\right) \quad (69)\\
&= \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right)\cdot\mathbb{E}\left[p_i - \hat{p}_{t,i}\lambda_{t,i}^c\right]\right) \quad (70)\\
&\le \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\left(\mathbb{P}\left(\lambda_{t,i}\right) + \mathbb{P}\left(\lambda_{t,i}^c\right)\cdot\mathbb{E}\left[\beta_{i,t}\right]\right) \quad (71)\\
&\le \mathbb{E}\left[\sum_{i=\min(a_t,a^*)}^{\max(a_t-1,a^*-1)}\left(\frac{C_r}{i} - \frac{C_m}{\max(a_t,a^*)}\right)\right]\cdot\left(T\delta + \mathbb{E}\left[\beta_{i,t}\right]\right) \quad (72)\\
&\le T\delta\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right) + \mathbb{E}\left[\sum_{i=1}^{K-1}\left(\frac{C_r}{i} - \frac{C_m}{K}\right)\cdot\mathbb{I}\{i < \max(a_t,a^*)\}\cdot\beta_{t,i}\right]. \quad (73)
\end{aligned}$$

Equality (66) uses that $\mathbb{E}[\hat{p}_{t,i} - p_i] = -\mathbb{E}[p_i - \hat{p}_{t,i}]$. The next step then bounds $-C_m/a_t$ and $-C_m/a^*$ by $-C_m/\max a_t, a^*$ before the summations are subtracted from each other. Then (68)-(72) use the event $\lambda_{t,i} = \{|p_i - \hat{p}_{t,i}| > \beta_{t,i}$ and the bound $\mathbb{P}(\lambda_{t,i}) \le T\delta$ which were also used in the proof of Theorem 3. The last inequality first works out the brackets and bounds the first term with the worst case: $a_t = 1$ and $a^* = K$ or the other way around. The second term is bounded as follows $\sum_{\min(a_t,a^*)}^{\max a_t-1,a^*-1} \le \sum_1^{\max a_t,a^*} = \sum_1^{K-1}\mathbb{I}\{\max(a_t,a^*)\}$.

If $a_t > a^*$ for all t, then the indicator function is equal to $\mathbb{I}\{i < a_t\}$. Then recognize that the bound above is equal to half of the pseudo regret bound for action a_t of (33). From this follows that when $a_t > a^*$ for all t, then the bound of the pseudo regret of the Empirical Bandit is exactly half of the bound found for the Optimistic Bandit of Theorem 3.

If $a_t > a^*$ does not hold for all t, we would need to find another way. Therefore look at the second rough bound of inequality (60). Here we used the worst case $n_t(i) = 1$ for all t: the algorithm chooses $a_t = 1$ all the time. But Claim 1 shows that this can not happen. The claim shows that the algorithm does explore and that therefore $n_t(i)$ will get bigger over time for $i \leq a^*$. To use this in the pseudo regret bound of the algorithm we would need to know how fast $n_t(i)$ grows for $i \leq a^*$.

If $a_t \leq a^*$ holds for all t, the pseudo regret can be bounded by

$$T\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + \mathbb{E}\left[\sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{i < \max(a_t, a^*)\} \cdot \beta_{t,i} \right] \tag{74}$$

$$= T\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + \mathbb{E}\left[\sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{i < a^*\} \cdot \beta_{t,i} \right] \tag{75}$$

$$= T\delta \sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) + \mathbb{E}\left[\sum_{i=1}^{K-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \mathbb{I}\{i < a_t\} \cdot \beta_{t,i} \right] + \mathbb{E}\left[\sum_{i=a_t}^{a^*-1} \left(\frac{C_r}{i} - \frac{C_m}{K} \right) \cdot \beta_{t,i} \right] \tag{76}$$

The second term can be bounded in a same way as is done in the proof of Theorem 3. Then the challenge remains to bound the last term.

References

Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, *47*(2-3), 235–256.

Gaillard, P., Stoltz, G., & van Erven, T. (2014). A second-order bound with excess losses. *Proceedings of the 27th Annual Conference on Learning Theory (COLT)*, *35*, 1–24. http://proceedings.mlr.press/v35/gaillard14.pdf

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, *58*(301), 13–30.

Luo, H. (2017). Lecture 14: Multi-armed bandits ii [CSCI 699 Lecture Notes, University of Southern California].

Maurer, A., & Pontil, M. (2009). Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*.

Weiss, N., Holmes, P., & Hardy, M. (2005). *A course in probability*. Pearson Addison Wesley. https://books.google.nl/books?id=p-rwJAAACAAJ