DDP PHASE-3: Multi-Armed Sticky Bandits

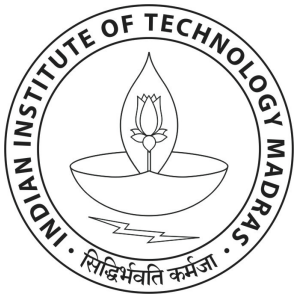
A Project Report

submitted by

A JAYAKRISHNAN

in partial fulfilment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY



DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

JANUARY-JUNE 2020

THESIS CERTIFICATE

This is to certify that the thesis titled Multi-Armed Sticky Bandits, submitted by **A JAYAKRISHNAN**, to the Indian Institute of Technology, Madras, for the award of the degree of **B.Tech & M.Tech**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Avhishek Chatterjee
Research Guide
Assistant Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: June 12, 2020

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and gratitude to Dr. Avhishek Chatterjee for guiding me with ideas, thoughts and insights for conducting research in this topic of Multi-Armed Sticky Bandits.

I express my sincere thanks to my faculty advisor Dr. Arun Pachai Kannu for his advices and help.

My sincere thanks are due to Prof. Ravinder David Koilpillai, Head of the Department Electrical Engineering and other Professors in the department for providing with teaching, facilities and other services.

ABSTRACT

KEYWORDS: Sticky bandit ; Regret upper bound; Reward; UCB.

In this project we have proposed UCB like algorithms for two cases of sticky bandit problem - with unknown and known delays (Y_k), referred to as case-1 and case-2, respectively. We obtained a sub-linear regret upper bound for the case-1 as proportional to $\sqrt{n}$, where n is run-time. For case-2 we have partially developed a UCB like algorithm, which we expect to outperform the algorithm of case-1.

TABLE OF CONTENTS

ABBREVIATIONS

MAB	Muti-Armed Bandit
UCB	Upper Confidence Bound

NOTATION

κ	Number of arms.
n	Run-time.
$T(t, k)$	Number of times k^{th} arm is pulled at time t.
$X_{k,j}$	Reward in pulling k^{th} arm as j^{th} pull.
$Y_{k,j}$	Delay in pulling k^{th} arm as j^{th} pull.
x_k	Expected value of X_k.
y_k	Expected value of Y_k.
$T_k(i)$	Number of times arm k is pulled after i^{th} pull.
$\hat{x}_k(i)$	Empirical value of x_k calculated after i^{th} pull.
$\hat{y}_k(i)$	Empirical value of y_k calculated after i^{th} pull .
$\widetilde{n}$	Upper bound for number of pulls.
R_n	Regret,
y_l	Smallest expected delay among κ arms.
y_m	Largest expected delay among κ arms.
$P_k(.)$	Probability distribution of Y_k.

CHAPTER 1

INTRODUCTION

Multi Armed Bandit(MAB) or $\kappa-$ armed bandit problem(Auer *et al.* (2002)), consists of κ alternative levers, each having a stochastic reward with an initial unknown probability distribution. A decision maker tries to maximize the sum of rewards earned through a sequence of lever pulls. By picking the arm with maximum expected reward, we can maximize the expected value of sum of rewards. Sampling an arm more number of times (exploration) leads to better the estimate of expected reward. This helps us find the optimal arm. Later committing to this arm (exploitation) leads to improving the expected total reward.

The regret R_n after n rounds is defined as the expected difference between the reward sum associated with an optimal strategy and the sum of the collected rewards.

$$R_n = n\mu_* - \mathbb{E}\left(\sum_{i=1}^{n} X_i\right) \tag{1.1}$$

Where X_i is the reward obtained in the i^{th} round. And μ_* is the expected reward of the optimal strategy. This expression can be reformulated as

$$R_n = \sum_{k=1}^{\kappa} \mathbb{E}\left[T_k(n)\right] \Delta_k \tag{1.2}$$

where $T_k(n)$ is number of times arm k is pulled in n rounds.
And $\Delta_k = \mu_* - \mu_k$ denotes the gap between the expected rewards of the optimal arm and of arm k.
An algorithm with less regret is expected to give a larger reward.
In a sticky bandits, picking an arm results in you being stuck with that arm for a period of time. This wait period is stochastic with different distributions for each arm. We introduce three different cases of the sticky bandit problem and discuss its convergence for a Upper Confidence Bound (UCB) like algorithm. The UCB algorithm for standard κ-armed bandits is given in APPENDIX A.

CHAPTER 2

PREVIOUS WORKS

Auer *et al.* (2002) have shown simple and efficient policies exhibiting uniform logarithmic regret for the bandit problem. The policies are deterministic and based on upper confidence bounds. A result from this is crucial for deriving a sub-linear regret upper bound.

Prashanth.L.A (2018) has shown a regret upper bound for UCB algorithm on MAB problem. We tried to extend this to the sticky bandit problem and derived a regret upper bound.

Jun and Nowak (2016) have introduced an anytime exploration for multi-armed bandits using confidence information to make a prediction of the top $\mathfrak{m}-$arms at every time step. They proposed an Any Time Lower and Upper Confidence Bound (AT-LUCB) algrithm, which is a nontrivial algorithm that provably solves anytime Explore-m.

Carrascosa and Bellalta (2019) have carried out a decentralized Access Point(AP) selection using multi-armed bandits using a novel approach called opportunistic ϵ-greedy with stickiness. This halts the exploration when a suitable AP is found, then, it remains associated to it while the user station(STA) is satisfied, only resuming the exploration after several unsatisfactory association periods.

Mannor (2011) has briefly reviewed some of the most popular bandit variants such as Bayesian, Markovian, adversarial, budgeted, and exploratory.

Lai and Robbins (1985) have constructed asymptotically efficient adaptive allocation rules for multi arm bandit problem. They have also shown a lower bound for the expected sample size form an inferior population.

CHAPTER 3

STICKY BANDIT-GENERAL MODEL

We consider a κ armed bandit with delayed reward. Another key difference from the regular bandit problem is that, instead of a fixed number of pulls, here we have a fixed run-time equals "n". The problem formulation is given as follows.

If arm k is pulled for the $T(t,k)$th pull at time t.

- the reward comes at a time $t + Y_{k,T(t,k)}$.
- the reward is $X_{k,T(t,k)}$.
- $Y_{k,j}$ & $X_{k,j}$ are i.i.d s
- $X_{k,j}$ & $Y_{k,j}$-sub-Gaussian with mean x_k & y_k

We are considering the following two cases.

Case I: $Y_{k,j}$ realization is not known on pulling k. You are not allowed to "quit" waiting for reward of k & pull another arm.

Case II: $Y_{k,j}$ realization is known as soon as we pull k. You are allowed to "quit" waiting for reward of k & pull another arm. You get no reward from k and incur a one unit time loss, if we quit.

CHAPTER 4

CASE-1

In this chapter we propose a Upper Confidence Bound(UCB) like algorithm for case-1 of the sticky bandit problem. We also find the regret upper bound which is a good metric to show the effectiveness of the algorithm. We try to obtain a sub-linear regret upper bound.

4.1 Regret Function

We obtain a regret upper bound for the sticky bandit problem using UCB, in a method similar to the standard MAB problem Prashanth.L.A (2018).
At each turn we pick an arm with gain $\{X_k \sim 1 \text{ sub-Gaussian}\}_{k=1}^{\kappa}$ and delays (time you get stuck in the arm) $\{Y_k \sim 1 \text{ sub-Gaussian}\}_{k=1}^{\kappa}$
Just as in Prashanth.L.A (2018), we assume 1^{st} arm to give largest sum of rewards.
Regret function is given by

$$\text{Regret} = \mathbb{E}\left[T_1^1\right] x_1 - \sum_{k=1}^{\kappa} \mathbb{E}\left[T_k\right] x_k \tag{4.1}$$

where,

$T_1^1 =$ # times 1st arm is pulled in most optimal route(Always picking arm-1)

$T_k =$ # times k^{th}-arm is pulled according to our algorithm

$$x_i = \mathbb{E}(X_i) \qquad y_i = \mathbb{E}(Y_i)$$

In the optimal strategy (picking arm-1), let the delays be $Y_1^{(1)}, Y_1^{(2)}, Y_1^{(3)}, \ldots$
From Wald's equation:

$$\sum_i Y_1^{(i)} = \mathbb{E}\left[T_1^1\right] y_1 = n$$
$$\implies \mathbb{E}\left[T_1^1\right] = \frac{n}{y_1} \tag{4.2}$$

Put equation(4.2) in (4.1),

$$\text{Regret} = n\frac{x_1}{y_1} - \sum_{k=1}^{\kappa} \mathbb{E}\left[T_k\right] x_k \tag{4.3}$$

Here $'n'$ is the time up to which we are allowed to pick an arm. As a result, we have the following.

$$\sum_{k=1}^{\kappa} \mathbb{E}\left[T_k\right] y_k \geq n \tag{4.4}$$

From equations (4.3) and (4.4), we get

$$\boxed{\text{Regret} \leq \sum_{k=1}^{\kappa} \mathbb{E}\left[T_k\right] \Delta_k}$$

$$\text{where, } \Delta_k = \frac{x_1}{y_1} y_k - x_k$$

This is what we try to optimize from now on.
Looking at the regret function it is clear that at each turn we want to pick k that maximizes expected reward rate (x_k/y_k).

4.2 Method

Our objective is to pick the arm with maximum expected reward rate x_k/y_k, which is the 1^{st}arm in our case. We now derive the Upper-Confidence-Bound(UCB) of the k^{th}arm calculated before the i^{th}pull.

$\hat{x}_k$ and $\hat{y}_k$ are empirical means of X_k and Y_k respectively.

Because of our initial assumption of X_k and Y_k for all k being 1-sub Gaussian, by following Prashanth.L.A (2018)

$$P\left[\hat{y}_k(i-1) \geq y_k + \epsilon\right] \leq \exp\left(\frac{-T_k(i-1)\epsilon^2}{2}\right)$$

$$P\left[\hat{x}_k(i-1) \leq x_k - \epsilon\right] \leq \exp\left(\frac{-T_k(i-1)\epsilon^2}{2}\right)$$

These can be re-written as

$$P\left[\hat{y}_k(i-1) \geq y_k + \sqrt{\frac{2\log(1/\delta)}{T_k(i-1)}}\right] \leq \delta$$

$$P\left[\hat{x}_k(i-1) \leq x_k - \sqrt{\frac{2\log(1/\delta)}{T_k(i-1)}}\right] \leq \delta$$

We choose $\delta = 1/i^4$ by following Auer *et al.* (2002) to get a sub-linear regret. From this

$$P\left[\hat{y}_k(i-1) \geq y_k + \sqrt{\frac{8\log\ i}{T_k(i-1)}}\right] \leq \frac{1}{i^4}$$

$$P\left[\hat{x}_k(i-1) \leq x_k - \sqrt{\frac{8\log\ i}{T_k(i-1)}}\right] \leq \frac{1}{i^4}$$

Taking this into consider we define UCB of arm k as

$$\mathbf{UCB}(k, i-1) = \frac{\hat{x}_k(i-1) + \sqrt{\frac{8\log i}{T_k(i-1)}}}{\hat{y}_k(i-1) - \sqrt{\frac{8\log i}{T_k(i-1)}}}$$

This upper bound is violated with the probability $\frac{2}{i^4}$.

$$P\left\{UCB(k, i-1) \leq \frac{x_k}{y_k}\right\} \leq \frac{2}{i^4}$$

At any i^{th} pull, we pick the k^{th}arm when its UCB is maximum among the different arms.

$$\frac{\hat{x}_k(i-1) + \sqrt{\frac{8\log i}{T_k(i-1)}}}{\hat{y}_k(i-1) - \sqrt{\frac{8\log i}{T_k(i-1)}}} > \frac{\hat{x}_1(i-1) + \sqrt{\frac{8\log i}{T_1(i-1)}}}{\hat{y}_1(i-1) - \sqrt{\frac{8\log i}{T_1(i-1)}}} \tag{4.5}$$

4.3 Finding Regret Upper Bound

4.3.1 Part-1

The above inequality(4.5) is satisfied in the following cases

$$\text{Case-A :} \quad \hat{x}_1(i-1) < x_1 - \sqrt{\frac{8\log i}{T_1(i-1)}}$$

$$\text{Case-B :} \quad \hat{y}_1(i-1) > y_1 + \sqrt{\frac{8\log i}{T_1(i-1)}}$$

$$\text{Case-C :} \quad \hat{x}_k(i-1) > x_k + \sqrt{\frac{8\log i}{T_k(i-1)}}$$

$$\text{Case-D :} \quad \hat{y}_k(i-1) < y_k - \sqrt{\frac{8\log i}{T_k(i-1)}}$$

Case-E : The original inequality can also be satisfied in cases where Case-A, B, C and D are not followed

$$\frac{\hat{x}_k + \sqrt{\frac{8\log i}{T_k}}}{\hat{y}_i - \sqrt{\frac{8\log i}{T_k}}} > \frac{\hat{x}_1 + \sqrt{\frac{8\log i}{T_1}}}{\hat{y}_1 - \sqrt{\frac{8\log i}{T_1}}}$$

From the converses of Case-A, B, C and D, we get

$$\frac{x_k + 2\sqrt{\frac{8\log i}{T_k}}}{y_k - 2\sqrt{\frac{8\log i}{T_k}}} > \frac{x_1}{y_1}$$

$$\implies T_k < \frac{32\log i}{\Delta_k^2}\left(1 + \frac{x_1}{y_1}\right)^2$$

Suppose there is an $\widetilde{n}$ for which the following inequality is always satisfied

$$\sum_{k=1}^{\kappa} T_k \leq \widetilde{n} \text{ (This makes sense in a few cases like delays being bounded)}$$

$$\implies i \leq \widetilde{n}$$

For $a = \left(1 + \frac{x_1}{y_1}\right)^2$ when $\boxed{T_k(i-1) \geq u = \frac{32a\log\widetilde{n}}{\Delta_k^2}}$

Case-E is not possible and we only have to look at Cases A, B, C and D.

Cases A, B, C and D each occurs with a probability $\leq \frac{1}{i^4}$.

4.3.2 Part-2

$$\begin{aligned}
T_k(n) &\leq u + \sum_{i=u+1}^{\infty} \mathbb{I}\left\{ \frac{\hat{x}_k + \sqrt{\frac{8\log i}{T_k}}}{\hat{y}_k - \sqrt{\frac{8\log i}{T_k}}} \geq \frac{\hat{x}_1 + \sqrt{\frac{8\log i}{T_1}}}{\hat{y}_1 - \sqrt{\frac{8\log i}{T_1}}} \right\} \\
&\leq u + \sum_{i=u+1}^{\infty} \mathbb{I}\left\{ \min_{u<s_k<i} \frac{\hat{x}_k + \sqrt{\frac{8\log i}{s_k}}}{\hat{y}_k - \sqrt{\frac{8\log i}{s_k}}} \geq \max_{0<s<i} \frac{\hat{x}_1 + \sqrt{\frac{8\log i}{s}}}{\hat{y}_1 - \sqrt{\frac{8\log i}{s}}} \right\} \\
&\leq u + \sum_{i=u+1}^{\infty} \sum_{s=1}^{i-1} \sum_{s_k=u}^{i-1} \mathbb{I}\left\{ \frac{\hat{x}_k + \sqrt{\frac{8\log i}{s_k}}}{\hat{y}_k - \sqrt{\frac{8\log i}{s_k}}} \geq \frac{\hat{x}_1 + \sqrt{\frac{8\log i}{s}}}{\hat{y}_1 - \sqrt{\frac{8\log i}{s}}} \right\} \\
\mathbb{E}[T_k] &\leq u + \sum_{i=0}^{\infty} \sum_{s=1}^{i-1} \sum_{s_k=u}^{i-1} P(G)
\end{aligned}$$

where

$$\begin{aligned}
G &= \left\{ \hat{x}_k + \sqrt{\frac{8\log i}{s_k}} > x_k \right\} \bigcup \left\{ \hat{x}_1 - \sqrt{\frac{8\log i}{s}} < x_1 \right\} \\
&\quad \bigcup \left\{ \hat{y}_k - \sqrt{\frac{8\log i}{s_k}} < y_k \right\} \bigcup \left\{ \hat{y}_1 + \sqrt{\frac{8\log i}{s}} > y_1 \right\}
\end{aligned}$$

$$\text{and} \qquad P(G) \leq \frac{4}{i^2}$$

$$\begin{aligned}
\text{we get} \qquad \mathbb{E}[T_k] &\leq u + \sum_{i=0}^{\infty} \sum_{s=1}^{i-1} \sum_{s_k=u}^{i-1} \frac{4}{i^2} \\
&\leq u + 2\left(1 + \frac{\pi^2}{3}\right)
\end{aligned}$$

4.3.3 Part-3

$$y_l = \min_k y_k \qquad y_m = \max_k y_k$$

$$\sum_{k=1}^{\kappa} y_k \mathbb{E}[T_k] \leq n + y_m$$

4.3.4 Part-4

$$
\begin{aligned}
R_n &\leq \sum_{k=1}^{\kappa} \Delta_k \mathbb{E}[T_k] \\
&\leq \left(\sum_{k=1}^{\kappa} \frac{\Delta_k^2 \mathbb{E}[T_k]}{y_k} \right)^{1/2} \left(\sum_{k=1}^{\kappa} y_k \mathbb{E}[T_k] \right)^{1/2} \qquad \text{(Cauchy-Schwarz inequality)} \\
&\leq \left(\sum_{k=1}^{\kappa} \frac{\Delta_k^2 \mathbb{E}[T_k]}{y_l} \right)^{1/2} (n + y_m)^{1/2} \\
&\leq \frac{1}{\sqrt{y_l}} \left[32\kappa a \log \widetilde{n} + 2\left(1 + \frac{\pi^2}{3}\right) \sum_{k=1}^{\kappa} \Delta_k^2 \right]^{1/2} \sqrt{n + y_m}
\end{aligned}
$$

$$
\boxed{R_n \leq C' \sqrt{n + y_m}}
$$

where

$$
C' = \frac{1}{\sqrt{y_l}} \left[32\kappa a \log \widetilde{n} + 2\left(1 + \frac{\pi^2}{3}\right) \sum_{k=1}^{\kappa} \Delta_k^2 \right]^{1/2}
$$

Here, we obtained a sub-linear regret for the case-1 of the sticky banded problem. $\boxed{\text{Regret} \propto \sqrt{n}}$.

CHAPTER 5

CASE-2

Given the distribution of Y_k, say $P_k(.)$, then the optimization strategy is to every time pull the arm k^* given by

$$k^* = \underset{k}{\operatorname{argmax}} \left\{ \max_{\Delta} \frac{x_k P_k(Y_k \leq \Delta)}{P_k(Y_k > \Delta) + \mathbb{E}\left[Y_k | Y_k \leq \Delta\right] P_k(Y_k \leq \Delta)} \right\}$$

$$\overline{\Delta}_k = \underset{\Delta}{\operatorname{argmax}} \frac{x_k P_k(Y_k \leq \Delta)}{P_k(Y_k > \Delta) + \mathbb{E}\left[Y_k | Y_k \leq \Delta\right] P_k(Y_k \leq \Delta)}$$

We wait for reward if $Y_{k^*,j} \geq \overline{\Delta}_{k^*}$
else "quit" immediately.

We try to explain the logic behind this algorithm in the following
For a given threshold Δ

$$\begin{aligned} \bar{t} &= \text{average time spent after you pick arm } k \\ &\quad (\text{We incur one unit loss of time on quitting}) \\ &= 1 \times P_k\left(Y_k > \Delta\right) + \mathbb{E}\left(Y_k | Y_k < \Delta\right) \times P_k\left(Y_k < \Delta\right) \end{aligned}$$

Since X_k and Y_k are independent, we have

$$\mathbb{E}\left(X_k | Y_k < \Delta\right) = \mathbb{E}\left(X_k\right) = x_k \tag{5.1}$$

$$\begin{aligned} \bar{x} &= \text{Average reward from picking arm } k \\ &\quad (\text{We obtain zero reward on quitting}) \\ &= 0 \times P_k\left(Y_k > \Delta\right) + \mathbb{E}\left(X_k | Y_k < \Delta\right) \times P_k\left(Y_k < \Delta\right) \\ &= x_k \times P_k\left(Y_k < \Delta\right) \quad (\text{By equation(5.1)}) \end{aligned}$$

Average reward rate for Δ

$$= \frac{x_k P_k(Y_k \leq \Delta)}{P_k(Y_k > \Delta) + \mathbb{E}\left[Y_k | Y_k \leq \Delta\right] P_k(Y_k \leq \Delta)} \tag{5.2}$$

$$= \frac{x_k \sum_{j=1}^{\Delta} P_k(j)}{\sum_{j=\Delta}^{\infty} P_k(j) + \sum_{j=1}^{\Delta} j P_k(j)} \tag{5.3}$$

From equation(5.3), it is clear that for larger values of Δ, as Δ increases, average reward rate decreases (since the denominator increases at a faster rate). Therefore Δ maximizes the above expression for a finite value.

We use the optimal Δ for each arm and pick the arm which offers the best average. This way we obtain maximum reward per unit time.

5.1 Our suggestion for a UCB like algorithm

We can re-write the optimum average reward rate for each arm as the following.

$$U_k = \max_{\Delta} \left[\mathbb{E}\left[X_k\right] \frac{\mathbb{E}\left[\mathbf{I}\left(Y_k \leq \Delta\right)\right]}{\left(\mathbb{E}\left[\mathbf{I}\left(Y_k > \Delta\right)\right] + \mathbb{E}\left[Y_k \mathbf{I}\left(Y_k \leq \Delta\right)\right]\right)} \right]$$

Steps involved in i^{th} pull

1. Take all the samples from the kth arm $(X_{k,r}, Y_{k,r})$ $r = 1, 2 \ldots T_k(i)$
2. Estimate all the expectations using corresponding sample averages.
3. For every arm, search over all Δ. The estimate of U_k after i^{th} pull is $U_k(i)$.
4. Choose the arm - $\text{argmax}_k \, UCB\left(U_k(i)\right)$

In this work we have not found an expression for a proper upper confidence bound(UCB) of U_k. Given an appropriate expression for $UCB(U_k)$, we expect the above algorithm to give a smaller regret than the one found in case-1. This is because the average reward rate in this case is greater than that of case-1(case-1 reward rate for arm k is x_k/y_k).

CHAPTER 6

CONCLUSIONS

In this project we have proposed UCB like algorithms for two cases of sticky bandit problem - with unknown and known delays (Y_k), referred to as case-1 and case-2, respectively.

A metric showing the effectiveness of the algorithm is the regret upper bound. We have derived an expression for regret which helped us identify that we have to pick the arm with maximum average reward rate x_k/y_k at each pull.

The regret upper bound for the case-1 is found to be directly proportional to $\sqrt{n}$, where n is run-time. Such a sub-linear regret upper bound is preferred.

For case-2 we have partially developed a UCB like algorithm, which we expect to outperform the algorithm of case-1. To convert this into a full fledged algorithm, we need an expression for $UCB(U_k)$.

APPENDIX A

UCB algorithm for κ-armed bandit

$X_1, X_2, X_3, \ldots\ldots X_\kappa$ are rewards for κ arms. n is the number of pulls.

Reward=0
for $i = 1, 2, 3..., \kappa$ **do**
  Select i^{th} arm.
  $T(i) = 1$
  Calculate $UCB_{T(i)}(i)$ Reward=Reward+X_i
end
for $i = \kappa + 1, \kappa + 2, \kappa + 3..., n$ **do**
  Select k^{th} arm that has maximum UCB
  $k = \text{argmax}_j\, UCB_{T(j)}(j)$
  $T(k) = T(k) + 1$
  Reward=Reward+X_k
end

REFERENCES

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

2. **Carrascosa, M.** and **B. Bellalta** (2019). Decentralized ap selection using multi-armed bandits: Opportunistic ϵ-greedy with stickiness. *arXiv preprint arXiv:1903.00281*.

3. **Jun, K.-S.** and **R. D. Nowak**, Anytime exploration for multi-armed bandits using confidence information. *In ICML*. 2016.

4. **Lai, T.** and **H. Robbins** (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, **6**(1), 4 – 22. ISSN 0196-8858. URL `http://www.sciencedirect.com/science/article/pii/0196885885900028`.

5. **Mannor, S.**, k-armed bandit. *In* **C. Sammut** and **G. I. Webb** (eds.), *Encyclopedia of machine learning*, chapter 10. Springer Science & Business Media, 2011, 561–563.

6. **Prashanth.L.A** (2018). Cs6046: Multi-armed bandits. Click here to open the link. Accessed: 17-04-2020.

Formulation of image deblurring as an optimization problem and its convergence using alternate minimization

Report of Phase-1 DD-Project
Submitted by

A. JAYAKRISHNAN
(EE15B067)

Under the guidance of

Dr. Avhishek Chatterjee

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

Abstract

In this project, we try different formulations for blind deblurring of images as optimization problem, then try to prove its convergence. We consider Alternate Minimization for solving all of the formulated problems.

Chapter 1

Introduction

We try different formulations for blind deblurring. The deblurred image made into a vector and kernal(filter) are the varables. For most of the project we use Alternate Minimization as method of solution, and formulations of the problem are made in such a way that, the problem can be proven to converge using Theorem 4.3 from Prateek Jain [2017].
We have considered a few other methods and proofs for convergence, and will make attempts towards that in the future.

Chapter 2

Previous work

1. Robust Blind Deconvolution via Mirror Descent by Ravi et al. [2018]
   This paper deals with a convergence of an algorithm named PRIDA. The algorithm used is similar to Alternate Minimization with projection.

2. Expectation-Maximization and Alternating Minimization Algorithms by Haas [2002]
   This paper mainly deals with Expectation-Maximization and its use for certain problems like mixture models. It also talks about the similarities in Expectation-Maximization and Alternating Minimization.

3. IITM JTG 2019 optimization lectures by Praneeth NetrapalliNetrapalli [2019]
   The lecture notes deals with Methods like Gradient Descent(GD), projected GD, Mirror Descent etc. and its rate of convergence for various cases of objective like convexity, strong convexity, and smoothness.

4. Non-convex Optimization for Machine Learning, 2017 Prateek Jain [2017]
   Chapter 8 of the book discusses the convergence of low rank matrix completion using Alternate-Minimization. **Robust Bistability Property** from chapter 4 is what we tried using till now to prove convergence for our different formulations.

Chapter 3

Modelling of optimization problem

3.1 1st formulation

One of the first formulations we tried was.

$$\min\ f(A, \mathbf{x}) = \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda \|A\|_F^2$$

were, A is kernal and $\mathbf{x}$ is image pixel intensity value in column vector.
First the term $\|A\|_F^2$, was added to avoid the trivial solution, $A = I$ (Identity matrix) and $\mathbf{x} = \mathbf{y}$.
Later we realised this term also offers a constraint on A to be a stochastic matrix.
Let $\lambda_1, \lambda_2, \lambda_3$, ..etc. be the eigen values of A, then while A is stochastic

$$\max_i \lambda_i \leq 1 \ \Rightarrow\ \|A\|_F^2 \leq \sum_i \lambda_i^2 \leq n$$

By changing λ, we expect to get desired deblurred image. Because of the form of the problem, being marginally convex in both A and $\mathbf{x}$, we consider Alternate Minimization(Algorithm 1) as method of solution. We try to prove convergence using Theorem 3.1. A motivation towards this is We also consider an alternate form, equivalent to the C-Robust Bistability Property, in Result 3.1.

Definition 3.1. *$f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ is $\alpha-$Marginally Strongly Convex(MSC) and $\beta-$Marginally Strongly Smooth(MSS) in its first variable if $\forall\ y \in \mathbb{R}^q$,*

$$\frac{\alpha}{2}\left\|\mathbf{x}^2 - \mathbf{x}^1\right\|_2^2 \leq f(\mathbf{x}^2, \mathbf{y}) - f(\mathbf{x}^1, \mathbf{y}) - \left\langle \mathbf{g}, \mathbf{x}^2 - \mathbf{x}^1 \right\rangle \leq \frac{\beta}{2}\left\|\mathbf{x}^2 - \mathbf{x}^1\right\|_2^2$$

where $g = \nabla_{\mathbf{x}} f\left(\mathbf{x}^1, \mathbf{y}\right)$ *and* $\mathbf{x}^1, \mathbf{x}^2 \in \mathbb{R}^p$.

Definition 3.2. A function $f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ satisfies the C-robust bistability Property if for some $C > 0$, for every $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^p \times \mathbb{R}^q$, $\tilde{\mathbf{y}} \in \mathbf{mOPT}_f(\mathbf{x})$ and $\tilde{\mathbf{x}} \in \mathbf{mOPT}_f(\mathbf{y})$, we have

$$f(\mathbf{x}, \mathbf{y}^*) + f(\mathbf{x}^*, \mathbf{y}) - 2f^* \leq C.\left[2f(\mathbf{x}, \mathbf{y}) - f(\mathbf{x}, \tilde{\mathbf{y}}) - f(\tilde{\mathbf{x}}, \mathbf{y})\right]$$

Data: Objective function $f : X \times Y \to \mathbb{R}$
Result: A point with near optimal objective value
$(x^1, y^1) \leftarrow$ Initialize();
for $t = 1, 2, 3..., T$ **do**

$$x^{t+1} \leftarrow \underset{\mathbf{x}}{\operatorname{argmin}}\, f(x, y^t)$$

$$y^{t+1} \leftarrow \underset{\mathbf{y}}{\operatorname{argmin}}\, f(x^{t+1}, y)$$

end
return $(\mathbf{x}^T, \mathbf{y}^T)$

Algorithm 1: Alternate Minimization

Theorem 3.1. *(from Prateek Jain [2017])* Let $f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ be a continuously differentiable function, within the region $S_0 = \left\{\mathbf{x}, \mathbf{y} : f(\mathbf{x}, \mathbf{y}) \leq f(\mathbf{0}, \mathbf{0})\right\} \subset \mathbb{R}^{p+q}$,satisfies $\alpha - MSC$, $\beta - MSS$ in both its variables and C-robust bistability . Let alternating minimization be executed with $(\mathbf{x}^1, \mathbf{y}^1) = (\mathbf{0}, \mathbf{0})$. Then after at most $T = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$ steps, we have $f\left(\mathbf{x}^T, \mathbf{y}^T\right) \leq f^* + \epsilon$

Result 3.1. *A function which is $\alpha-$MSC and $\beta-$MSS follows Robust Bistability Property if it follows*

$$\left\|\mathbf{x} - \mathbf{x}^*\right\|_2^2 + \left\|\mathbf{y} - \mathbf{y}^*\right\|_2^2 \leq k\{\|\mathbf{x} - \tilde{\mathbf{x}}\|_2^2 + \|\mathbf{y} - \tilde{\mathbf{y}}\|_2^2\}$$

Along with the original inequality, we try to show that the objective follows this inequality to prove convergence.

Proof: As $\nabla f|_{(\mathbf{x}^*, \mathbf{y}^*)} = \mathbf{0}$, $\alpha-$MSC and $\beta-$MSS of $f(\mathbf{x}, \mathbf{y})$ gives us

$$\frac{\alpha}{2}\left\|\mathbf{x} - \mathbf{x}^*\right\|_2^2 \leq f(\mathbf{x}, \mathbf{y}^*) - f(\mathbf{x}^*, \mathbf{y}^*) \leq \frac{\beta}{2}\left\|\mathbf{x} - \mathbf{x}^*\right\|_2^2 \tag{3.1}$$

$$\frac{\alpha}{2}\left\|\mathbf{y} - \mathbf{y}^*\right\|_2^2 \leq f(\mathbf{x}^*, \mathbf{y}) - f(\mathbf{x}^*, \mathbf{y}^*) \leq \frac{\beta}{2}\left\|\mathbf{y} - \mathbf{y}^*\right\|_2^2 \tag{3.2}$$

As $\nabla_x f|_{(\tilde{\mathbf{x}},\mathbf{y})} = \mathbf{0}$

$$\frac{\alpha}{2}\|\mathbf{x}-\tilde{\mathbf{x}}\|_2^2 \leq f(\mathbf{x},\mathbf{y}) - f(\tilde{\mathbf{x}},\mathbf{y}) \leq \frac{\beta}{2}\|\mathbf{x}-\tilde{\mathbf{x}}\|_2^2 \tag{3.3}$$

As $\nabla_y f|_{(\mathbf{x},\tilde{\mathbf{y}})} = \mathbf{0}$

$$\frac{\alpha}{2}\|\mathbf{y}-\tilde{\mathbf{y}}\|_2^2 \leq f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\tilde{\mathbf{y}}) \leq \frac{\beta}{2}\|\mathbf{y}-\tilde{\mathbf{y}}\|_2^2 \tag{3.4}$$

From equations 3.1, 3.2, 3.3 and 3.4

$$\begin{aligned} &f(\mathbf{x},\mathbf{y}^*) + f(\mathbf{x}^*,\mathbf{y}) - 2f^* \leq C.\left[2f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\tilde{\mathbf{y}}) - f(\tilde{\mathbf{x}},\mathbf{y})\right] \\ &\Rightarrow \alpha\{\|\mathbf{x}-\mathbf{x}^*\|_2^2 + \|\mathbf{y}-\mathbf{y}^*\|_2^2\} \leq C \times \beta\{\|\mathbf{x}-\tilde{\mathbf{x}}\|_2^2 + \|\mathbf{y}-\tilde{\mathbf{y}}\|_2^2 \end{aligned} \tag{3.5}$$

,and

$$\begin{aligned} &\left\{\|\mathbf{x}-\mathbf{x}^*\|_2^2 + \|\mathbf{y}-\mathbf{y}^*\|_2^2\right\} \leq C^1 \times \alpha\left\{\|\mathbf{x}-\tilde{\mathbf{x}}\|_2^2 + \|\mathbf{y}-\tilde{\mathbf{y}}\|_2^2\right\} \\ &\Rightarrow \alpha f(\mathbf{x},\mathbf{y}^*) + f(\mathbf{x}^*,\mathbf{y}) - 2f^* \leq C^1 \times \beta\left[2f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\tilde{\mathbf{y}}) - f(\tilde{\mathbf{x}},\mathbf{y})\right] \end{aligned} \tag{3.6}$$

From equations 3.5 and 3.5, it is clear that robust bistability is equivalent to our formulation.

We have tried to prove that the objective satisfies the inequalities in Definition 3.2 and Result 3.1, but failed at it. One of the issues faced was, not knowing the values of $(\mathbf{x}^*,\mathbf{y}^*)$ that were necessary for the both inequalities. Our hope was that, we could prove it with some assumptions of optimal point and function characteristics.

$$\tilde{\mathbf{x}} = (AA^T)A\mathbf{y}$$
$$\tilde{A} = (\mathbf{x}\mathbf{x}^T + \lambda I)^{-1}\mathbf{x}\mathbf{y}^T$$

here A at every step of the AM-algorithm is a rank 1 matrix. This causes trouble in finding $\mathbf{x}$ in the next step. We try other formulations that better characterises the problem.

3.2 2nd formulation

In the second formulation, we apply the condition that A is stochastic.

$$\min f(A, \mathbf{x}) = \{\sum_i \left\| y_i - \langle \mathbf{a}_i, \mathbf{x} \rangle \right\|_2^2 + \lambda \|\mathbf{a}_i\|^2\}$$

$$\text{Subject to } \langle \mathbf{a}_i, \mathbf{1} \rangle = 1$$

We face the same problems as the first formulation ($\tilde{A}$ is a rank 1 matrix, and the problem with lack of information of $(\mathbf{x}^*, \mathbf{y}^*)$).

3.3 3rd formulation

We now tried to add the constraint that A is doubly stochastic. Doubly stochastic matrices can be represented as linear combinations of permutation matrices.

$$A = \sum_i z_i P_i$$

where $\forall i$, $z_i \geq 0$, $\sum_i z_i = 1$ and P_i are permutation matrices.
Our problem becomes

$$\min f(\mathbf{z}, \mathbf{x}) = \left\| \mathbf{y} - \{\sum_i z_i P_i\}\mathbf{x} \right\|_2^2 + \sum_i z_i^2$$

$$\text{Subject to: } \mathbf{z} \succcurlyeq \mathbf{0} \text{ and } \langle \mathbf{1}, \mathbf{z} \rangle = 1$$

$$\mathbf{y}^T\mathbf{y} + \mathbf{x}^T \sum_{i,j} z_i z_j P_i^T P_j \, \mathbf{x} - 2\mathbf{y}^T \sum_i z_i P_i \, \mathbf{x} + \lambda \sum_i z_i^2$$

$$\frac{\partial f}{\partial z_k} = \sum_{j,i} \mathbf{x}^T \left(z_i P_i^T P_j + z_j P_i^T P_j \right) \mathbf{x} - 2\mathbf{y}^T P_k \mathbf{x} + 2\lambda z_k$$

$$\nabla_x f = 2 \sum_{j,i} z_i z_j P_i^T P_j \mathbf{x} - 2 \sum_i z_i P_i^T \mathbf{y} = 0$$

We obtain the $\tilde{\mathbf{x}}$ and $\tilde{A}$ from the above equations.

We still face problems with proving convergence through Theorem 3.1 as we do not have knowledge about $(\mathbf{x}^*, \mathbf{y}^*)$.

3.4 Trying simpler problems

To check if it is possible to prove convergence using Theorem 3.1, we consider a simpler problem whose solution we know.

$$\min \ f(A, \mathbf{x}) = \|\mathbf{y} - A\mathbf{x}\|_2^2$$

The solution to this is $A^* = I$ and $\mathbf{x}^* = \mathbf{y}$.
This is when we realised a fault to most of our formulations. None of them follow robust bistability property. Even for the above simple problem, there are multiple bistable points to which the algorithm could converge. By interchanging any two any two rows of A^* and corresponding elements of $\mathbf{x}^*$, we get another solution(value of objective does not change with this operation).

In Result 3.1 , Suppose $(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^1, \mathbf{y}^1)$, where $(\mathbf{x}^1, \mathbf{y}^1)$ which is another bistable point, other than $(\mathbf{x}^*, \mathbf{y}^*)$. In this case we get $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = (\mathbf{x}^1, \mathbf{y}^1)$, and the inequality needed to be proven becomes

$$\left\|\mathbf{x}^1 - \mathbf{x}^*\right\|_2^2 + \left\|\mathbf{y}^1 - \mathbf{y}^*\right\|_2^2 \leq 0$$

. Which is not true.

Chapter 4

Results and Work to be done

A mentioned in the end of the previous chapter there are other considerations to be taken to the objective of the optimization problem, if we want to prove convergence using Theorem 3.1. Terms can be added to the objective to avoid multiple bistable points. Gradient term of $\mathbf{x}$ or terms like $\|\mathbf{y} - \mathbf{x}\|$ could be used for this purpose. We are considering this for the future of this project.

Theorem 3.1 is too general. We might need to consider a proof that is specific to the problem. Chapter-8 of Prateek Jain [2017] has convergence proof for low-rank matrix completion by Prateek Jain. We are going through such proofs as part of research study, to possible find a proof for our problem.

Bibliography

Shane M. Haas. The expectation-maximization and alternating minimization algorithms. 2002.

Praneeth Netrapalli. Optimization, lecture, 2019. URL `http://www.ee.iitm.ac.in/jtg/program.html`.

Purushottam Kar Prateek Jain. *Non-convex Optimization for Machine Learning*. http://dx.doi.org/10.1561/2200000058, 2017.

Sathya N. Ravi, Ronak Mehta, and Vikas Singh. Robust blind deconvolution via mirror descent. 2018.