ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αξιοποίηση Εκτιμήσεων Διασποράς σε Στοχαστικά MAB Προβλήματα με Αλλοιωμένες Ανταμοιβές

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΕΡΑΛΝΤ ΣΙΝΑΝΑΙ

Επιβλέπων: Δημήτρης Φωτάκης
Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2024

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Αξιοποίηση Εκτιμήσεων Διασποράς σε Στοχαστικά MAB Προβλήματα με Αλλοιωμένες Ανταμοιβές

Διπλωματική Εργασία

του

ΕΡΑΛΝΤ ΣΙΝΑΝΑΙ

Επιβλέπων: Δημήτρης Φωτάκης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 24η Σεπτεμβρίου 2024.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
....	....	....
Δημήτρης Φωτάκης	Αριστείδης Παγουρτζής	Νικόλαος Λεονάρδος
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Επίκουρος Καθηγητής Ε.Μ.Π

Αθήνα, Σεπτέμβριος 2024

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Υπεύθυνη Δήλωση

Βεβαιώνω ότι είμαι συγγραφέας αυτής της πτυχιακής εργασίας, και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην πτυχιακή εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η πτυχιακή εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τομέα Τεχνολογίας Πληροφορικής και Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου.

(Υπογραφή)

....
Έραλντ Σινανάι
Διπλωματούχος Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Στην παρούσα διπλωματική εργασία μελετάμε το πρόβλημα των *Multi Armed Bandits (MAB)*, το οποίο αφορά την άμεση μάθηση σε περιβάλλοντα περιορισμένης ανάδρασης. Το πρώτο μέρος της διπλωματικής αφορά τη στοχαστική μορφή του προβλήματος. Διάφοροι αλγόριθμοι έχουν προταθεί και μελετηθεί για αυτό το πρόβλημα, μερικοί πιο απλοί και 'μη προσαρμοστικοί', ενώ άλλοι πιο ισχυροί, αλλά πιο περίτεχνοι. Μελετάμε τους σημαντικότερους από αυτούς και αναλύουμε την απόδοση τους μέσω της μετρικής της 'μετάνοιας' (*regret*), αποδεικνύοντας εγγύησεις με *υψηλή πιθανότητα* (*high probability regret guarantees*). Ακόμα, μελετάμε τον αλγόριθμο UCBV που εκτιμάει πέρα από τη μέση τιμή και την διασπορά κάθε ενέργειας. Αυτό οδηγεί σε βελτιωμένη εγγύηση όταν οι ενέργειες είναι πιο στατικές, παράλληλα εξασφαλίζοντας την εγγύηση των σχεδόν βέλτιστων αλγορίθμων που διατηρούν μόνο εκτιμήσεις μέσης τιμής. Ταυτόχρονα, αποδεικνύουμε αυτό το αποτέλεσμα με μια σημαντικά απλούστερη ανάλυση από αυτή στην αρχική δημοσίευση.
Στο δεύτερο μέρος μελετάμε το πρόβλημα των ανταγωνιστικών MAB (*Adversarial MAB*), όπου δεν κάνουμε καμία στατιστική υπόθεση. Αρχικά μελετάμε το πρόβλημα του *Online Learning* για να κατανοήσουμε στην πορεία τον ευρέως γνωστό αλγόριθμο Exp3. Στη συνέχεια εστιάζουμε στο πρόβλημα των Στοχαστικών MAB υπό ανταγωνιστικές αλλοιώσεις (*Adversarially Corrupted MABs*). Σε αυτό το πρόβλημα το υποκείμενο περιβάλλον είναι στοχαστικό, αλλά κάποιος ανταγωνιστής μπορεί να αλλοιώσει τις ανταμοιβές πριν παρατηρηθούν. Ο πρώτος αλγόριθμος που μελετάμε σε αυτό το πρόβλημα εξασφαλίζει σχεδόν-βέλτιστη εγγυήση μετάνοιας στο αμιγώς στοχαστικό περιβάλλον η οποία μεταβάλλεται ομαλά με την αύξηση της σωρευτικής αλλοίωσης. Ακόμη, μελετάμε τον αλγόριθμο BARBAR ο οποίος επιτυγχάνει σημαντικά καλύτερη εγγύηση. Τέλος, βασισμένοι σε αυτόν παρουσιάζουμε έναν αλγόριθμο ο οποίος χρησιμοποιεί επιπλέον εκτιμήσεις διασποράς των ανταμοιβών, σε μια προσπάθεια να επιτύγχουμε μετάνοια η οποία μεταβάλλεται ομαλά από αυτό που εξασφαλίζει ο UCBV σε στοχαστικό περιβάλλον. Πράγματι, αποδεικνύουμε ότι ο αλγόριθμος μας εξασφαλίζει σχεδόν όμοια εγγύηση μετάνοιας σε αμιγώς στοχαστικό περιβάλλον, αλλά ένα παράδειγμα οικογενειών στιγμιοτύπων δείχνει ότι ο αλγόριθμος στην τωρινή μορφή του δεν μπορεί να επιτύχει τα επιθυμητά αποτελέσματα.

Λέξεις Κλειδιά

Περιορισμένη Ανάδραση, Διαδοχική Λήψη Αποφάσεων Υπό Αβεβαιότητα, Μετάνοια, Ανταγωνιστική Μάθηση, Ανταγωνιστική Αλλοίωση, Εκτίμηση Μέσης Τιμής, Εκτίμηση Διασποράς

Abstract

In this thesis we study the problem of *Multi Armed Bandits (MAB)*, a central problem in the interface of Statistics and Computer Science where a learner has to make sequential decisions interacting with an environment. The first part concerns the *Stochastic MAB* problem. Many algorithms have been employed for this environment achieving sublinear regret, ranging from naive non-adaptive ones to adaptive ones based on uniform confidence bounds on the sample average. We study some of the major ones, proving their regret guarantees in a high probability fashion on the way. We also study UCBV, a MAB algorithm which keeps track of the empirical variance of the arms, providing a gracefully improved regret bound when variances are sufficiently small and attaining at most the regret bound of more well known almost-optimal algorithms. We also provide a much simplified analysis of its regret guarantee than the one on the original paper.

In the second part of the thesis we turn our attention to more complicated settings. First we study the *Adversarial Multi Armed Bandit* problem and the very well known Exp3 algorithm. We study the simpler problem of Online Learning on the way to understand the well-known Exp3 algorithm and then present the analysis of its sublinear regret bound. After that, we move to the problem of *Adversarially Corrupted Multi Armed Bandits*, where the environment is inherently stochastic but an adversary can corrupt the rewards that the learner sees. The first algorithm we study is due to the authors who also introduced this setting and it attains the close to optimal regret bound for the purely stochastic case while gracefully degrading as the corruption increases. We also study an algorithm called BARBAR for the *adversarially corrupted MAB* problem that attains even better regret bounds, handling corruption up to $o(T)$. Finally, based on this we provide an algorithm that uses variance estimates, in an effort to attain UCBV-like improved bounds whenever the stochastic nature of the arms does not exhibit large variance. We show that our algorithm does recover the regret bound of UCBV in the fully stochastic case, but a key example family of instances shows that in its current form it cannot attain the desired bounds.

Keywords

Bandit Feedback, Sequential Decision Making Under Uncertainty, Regret, Confidence Bound, Adversarial Learning, Adversarial Corruptions, Mean Estimates, Variance Estimates

Ευχαριστίες

Ο πρώτος άνθρωπος που θα ήθελα να ευχαριστήσω είναι ο επιβλέποντας αυτής της διπλωματικής, ο κ. Δημήτρης Φωτάκης. Ο κινητήριος τροχός που με ώθησε να ασχοληθώ με την Θεωρητική Πληροφορική ήταν ο ίδιος, εισάγοντας με σε αυτόν τον κόσμο με το αστείρευτο πάθος του και τις γνώσεις τις οποίες μετέδιδε στις διαλέξεις. Τον ευχαριστώ για την συνεχή υποστήριξη του και καθοδήγηση η οποία δεν περιορίστηκε μόνο στα πλαίσια της διπλωματικής εργασίας. Είμαι ευγνώμων που έλαβα την εμπιστοσύνη του, το ενδιαφέρον του και τον πολύτιμο χρόνο που αφιέρωσε σε εμένα και τον ευχαριστώ για όλα αυτά βαθύτατα. Θέλω ακόμα να ευχαριστήσω την Ελισάβετ, που ήταν πάντα στο πλευρό μου, σε κάθε χαρά αλλά και σε κάθε δυσκολία όντας παρούσα. Την ευχαριστώ αμέτρητα που έκανε την φοιτηση και τη ζωή μου τόσο πιο όμορφη και καλύτερη. Τέλος, θέλω να ευχαριστήσω την μητέρα μου, έναν άνθρωπο που αντιμετώπισε αμέτρητες δυσκολίες, αλλά παρ’όλα αυτά επέμεινε ως το τέλος και με την ατέρμονη αγάπη και δύναμη που είχε προσπαθούσε πάντα να μου παρέχει μια καλύτερη ζωή. Χωρίς αυτούς τους ανθρώπους δε θα ήμουν σε αυτό το σημείο, σας ευχαριστώ βαθιά από την καρδιά μου.

Περιεχόμενα

Περίληψη 1

Abstract 3

Ευχαριστίες 5

Εκτεταμένη Ελληνική Περίληψη 9

0.1 Εισαγωγή . . . 9

0.2 Μοντέλο . . . 10

0.3 Μετάνοια . . . 11

0.4 Στοχαστικό πρόβλημα MAB . . . 11

0.5 Εξερεύνηση πρώτα (ETC: Explore then Commit) . . . 12

0.6 Active Arm Elimination (AAE) . . . 13

0.7 Κάτω φράγματα μετάνοιας . . . 14

0.8 Upper Confidence Bound (UCB) . . . 15

0.9 UCB-V (UCB με εκτίμηση διασποράς) . . . 15

0.10 Ανταγωνιστικό πρόβλημα MAB (Adversarial MAB problem) . . . 17

0.11 Στοχαστικό πρόβλημα MAB υπό ανταγωνιστικές αλλοιώσεις . . . 18

0.12 Multi-Layer AAE Race . . . 19

0.13 Ο αλγόριθμος BARBAR . . . 21

0.14 Εκτιμήσεις Διασποράς σε αλλοιωμένο περιβάλλον . . . 22

0.15 Επίλογος . . . 24

Κείμενο στα Αγγλικά 27

1 The Multi Armed Bandit problem 28

1.1 Introduction . . . 28

1.2 Structure . . . 29

1.3 The model . . . 30

1.4 Regret . . . 31

2 Stochastic MAB 33

2.1 Explore then Commit (ETC) . . . 33

2.2 Active Arm Elimination (AAE) . . . 36

2.3 Upper Confidence Bound Algorithm (UCB) 40

2.4 Upper Confidence Bound with Variance estimates (UCB-V) 42

3 Adversarial MAB (and Online Learning) 50

3.1 Introduction 50

3.2 Full Feedback 51

3.3 Bandit Feedback and Sublinear regret through EXP3 54

4 Stochastic Bandits with Adversarial Corruptions 58

4.1 Introduction 58

4.2 Adversarial corruptions 59

4.3 Multi Layer Active Elimination Race 60

4.4 BARBAR algorithm 63

5 Variance Estimates in Corrupted Bandits 67

5.1 Intro 67

5.2 Preliminary Analysis 69

5.3 Bounding the empirical variance 70

5.4 Guaranteeing a good estimation of the rewards 74

5.5 Bounding estimated gaps 75

5.6 Regret 78

5.7 Future work 79

A Concentration Inequalities 80

Εκτεταμένη Ελληνική Περίληψη

0.1 Εισαγωγή

Θεωρήστε ότι βρίσκεστε σε ένα καζίνο και σας αρέσουν πολύ τα τυχερά παιχνίδια και ιδιαίτερα οι "κουλοχέρηδες" ή αλλιώς ληστές (bandits). Το καζίνο που συνηθίζετε να επισκέφτεστε διαθέτει ένα συγκεκριμένο αριθμό τέτοιων μηχανών. Σε δεδομένο χρονικό διάστημα σχεδιάζετε να μεγιστοποιήσετε τα κέρδη σας μαθαίνοντας ποιος ληστής είναι ο "καλύτερος". Αυτό είναι το πρόβλημα της μάθησης με περιορισμένη ανάδραση (Multi Armed Bandits Problem).

Το πρόβλημα *MAB* (Multi Armed Bandits) αποτελεί ένα κεντρικό πρόβλημα στη τομή της Στατιστικής και της Θεωρητικής Πληροφορικής και είναι ένα πρόβλημα που ήδη από παλαιότερα απασχολούσε ερευνητές με πολλές πρακτικές εφαρμογές. Προτάθηκε για πρώτη φορά από τον Robbins το 1952, όταν μελετούσε στατιστικά συμπεράσματα από πληθυσμούς με στοχαστικό πλήθος δειγμάτων που εξαρτώ- νται από τις παρατηρήσεις [50]. Παρ'όλα αυτά το πρόβλημα *MAB* έχει τις ρίζες του ακόμα πιο πριν το 1933 στην δουλειά του Thompson η οποία εξέταζε πότε μια πιθανότητα υπερέχει μιας άλλης, δεδομένων περιορισμένων δειγμάτων.

Ένα από τα πρώτα πρακτικά ζητήματα που ερευνήθηκαν, όπου αναδύεται το πρόβλημα *MAB* είναι αυτό του αποδοτικού σχεδιασμού κλινικών δοκιμών. Πιο συγκεκριμένα, μια κλινική δοκιμή μπορεί να μοντελοποιηθεί ως μια σειρά από διαδοχικές αφίξεις ασθενών, οι οποίοι υποθέτουμε ότι έχουν μια κοινή "ομοιομορφία" ως προς τα χαρακτηριστικά της ασθένειας. Υποθέτουμε ότι υπάρχει ένας αριθμός θεραπειών ή συνδυασμός τους και καθεμία προτίθεται για δοκιμή και εξέταση της απόδοσης της. Θεωρώντας ότι κάθε απόβαση μιας πιθανής θεραπείας μπορεί να μετρηθεί σε κάποια κλίμακα, τότε αυτό το πρόβλημα μπορεί να συγκριθεί με το αρχικό πρόβλημα *MAB*. Οι ιατρικές εφαρμογές επεκτείνονται και σε πιο καθημερινές περιπτώσεις, όπως αυτό της έυρεσης της σωστής δοσολογίας ενός φαρμάκου (συχνό πρόβλημα για ασθενείς με προβλήματα θυρεοειδούς ή ψυχιατρικά προβλήματα).

Μια από τις πιο γνωστές και χαρακτηριστικές εφαρμογές τη σήμερον ημέρα είναι αυτό της τοποθέτησης/επιλογής διαφημίσεων (π.χ. σε αποτελέσματα μηχανής αναζητήσεως). Οι μηχανές αναζήτησης φιλοξενούν ένα μεγάλο αριθμό από διαφημιστές με αντάλλαγμα χρηματική αμοιβή, το μεγαλύτερο ποσοστό της οποίας προέρχεται από το λεγόμενο "click through rate", δηλαδή με λίγα λόγια, την αλληλεπίδραση των χρηστών με τη διαφήμιση. Επομένως κάθε μηχανή αναζήτησης έχει ώφελος να παρουσιάζει στον κάθε χρήστη μια διαφήμιση που θα μεγιστοποιεί της πιθανότητες αυτός να αλληλεπιδράσει μαζί της. Αυτό το πρόβλημα είναι επομένως πολύ όμοιο με το αρχικό πρόβλημα, αλλά παρ'όλα αυτά πιο περίπλοκο, καθώς ο κάθε χρήστης μπορεί να έχει διαφορετικά ενδιαφέροντα και χαρακτηριστικά. Οι εφαρμογές στην διαφήμιση και στις μηχανές αναζήτησης έχουν ωθήσει ένα μεγάλο ποσοστό της έρευνας στο πρόβλημα *MAB* τα τελευταία χρόνια.

0.2 Μοντέλο

Θα μοντελοποιήσουμε το πρόβλημα *MAB* ως εξης. Υπάρχουν k ενέργειες[1] τις οποίες μπορεί να επιλέξει ο πράκτορας (ή *learner*[2]). Ο πράκτορας έχει ένα συγκεκριμένο χρονικό διάστημα (διακριτό, μπορεί να θεωρηθεί ως ένα διάστημα γύρων) όπου επιλέγει κάθε φορά κάποια ενέργεια. Εφ'όσον επιλέξει κάποια ενέργεια, αυτή αποκαλύπτει την ανταμοιβή της σε αυτόν τον γύρο και ο πράκτορας την αποκτά. Ο στόχος του πράκτορα είναι να μεγιστοποιήσει τα κέρδη του μετά το πέρας των γύρων. Θα αναφερόμαστε σε κάποια ενέργεια με το σύμβολο a (από το arm) και στην ενέργεια που επιλέχθηκε στον γύρο t με a_t. Κάθε τέτοια ενέργεια μπορεί να θεωρηθεί ως ένας δείκτης στο σύνολο $[k] := \{1, 2, \ldots, k\}$. Ακόμα, συμβολίζουμε με $r_t(a)$ την ανταμοιβή της ενέργειας a στον γύρο t και με T το σύνολο των δεδομένων γύρων. Έχουμε λοιπόν το παρακάτω πρωτόκολλο/πρόβλημα:

Πρόβλημα 0.2.1. Multi Armed Bandit Problem (MAB)
1. Ο πράκτορας έχει στην διάθεση του k ενέργειες σε κάθε γύρο t και συνολικά T γύρους.
2. Σε κάθε γύρο t επιλέγει μια ενέργεια a_t.
3. Στον γύρο t λαμβάνει την ανταμοιβή $r_t(a_t)$ της ενέργειας a_t που επέλεξε.
4. Στόχος είναι η μεγιστοποίηση του κέρδους.

Έχουμε αμελήσει μια σημαντική λεπτομέρεια: πώς προκύπτουν οι ανταμοιβές; Είναι στοχαστικά τυχαίες, δηλαδή δείγματα μιας κατανομής; Αν ναι, αυτή η κατανομή είναι στατική στον χρόνο ή μεταβάλλεται; Αν δεν κάνουμε καμία υπόθεση πάρα ταύτα και αφήνουμε το ενδεχόμενο κάποιος ανταγωνιστής να τις παράγει; Αυτές και διάφορες άλλες κατευθύνσεις αποτελούν περιβάλλοντα αυτού του προβλήματος που έχουν μελετηθεί ενδελεχώς. Εμείς θα επικεντρωθούμε αρχικά στα δύο σημαντικότερα και πιο μελετημένα σενάρια.

Πρόβλημα 0.2.2. Στοχαστικό πρόβλημα MAB
Η ανταμοιβή $r_t(a)$ ακολουθεί μια στατική κατανομή D_a, με αναμενόμενη τιμή $\mu(a)$ και διασπορά σ_a^2 (άγνωστα στον πράκτορα). Καμία άλλη υπόθεση δεν γίνεται για τις κατανομές.

Πρόβλημα 0.2.3. Ανταγωνιστικό πρόβλημα MAB
Δεν υπάρχει καμία υπόθεση για τις ανταμοιβές $r_t(a)$. Θα μπορούσαν να δημιουργούνται ανταγωνιστικά από κάποιον εχθρό.

[1] bandits, arms ή actions στη βιβλιογραφία

[2] αυτός που *μαθαίνει*

0.3 Μετάνοια

Πώς μπορεί ο πράκτορας να αξιολογήσει την επιτυχία του; Δεν είναι δυνατό να μεγιστοποιήσει τα κέρδη του, σταθερά με μεγάλη πιθανότητα, ακόμα και στην φαινομενικά απλούστερη στοχαστική εκδοχή του προβλήματος, ακόμα και να ήξερε με κάποιον τρόπο την ενέργεια που αποδίδει καλύτερα κατά αναμενόμενη τιμή. Πρέπει να ορίσουμε ένα λογικό μέτρο επιτυχίας το οποίο θα εκτιμά την απόδοση, δεν φαίνεται λογικό το να συγκριθεί με την καλύτερη αλληλουχία ενεργειών. Η σύγκριση με έναν μάντη που ακολουθεί μια συγκεκριμένη ενέργεια σε κάθε γύρο γνωρίζοντας ποια θα επιφέρει το μέγιστο κέρδος θα δούμε ότι είναι εφικτή. Ορίζουμε λοιπόν την έννοια της *μετάνοιας* υπό αυτό το πρίσμα.

Ορισμός 0.3.1. Μετάνοια (Regret)
Ορίζουμε ως μετάνοια την ποσότητα:

$$R(T) := \sum_{t=1}^{T} r_t(a^*) - \sum_{t=1}^{T} r_t(a_t)$$

Όπου $a^* := \operatorname{argmax}_{a \in [k]} \sum_{t=1}^{T} r_t(a)$ *(η ”καλύτερη” ενέργεια)*

Ονομάζουμε αυτή την ποσότητα μετάνοια, καθώς αναπαριστά πόσο μετανιώνει ο πράκτορας μη επιλέγοντας την ”καλύτερη” ενέργεια. Αυτή είναι η συνηθισμένη έννοια μετάνοιας (regret) που συναντάται στην βιβλιογραφία. Υπάρχει μια άλλη μετρική, η οποία συχνά καλείται ψευδο-μετάνοια (pseudo-regret) και συναντάται στο στοχαστικό πρόβλημα MAB.

Ορισμός 0.3.2. Ψευδο-Μετάνοια (Pseudo-Regret)
Ορίζουμε ως ψευδο-μετάνοια την ποσότητα:

$$R(T) := \sum_{t=1}^{T} \mu^* - \sum_{t=1}^{T} \mathbb{E}\left[r_t(a_t)\right]$$

Όπου: $\mu^* := \operatorname{argmax} a \in [k] \mathbb{E}\left[r_t(a)\right]$ *(η καλύτερη αναμενόμενη ανταμοιβή)*

Μια άλλη πολύ χρήσιμη διατύπωση της παραπάνω ποσότητας είναι η επόμενη:

$$R(T) = \sum_{a \in [k]} n_T(a)\Delta(a)$$

Όπου:

$$n_T(a) := \sum_{i \in [T]} \mathbb{I}\{a_t = a\} \qquad \text{(Φορές που επιλέχθηκε η ενέργεια } a \text{ μέχρι τον γύρο } T\text{)}$$

$$\Delta(a) := \mu^* - \mu(a) \qquad \text{(το χάσμα (gap) της ενέργειας } a\text{)}$$

0.4 Στοχαστικό πρόβλημα MAB

Το πρώτο πρόβλημα που μελετάμε είναι η στοχαστική περίπτωση του MAB. Κάθε ανταμοιβή μιας ενέργειας a προέρχεται από μια στατική κατανομή D_a με αναμενόμενη τιμή $\mu(a)$. Συμβολίζουμε με μ^* την υψηλότερη τέτοια αναμενόμενη τιμή και με a^* την ενέργεια που αντιστοιχεί σε αυτή (ισοπαλίες επιλύονται αυθαιρέτως).

0.5 Εξερεύνηση πρώτα (ETC: Explore then Commit)

Η στοχαστικότητα των ανταμοιβών μας επιτρέπει να ισχυριστούμε με κάποια εμπιστοσύνη - έπειτα από έναν επαρκή αριθμό γύρων - ότι κάποια ενέργεια είναι "καλύτερη" ή "χειρότερη" από κάποια άλλη. Μια λογική, αλλά αφελής (όπως θα δούμε) σκέψη που εκμεταλλέυεται αυτό το γεγονός, οδηγεί στην διατύπωση του πρώτου αλγορίθμου που θα μελετήσουμε για το στοχαστικό πρόβλημα MAB. Ο αλγόριθμος διακρίνεται σε δύο στάδια, στο πρώτο στάδιο επιλέγει κάθε ενέργεια ένα συγκεκριμένο αριθμό φορών και έπειτα στο δεύτερο στάδιο επιμένει στην φαινομενικά καλύτερη. Πιο συγκεκριμένα, ορίζουμε τον παρακάτω αλγόριθμο που αποκαλούμε *Εξερεύνηση πρώτα (ETC)*.

Algorithm 1 Εξερεύνηση πρώτα (ETC)

```
for $t \in [k \cdot M]$ do                                ▷ Κάθε ενέργεια θα επιλεχθεί $M$ φορές
    $a_t = t \mod k$
Ορίζουμε ως $\bar{\mu}_a$ την $M$-δειγμάτων μέση τιμή της κατανομής $D_a$ από τα προηγούμενα
δείγματα.
for $t \in [T] \setminus [kM]$ do                ▷ Επιλογή της "καλύτερης" ενέργειας από εδώ και πέρα
    $a_t = \text{argmax}_{a \in [k]} \bar{\mu}_a$
```

Για την ανάλυση της μετάνοιας του αλγορίθμου είναι απαραίτητη η εκτίμηση της απόκλισης του εμπειρικού μέσου κάθε ανταμοιβής από την αναμενόμενη τιμή της, ώστε να επιχειρηματολογήσουμε για την μετάνοια ως προς την "καλύτερη" ανταμοιβή. Αυτό μπορεί να γίνει εφικτό μέσω μιας *ανισότητας σύγκεντρωσης μέτρου* (concentration inequality). Ίσως η γνωστότερη είναι αυτή του Hoeffding [38]:

Θεώρημα 0.5.1. *[38] Έστω $\{X_i\}_{i=1}^n$ μια ακολουθία ανεξάρτητων τυχαίων μεταβλητών η οποία καθεμία ανήκει στο $[a_i, b_i]$. Για κάθε $\delta \geq 0$:*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i\in[n]} X_i - E\left[\frac{1}{n}\sum_{i\in[n]} X_i\right]\right| \geq \delta\right) \leq 2exp\left\{-2\frac{n^2\delta^2}{\sum_{i\in[n]}(b_i - a_i)^2}\right\}$$

Ισοδύναμα έχουμε το παρακάτω υψηλής πιθανότητας *φράγμα:*

$$\mathbb{P}\left(\forall a \in [k] : \left|\frac{1}{n}\sum_{i=1}^{n} X_i(a) - \mu(a)\right| \leq \sqrt{\frac{1}{2n}\ln\frac{2k}{\delta}}\right) \geq 1 - \delta$$

Με τη βοήθεια της σημαντικής αυτής ανισότητας αποδεικνύουμε το παρακάτω θεώρημα για την μετάνοια του αλγορίθμου.

Θεώρημα 0.5.2. *Με πιθανότητα τουλάχιστον $1 - \delta$ ο αλγόριθμος* Εξερεύνηση Πρώτα *με $M = \left(\frac{T}{k}\right)^{2/3} \sqrt[3]{\ln(kT^2)}$ επιδεικνύει μετάνοια*

$$R(T) = O\left(T^{2/3}\sqrt[3]{k\ln\frac{k}{\delta}}\right)$$

Ακόμα, δείχνουμε ότι ο ίδιος αλγόριθμος επιτυγχάνει κατά τη μέση περίπτωση μετάνοια της τάξης $O\left(T^{2/3}\sqrt[3]{k\ln(kT)}\right)$. Παρ'ότι μια αναμενόμενη τιμή στην μετάνοια αποτελεί ένα καλό και ευκολοκατανόητο σημείο αναφοράς, μπορεί να ισχύει ότι γεγονότα με υψηλή μετάνοια αλληλοεξουδετερώνουν γεγονότα μικρής μετάνοιας. Για αυτό επιδιώκουμε εγγυήσεις *υψηλής πιθανότητας* για τους αλγορίθμους που παρουσιάζουμε.

0.6 Active Arm Elimination (AAE)

Ο αλγόριθμος *ETC* είναι ένας πολύ απλός αλγόριθμος που εγγυάται υπογραμμική μετάνοια, αλλά είναι μη βέλτιστος. Ο *ETC* είναι αυτό που αποκαλούμε *μη-προσαρμοστικός* αλγόριθμος, καθώς δεν προσαρμόζει ποτέ τις επιλογές του, όποιο και να είναι το περιβάλλον και η προϊστορία. Παρ'όλα αυτά, προϋποθέτοντας έναν επαρκή αριθμό γύρων ο πράκτορας μπορεί να εντοπίσει ότι κάποια ή κάποιες ενέργειες είναι αρκετά χειρότερες από άλλες με μεγάλη βεβαιότητα, αρκετά νωρίς στην ”εξερεύνηση”. Αυτή η παρατήρηση είναι η βάση του επόμενου αλγορίθμου.

Algorithm 2 Active Arm Elimination (AAE)

```
$S \leftarrow [k]$
for $t \in [T]$ do
    $CB(a;t) = \left[\bar{\mu}_t(a) - \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_t(a)}},\ \bar{\mu}_t(a) + \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_t(a)}}\right]$
    Αν υπάρχει ζευγάρι ενεργειών $a, a'$ με $UCB(a') < LCB(a)$ , τότε $S \leftarrow S \backslash \{a'\}$
    Επίλεξε την ενέργεια $a \in S$ που έχει επιλεχθεί τις λιγότερες φορές (ισοπαλίες επιλύονται αυθαίρετα)
```

Ο αλγόριθμος *AAE* ουσιαστικά λειτουργεί όπως ο *ETC* επιλέγοντας κάθε ενέργεια την μια μετά την άλλη, αλλά μόλις εντοπίσει ότι κάποια ενέργεια είναι με μεγάλη πιθανότητα αρκετά χειρότερη της άλλης την ”απενεργοποιεί” (δεν την επιλέγει ποτέ ξανά). Διαισθητικά αυτή η προσαρμοστικότητα πρέπει να οδηγεί σε καλύτερη εγγύηση μετάνοιας.

Κατ'αρχάς αποδεικνύουμε ότι το $CB(a;t)$ είναι όντως ένα διάστημα εμπιστοσύνης για την αναμενόμενη ανταμοιβή με μεγάλη πιθανότητα. Θα θέλαμε να χρησιμοποιήσουμε την ανισότητα του Hoeffding, αλλά αυτό δεν είναι εφικτό, καθώς τώρα πια ο αριθμός δειγμάτων για κάθε ενέργεια είναι στοχαστικός και εξαρτάται και από τις παρατηρήσεις. Παρ'όλα αυτά μπορούμε να χρησιμοποιήσουμε την ανισότητα *Azuma-Hoeffding* η οποία ισχύει για γενικότερες ακολουθίες τυχαίων μεταβλητών.

Θεώρημα 0.6.1. *Έστω Y_i μια ακολουθία MDS (Martingale Difference Sequence) με $Y_i \in [a_i, b_i]$ σχεδόν σίγουρα, τότε ισχύει:*

$$\mathbb{P}\left(\left|\sum_{i=1}^{t} Y_i(a)\right| \geq \epsilon\right) \leq 2\exp\left\{-\frac{2\epsilon^2}{\sum_{\tau=1}^{t}(b_i - a_i)^2}\right\}$$

Με την βοήθεια αυτής της ανισότητας και εξετάζοντας πότε επιλέχθηκε για τελευταία φορά κάποια μη βέλτιστη ενέργεια, αποδεικνύουμε ότι η μετάνοια του αλγορίθμου έχει την παρακάτω εγγύηση.

Θεώρημα 0.6.2. *Ο αλγόριθμος* AAE *επιδεικνύει την παρακάτω μετάνοια με πιθανότητα τουλάχιστον* $1 - \delta$*:*

$$R(T) = O\left(\ln \frac{kT}{\delta} \cdot \sum_{\substack{a\in[k]: \\ \mu(a)<\mu^*}} \frac{1}{\Delta(a)}\right)$$

και κατά τη μέση περίπτωση:

$$\mathbb{E}\left[R(T)\right] = O\left(\ln(kT) \cdot \sum_{\substack{a\in[k]: \\ \mu(a)<\mu^*}} \frac{1}{\Delta(a)}\right)$$

Το φράγμα παραπάνω αποτελεί ένα λεγόμενο *κατά-στιγμιότυπο-φράγμα* (instance independent bound), καθώς εξαρτάται από τα χάσματα των ενεργειών, άρα από το λεγόμενο *στιγμιότυπο*. Παρατηρούμε ότι αν το άθροισμα είναι της τάξης του $o(T^{2/3})$ έχουμε καλύτερη εγγύηση μετάνοιας από τον *ETC*.

Μπορούμε να δείξουμε το παρακάτω *γενικό φράγμα* (instance-independent bound) που ισχύει για οποιοδήποτε *στιγμιότυπο* του προβλήματος.

Θεώρημα 0.6.3. *Ο αλγόριθμος* AAE *επιδεικνύει την παρακάτω μετάνοια με πιθανότητα τουλάχιστον* $1 - \delta$*:*

$$R(T) = O\left(\sqrt{kT \cdot \ln \frac{kT}{\delta}}\right)$$

και κατά τη μέση περίπτωση:

$$\mathbb{E}\left[R(T)\right] = O\left(\sqrt{kT \cdot \ln(kT)}\right)$$

Είναι εμφανές λοιπόν ότι ο αλγόριθμος *AAE* έχει σημαντικά καλύτερη εγγύηση μετάνοιας από τον αφελή *ETC*.

0.7 Κάτω φράγματα μετάνοιας

Το παρακάτω θεώρημα δείχνει ότι ο αλγόριθμος *AAE* επιδίδει σχεδόν βέλτιστη εγγύηση μετάνοιας στη γενική περίπτωση.

Θεώρημα 0.7.1. *[4] Κανένας αλγόριθμος για το στοχαστικό πρόβλημα MAB δεν μπορεί να επιτύχει μετάνοια κατά τη μέση περίπτωση:*

$$\mathbb{E}\left[R(T)\right] = o\left(\sqrt{kT}\right)$$

Μάλιστα ακόμα και για κατά-περίπτωση μετάνοια ο *AAE* εξασφαλίζει σχεδόν βέλτιστη εγγύηση μετάνοιας [41]:

Θεώρημα 0.7.2. *Κανένας αλγόριθμος για το στοχαστικό πρόβλημα MAB δεν μπορεί να επιτύχει μετάνοια κατά τη μέση περίπτωση:*

$$\mathbb{E}\left[R(T)\right] = o(C \ln T)$$

όπου η σταθερά C εξαρτάται μόνο από το στιγμιότυπο *του προβλήματος (και όχι από τον χρονικό ορίζοντα T).*

0.8 Upper Confidence Bound (UCB)

Διατυπώνουμε άλλον έναν αλγόριθμο για το στοχαστικό πρόβλημα MAB ο οποίος, ίσως απρόσμενα, επιδεικνύει τις ίδιες εγγύησεις με τον *AAE* για την μετάνοια. Ο αλγόριθμος είναι αρκετά απλός.

Algorithm 3 UCB

```
for $t \in [T]$ do
    Επίλεξε την ενέργεια $a$ που μεγιστοποιεί το $UCB(a;t) := \bar{\mu}_t(a) + \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_t(a)}}$
```

Πριν παρουσιάσουμε τις εγγύησεις του αλγορίθμου, θα προσπαθήσουμε να ανακαλύψουμε διαισθητικά γιατί "δουλεύει". Όπως είχαμε δει προηγουμένως το διάστημα παρακάτω αποτελεί ένα $1-\delta$ διάστημα εμπιστοσύνης για την πραγματική αναμενόμενη ανταμοιβή $\mu(a)$:

$$[LCB(a;t), UCB(a;t)]$$

Όπου ορίζουμε το UCB όπως παραπάνω και $LCB(a;t) := \bar{\mu}_t(a) - \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_t(a)}}$
Ένα υψηλό άνω φράγμα εμπιστοσύνης (UCB) μπορεί να είναι αποτέλεσμα δύο παραγόντων: είτε η ενέργεια έχει αρκετά υψηλή ανταμοιβή στη μέση περίπτωση, είτε η ενέργεια δεν έχει επιλεχθεί αρκετές φορές. Ο πράκτορας έχει κίνητρο να επιλέξει την ενέργεια και στις δύο προαναφερθείσες περιπτώσεις. Σημειώνουμε ότι ο UCB ανήκει σε μια μεγαλύτερη οικογένεια αλγορίθμων (ή πρωτοκόλλων αποφάσεων) που ακολουθούν την φιλοσοφία της *αισιοδοξίας υπό αβεβαιότητα.*
Δείχνουμε ότι ο αλγόριθμος *UCB* επιτυγχάνει τις επόμενες εγγυήσεις μετάνοιας με μεγάλη πιθανότητα.

Θεώρημα 0.8.1. *Ο UCB επιτυγχάνει μετάνοια κατά-στιγμιότυπο*

$$R(T) = O\left(\ln \frac{kT}{\delta} \cdot \sum_{\substack{a \in [k]: \\ \mu(a) < \mu^*}} \frac{1}{\Delta(a)}\right)$$

με πιθανότητα τουλάχιστον $1-\delta$
Ακόμα, επιτυγχάνει ασχέτως στιγμιοτύπου μετάνοια

$$R(T) = O\left(\sqrt{kT \cdot \ln \frac{kT}{\delta}}\right)$$

με πιθανότητα τουλάχιστον $1-\delta$.

0.9 UCB-V (UCB με εκτίμηση διασποράς)

Οι αποφάσεις όλων των προηγούμενων αλγορίθμων καθοδηγούνταν μόνο από την εμπειρική εκτίμηση της αναμενόμενης ανταμοιβής (και κατ'επέκταση του διαστήματος εμπιστοσύνης γύρω από αυτήν). Οι Audibert κ.α. στο [3] διατυπώνουν μια σημαντική παρατήρηση, ο πράκτορας θα πρέπει να έχει μεγαλύτερη εμπιστοσύνη στην εμπειρική μέση τιμή για ενέργειες που συμπεριφέρονται σχετικά στατικά (δηλαδή οι παρατηρούμενες ανταμοιβές δεν αποκλινουν πολύ μεταξύ τους), επομένως θα μπορεί να αποκλείσει κάποιες ενέργειες αρκετά πιο γρήγορα, αν έχουν μικρή διασπορά. Με λίγα λόγια, μια σχετικά χαμηλή διασπορά θα πρέπει να οδηγεί σε μικρότερο διάστημα εμπιστοσύνης γύρω από την εμπειρική τιμή. Αυτή η παρατήρηση αποδίδεται στον αλγόριθμο $UCB - V$.

Algorithm 4 UCB-V

```
for $t \in [T]$ do
    Επίλεξε την ενέργεια $a$ που μεγιστοποιεί το $UCBV(a;t) := \bar{\mu}_t(a) + \sqrt{8V_t(a) \cdot \ln\frac{4kT}{\delta} \cdot \frac{1}{n_t(a)}} + \ln\frac{4kT}{\delta} \cdot \frac{13}{n_t(a)}$
```

Παραπάνω η τιμή $V_t(a)$ είναι μια εμπειρική εκτίμηση της διασποράς. Συγκεκριμένα στην απόδειξη μας για την εγγύηση μετάνοιας του αλγορίθμου χρησιμοποιούμε την παρακάτω εκτίμηση:

Ορισμός 0.9.1. *Ορίζουμε την εκτίμηση διασποράς ως:*

$$V_t(a) := \frac{1}{2\lfloor n_t(a)/2 \rfloor} \sum_{i=1}^{\lfloor n_t(a)/2 \rfloor} U_i(a)$$

Όπου

$$U_i(a) := (X_{2i-1}(a) - X_{2i}(a))^2$$

και $X_i(a)$ είναι η ανταμοιβή την i-στη φορά που επιλέγεται η ενέργεια a από τον αλγόριθμο.

Δείχνουμε ότι η παραπάνω ποσότητα οδηγεί σε μια μη-μεροληπτική εκτίμηση της διασποράς, δηλαδή:

$$\mathbb{E}\left[V_t(a) = \sigma_a^2\right]$$

Μέσω ανισοτήτων για MDS ακολουθίες αποδεικνύουμε την παρακάτω ανισότητα συγκέντρωσης μέτρου:

Λήμμα 0.9.1.

$$|\bar{\mu}_t(a) - \mu(a)| \leq 2\sqrt{\frac{1}{n_t(a)}\sigma_a^2 \cdot \log\frac{2kT\log T}{\delta_1}} + \ln\frac{2kT\log T}{\delta_1} \cdot \frac{4}{3n_t(a)}$$

για όλες τις ενέργειες και χρόνους t με πιθανότητα τουλάχιστον $1 - \delta_1$, όπου $\bar{\mu}_t(a) := \frac{1}{n_t(a)} \sum_{i=1}^{n_t(a)} X_i(a)$ και $X_i(a)$ είναι η ανταμοιβή την i-στή φορά που επιλέγεται η ενέργεια a.

Την παραπάνω ανισότητα χρησιμοποιούμε και για την εκτίμηση της διασποράς, ώστε να δείξουμε πολλαπλασιαστική προσέγγιση όταν η πραγματική διασπορά είναι αρκετά μικρή (το οποίο οδηγεί σε ένα πραγματικό άνω φράγμα εμπιστοσύνης). Όταν η διασπορά είναι αρκετά μικρή, τότε δεν μπορούμε να έχουμε τέτοια προσέγγιση, αλλά καθώς το άνω φράγμα αποτελείται από έναν πρόσθετο ανεξάρτητο της εκτίμησης διασποράς όρο, εξασφαλίζουμε ένα άνω φράγμα εμπιστοσύνης και σε αυτήν την περίπτωση. Μάλιστα, στην δεύτερη περίπτωση δείχνουμε ότι η εκτίμηση της διασποράς δεν μπορεί να είναι αρκετά μεγάλη, ώστε κάποια ενέργεια να επιλεχθεί δυσανάλογα πολλές φορές.

Συνοψίζοντας, αποδεικνύουμε με αρκετά πιο προσεγγίσιμη ανάλυση από την αρχική δημοσίευση την παρακάτω εγγύηση για τον αλγόριθμο *UCB-V*:

Theorem 0.9.1. UCB-V *επιτυγχάνει μετάνοια*

$$O\left(\ln\frac{kT}{\delta}\cdot\sum_{\substack{a\in[k]:\\ \mu(a)<\mu^*}}\left\{\frac{\sigma_a^2}{\Delta(a)}+1\right\}\right)$$

με πιθανότητα τουλάχιστον $1-\delta$

0.10 Ανταγωνιστικό πρόβλημα MAB (Adversarial MAB problem)

Μελετήσαμε το στοχαστικό πρόβλημα MAB και είδαμε διάφορους αλγόριθμους για την αντιμετώπιση του. Σε αυτό το σημείο ανακαλούμε την υπόθεση της στοχαστικότητας των ανταμοιβών. Όταν δεν έχουμε υποθέσεις, μπορούμε κάλλιστα να θεωρήσουμε ότι υπάρχει κάποιος ανταγωνιστής που δημιουργεί τις ανταμοιβές με σκοπό να μας βλάψει. Το πρόβλημα αυτό μελετήθηκε για πρώτη φορά από τους Auer κ.α. στο [6].
Είναι εύκολο να δούμε ότι ένας απλός αλγόριθμος για το στοχαστικό πρόβλημα, όπως ο *ETC* μπορεί πάντα να οδηγηθεί σε γραμμική μετάνοια. Αρκεί να θεωρήσουμε έναν ανταγωνιστή που στο πρώτο στάδιο του αλγορίθμου δημιουργεί ανταμοιβές 0 για κάθε ενέργεια και την μέγιστη (1) για μια συγκεκριμένη, ενώ στο δεύτερο κάνει ακριβώς το αντίστροφο. Γενικότερα, κάθε ντετερμινιστικός αλγόριθμος δεν μπορεί να εγγυηθεί υπογραμμική μετάνοια κατά τη μέση περίπτωση υπό ανταγωνιστικές συνθήκες. Αυτό μπορεί να συμβεί ακόμα και όταν ο ανταγωνιστής δεν "αντιδρά" στις επιλογές του αλγορίθμου/πράκτορα, σε αυτή την περίπτωση καλούμε τον ανταγωνιστή *μη-προσαρμοστικό* (oblivious). Αν ο ανταγωνιστής αντιδρά στις επιλογές του πράκτορα καλείται *προσαρμοστικός* (adaptive). Ταυτόχρονα και στις δύο άνω περιπτώσεις ο πράκτορας μπορεί να είναι είτε ντετερμινιστικός, είτε τυχαιοκρατικός.
Η μετάνοια κάποιου πράκτορα/αλγορίθμου ορίζεται με βάση την εκ των υστέρων καλύτερη ενέργεια:

$$a^* := \underset{a\in[k]}{\text{argmax}}\sum_{t=1}^{T} r_t(a)$$

Ίσως απρόσμενα, θα αποδείξουμε ότι υπάρχει αλγόριθμος που επιτυγχάνει εγγύηση μετάνοιας σχεδόν όσο το βέλτιστο ασχέτως-στιγμιοτύπου θεωρητικό κάτω φράγμα για το στοχαστικό πρόβλημα! Συγκεκριμένα θα δούμε ότι:

Θεώρημα 0.10.1. *Υπάρχει αλγόριθμος για το ανταγωνιστικό πρόβλημα MAB που εγγυάται μετάνοια:*

$$\mathbb{E}\left[R(T)\right]=O\left(\sqrt{kTlogk}\right)$$

Η μετάνοια αυτή είναι σχεδόν βέλτιστη, όπως δείχνει το παρακάτω θεώρημα.

Θεώρημα 0.10.2. *[6] Οποιοσδήποτε αλγόριθμος για το ανταγωνιστικό πρόβλημα MAB έχει αναμενόμενη μετάνοια:*

$$\mathbb{E}\left[R(T)\right] \geq \Omega(\sqrt{kT})$$

Οι Auer κ.α. παρουσίασαν τον αλγόριθμο *Exp3* για το ανταγωνιστικό πρόβλημα MAB το οποίο όπως αναφέραμε οι ίδιοι μελέτησαν για πρώτη φορά. Ο αλγόριθμος είναι επηρεασμένος από τον γνωστό αλγόριθμο Hedge (ή Multiplicative Weights Update) ο οποίος παρουσιάστηκε πρώτη φορά από τους Freund and Schapire [32]. Ο αλγόριθμος ακολουθεί παρακάτω.

Algorithm 5 Exp3

```
Παράμετροι $\gamma, \eta \in (0, 1/2)$
Για όλες τις ενέργειες $a \in k$ : $w_1(a) = 1$
for $t \in [T]$ do
    Επιλογή ενέργειας $a$ με πιθανότητα $p_t(a) = (1-\gamma)\frac{w_t(a)}{\sum_{\tilde{a}\in[k]} w_t(\tilde{a})} + \gamma/k$
    Υποδοχή $r_t(a)$
    $\hat{r}_t(a) = \mathbb{I}\{a_t = a\} \cdot \frac{r_t(a)}{p_t(a)}$
    $w_{t+1}(a) = w_t(a) \cdot \exp\{\eta \hat{r}_t(a)\}$
```

Θεώρημα 0.10.3. *[6] Ο αλγόριθμος Exp3 επιτυγχάνει την παρακάτω εγγύηση για την αναμενόμενη μετάνοια, για οποιονδήποτε ανταγωνιστή.*

$$\mathbb{E}\left[R(T)\right] \leq 2.63\sqrt{Tk\ln k}$$

0.11 Στοχαστικό πρόβλημα MAB υπό ανταγωνιστικές αλλοιώσεις

Μέχρι τώρα έχουμε μελετήσει το στοχαστικό πρόβλημα MAB, αλλά και το αντίστοιχο ανταγωνιστικό πρόβλημα. Παρ'όλα αυτά οι δύο αυτές περιπτώσεις μπορεί να αποτελούν ακραίες υποθέσεις για το περιβάλλον του εκάστοτε προβλήματος, από τη μία η υπερβολικά ελπιδοφόρα θεώρηση ότι οι ανταμοιβές προκύπτουν από όμοια ανεξάρτητα δείγματα της ίδιας κατανομής και από την άλλη η υπερβολικά πεσιμιστική θεώρηση ότι πρέπει να φυλαχτούμε από κάποιον ανταγωνιστή με κακές προθέσεις.

Οι Λυκούρης κ.α. πρότειναν το πρόβλημα των *Στοχαστικών MAB υπό ανταγωνιστικές αλλοιώσεις* στο [46] για περιβάλλοντα όπου η εγγενής στοχαστικότητα διαβάλλεται από αλλοιώσεις που δεν οδηγούν σε εντελώς ανταγωνιστικό περιβάλλον. Η μελέτη τέτοιων περιβάλλοντων παρακινήθηκε περαιτέρω από το συνεχώς αυξανόμενο φαινόμενο του *click fraud*, σε προτάσεις μηχανών αναζήτησης, δηλαδή την οργανωμένη επιχείρηση καθοδήγησης προτάσεων, μέσω ψευδών δηλώσεων ενδιαφέροντος. Άλλες εφαρμογές ενδιαφέροντος αποτελούν τα *spam* και οι κακόβουλες κριτικές σε εφαρμογές προτάσεων.

Ορίζουμε το παρακάτω πρωτόκολλο/πρόβλημα:

Πρόβλημα 0.11.1. Στοχαστικά MAB υπό ανταγωνιστικές αλλοιώσεις

1. Ο πράκτορας έχει στην διάθεση του k ενέργειες σε κάθε γύρο t και συνολικά T γύρους.

2. Σε κάθε γύρο t επιλέγει μια ενέργεια a_t με στοχαστική ανταμοιβή $\tilde{r}_t(a_t)$.

3. Στον γύρο t ο ανταγωνιστής αλλοιώνει την ανταμοιβή κατά $c_t(a_t)$.
4. Ο πράκτορας παρατηρεί την ανταμοιβή $r_t(a_t) = \tilde{r}_t(a_t) + c_t(a_t)$ 5. Στόχος είναι η ελαχιστοποίηση της ψευδο-μετάνοιας.

Ακόμα, το ποσό αλλοίωσης ή το πόσο κοντά είναι το πρόβλημα στο να είναι πλήρως ανταγωνιστικό ορίζεται από την *σωρευτική αλλοίωση C.*

Ορισμός 0.11.1. *Η σωρευτική αλλοίωση C ορίζεται ως:*

$$C := \sum_{t\in[T]} \max_{a\in[k]} |c_t(a)|$$

Οι στοχαστικοί αλγόριθμοι όπως ο *AAE* μπορούν να οδηγηθούν σε γραμμική μετάνοια στο παραπάνω πρόβλημα ακόμα και με ένα πολύ μικρό ποσό σωρευτικής αλλοίωσης (λογαριθμικό ως προς τον χρονικό ορίζοντα). Από την άλλη, αλγόριθμοι για το ανταγωνιστικό πρόβλημα, όπως ο *Exp3* διατηρούν υπογραμμικές εγγυήσεις, αλλά δεν εκμεταλλεύονται την εγγενή στοχαστικότητα, ώστε να εξασφαλίσουν καλύτερες εγγυήσεις όταν η αλλοίωση είναι σχετικά μικρή.

0.12 Multi-Layer AAE Race

Οι Λυκούρης κ.α. τονίζουν ότι αν ο πράκτορας γνώριζε την αλλοίωση C, ή τουλάχιστον κάποιο άνω φράγμα, τότε θα μπορούσε αυξάνοντας το διάστημα εμπιστοσύνης του *AAE* για την κάθε ενέργεια, να λειτουργεί όπως αναμενόταν σε πλήρη στοχαστικό περιβάλλον, δηλαδή με μεγάλη πιθανότητα να μην αποκλειστεί η βέλτιστη ενέργεια. Συγκεκριμένα αποδεικνύουν ότι:

Θεώρημα 0.12.1. *Ο αλγόριθμος* AAE *με διάστημα εμπιστοσύνης*

$$rad(a;t) = \sqrt{\frac{2\ln 2kT/\delta}{n_t(a)} + \frac{C}{n_t(a)}}$$

έχει μετάνοια της τάξης $O\left(\sum_{a\neq a^*} \frac{\ln\frac{kT}{\delta}+C}{\Delta(a)}\right)$, αν C είναι ένα άνω φράγμα για την σωρευτική αλλοίωση.

Στη συνέχεια αποδεικνύουν το επόμενο σημαντικό θεώρημα εν πορεία για την παρουσίαση του αλγορίθμου για το αρχικό πρόβλημα.

Θεώρημα 0.12.2. *Αν το περιβάλλον είναι είτε στοχαστικό είτε αλλοιωμένο με σωρευτική αλλοίωση C, τότε ο παρακάτω αλγόριθμος επιτυγχάνει μετάνοια της τάξης $O\left(\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$ στην πρώτη περίπτωση και μετάνοια της τάξης $O\left(kC\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$ στην δεύτερη.*

Algorithm 6 Αργό-Γρήγορο ΑΑΕ

```
for $t \in [T]$ do
    $rd_F(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^F(a)}}$
    $rd_S(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^S(a)}} + \frac{2\ln\frac{2kT}{\delta}}{n_t^S(a)}$
    Με πιθανότητα $1/C$ χρησιμοποίησε το αργό ΑΑΕ ($S$), αλλιώς χρησιμοποίησε το γρήγορο ($F$).
    Αν ο $S$ αποκλείσει μια ενέργεια an $a$, τότε η ίδια ενέργεια αποκλείεται και στον $F$.
    Αν ο $F$ δεν διαθέτει πια ενέργειες προς επιλογή, επίλεξε μια ενέργεια στην τύχη από αυτές που δεν έχουν αποκλειστεί από τον $S$. ▷ Χωρίς να ανανεωθεί/αλλάξει κάποια πληροφορία στον S
```

Βασισμένοι λοιπόν στην προηγούμενη παρατήρηση, επεκτείνουν την ιδέα όταν η αλλοίωση δεν είναι γνωστή. Ο αλγόριθμος *Multi-Layer Active Arm Elimination Race* που παρουσιάζουν συντελείται από λογαριθμικά πολλά στιγμιότυπα *AAE*, καθένα από το οποίο είναι πιο ”αργό” από το προηγούμενο του. Συγκεκριμένα, το πρώτο επίπεδο είναι το πιο γρήγορο, αλλά και το πιο επιρρεπές σε λάθη, ενώ κάθε επόμενο επίπεδο επιλέγεται με πιθανότητα που μειώνεται εκθετικά (καθένα μπορεί να χειριστεί αλλοίωση ανάλογη με το αντίστροφο της πιθανότητας αυτής). Ο αλγόριθμος φαίνεται παρακάτω.

Algorithm 7 Ο αλγόριθμος Multi-layer Active Arm Elimination Race

```
for $t \in [T]$ do
    $rad_0(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^0(a)}}$
    $rad_l(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^l(a)}} + \frac{2\ln\frac{2kT}{\delta}}{n_t^l(a)}$ for $l \in \{1, \ldots, \lfloor \log T \rfloor\}$
    Με πιθανότητα $(1/2)^{l+1}$ επίλεξε τον αλγόριθμο του επιπέδου $l$ ή με την υπολοιπόμενη πιθανότητα τον αλγόριθμο του επιπέδου $l = 0$.
    Αν το επίπεδο $l$ αποκλείσει μια ενέργεια $a$, απόκλεισε την ίδια ενέργεια σε όλα τα επίπεδα $l' < l$.
    Αν κάποιο επιλεγμένο επίπεδο δεν έχει ενέργειες προς επιλογή, επίλεξε μια ενέργεια από το κοντινότερο άνω επίπεδο το οποίο έχει ενέργειες προς επιλογή.
```

Ο παραπάνω αλγόριθμος δεν προϋποθέτει την γνώση της αλλοίωσης ή τη γνώση του αν το περιβάλλον πρόκειται να είναι εντελώς στοχαστικό ή αλλοιωμένο. Μάλιστα, επιτυγχάνει αυτό που ζητείται, καθώς εξασφαλίζει την βέλτιστη εγγύηση μετάνοιας σε πλήρως στοχαστικό περιβάλλον η οποία αυξάνεται ομαλά όσο αυξάνεται η αλλοίωση, όπως δείχνει το παρακάτω θεώρημα.

Θεώρημα 0.12.3. *Ο αλγόριθμος Multi-Layer Active Arm Elimination Race επιτυγχάνει μετάνοια*

$$O\left(\ln\frac{kT}{\delta}\sum_{a\neq a^*}\frac{kC\ln\frac{kT}{\delta}+\log T}{\Delta(a)}\right)$$

με πιθανότητα τουλάχιστον $1-\delta$.

0.13 Ο αλγόριθμος BARBAR

Παρ'όλο που ο αλγόριθμος *MLAEER* είναι διαισθητικός και απλός, η εγγύηση μετάνοιας που υπόσχεται διαθέτει πολλαπλασιαστική εξάρτηση από την αλλοίωση και μπορεί να οδηγηθεί σε γραμμική μετάνοια ακόμα και για αλλοίωση $C = \Omega(\sqrt{T})$. Οι Gupta κ.α. παρουσιάζουν έναν αλγόριθμο για τον οποίο δείχνουν ότι εξασφαλίζει ένα αισθητά καλύτερο φράγμα στη μετάνοια, ουσιαστικά ανταλλάσοντας την πολλαπλασιαστική εξάρτηση με αθροιστική. Ο αλγόριθμος αυτός επίσης δεν χρειάζεται γνώση της αλλοίωσης και είναι σχετικά απλός.

Το κύριο συστατικό του αλγορίθμου είναι ότι χρησιμοποιεί λογαριθμικά πολλές εποχές, οι οποίες χρησιμοποιούν πληροφορίες μόνο από την αμέσως προηγούμενη εποχή (το οποίο εξασφαλίζει κατά κάποιο τρόπο μια φραγμένη επιρροή της αλλοίωσης). Συγκεκριμένα, κάθε εποχή m διατηρεί μια εκτίμηση $\Delta_m(a)$ για το χάσμα $\Delta(a)$ της ενέργειας a και παίζει με πιθανότητα τέτοια, ώστε κατά αναμενόμενη τιμή, η ενέργεια a να επιλεγεί όσες φορές θα επιλεγόταν (το πολύ) από τον αλγόριθμο *UCB*, αν όντως είχε χάσμα $\Delta_m(a)$. Μια απαίτηση είναι καμία ενέργεια να μην επιλεγεί πάνω από 2^{2m} φορές, το οποίο μαζί με την πιθανοτική επιλογή δίνει ευκαιρίες σε φαινομενικά κακές ενέργειες. Ο αλγόριθμος ακολουθεί.

Algorithm 8 BARBAR

```
$\lambda = 1024 \ln\left(\frac{8k}{\delta}\log T\right)$
$T_0 = 1$
for $a \in [k]$ do
    $\Delta_0(a) = 1$
for $m \in \{1, 2, ...\}$ do
    $\bar{n}_m(a) = \lambda \cdot \Delta_{m-1}^{-2}(a)$
    Θέσε $N_m = \sum_{a \in [k]} \bar{n}_m(a)$ και $q_m(a) = \frac{\bar{n}_m(a)}{N_m}$
    Θέσε $T_m = T_{m-1} + N_m$ και $E_m = [T_{m-1}, T_m]$
    for $t \in E_m \cap [T]$ do
        επίλεξε την ενέργεια $a$ με πιθανότητα $q_m(a)$
    $\mu_m(a) = \frac{1}{\bar{n}_m(a)} \sum_{t \in E_m} \mathbb{I}\{a_t = a\} r_t(a)$
    $a_m^* = \mathrm{argmax}_{a \in [k]} \{\mu_m(a) - \frac{1}{16}\Delta_{m-1}(a)\}$
    $\mu_m^* = \mu_m(a_m^*) - \frac{1}{16}\Delta_{m-1}(a)$
    $\Delta_m(a) = \max\{2^{-m}, \mu_m^* - \mu_m(a)\}$
```

Αρχικά οι Gupta κ.α. δείχνουν το επόμενο κύριο λήμμα το οποίο περιέχει την ανισότητα συγκέντρωσης μέτρου για την απόκλιση της εκτίμησης $\mu_m(a)$ από την πραγματική αναμενόμενη ανταμοιβή:

Λήμμα 0.13.1. *Έστω*

$$\mathcal{E} := \left\{ \forall m, i : |\mu_m(a) - \mu(a)| \leq \frac{2C_m}{N_m} + \frac{\Delta_{m-1}(a)}{16} \text{ και } n_m(a) \leq 2\bar{n}_m(a) \right\}$$

. Τότε ισχύει ότι: $\mathbb{P}(\mathcal{E}) \geq 1 - \delta$

Αυτή η ανισότητα οδηγεί στο επόμενο λήμμα για τις εκτιμήσεις του χάσματος κάθε ενέργειας.

Λήμμα 0.13.2.

$$\Delta_m(a) \leq 2(\Delta(a) + 2^{-m} + \rho_m)$$

και

$$\Delta_m(a) \geq \frac{1}{2}\Delta(a) - \frac{3}{4}2^{-m} - 3\rho_m$$

όπου $\rho_m := \sum_{e=1}^{m} \frac{2C_e}{8^{m-s}N_e}$

Παρατηρούμε ότι η αλλοίωση προηγούμενων εποχών συνεχίζει να επηρεάζει και επόμενες, αλλά αυτή η επιρροή μειώνεται εκθετικά με την πάροδο των εποχών. Με τη βοήθεια του προηγούμενου λήμματος αποδεικνύουν το επόμενο κύριο θεώρημα και εγγύηση μετάνοιας του αλγορίθμου.

Θεώρημα 0.13.1. *Ο αλγόριθμος* BARBAR *επιτυγχάνει μετάνοια*

$$R(T) \leq O\left(kC + \log(T) \cdot \log\left(\frac{k}{\delta}\log T\right) \sum_{a \neq a^*} \frac{1}{\Delta(a)}\right)$$

με πιθανότητα τουλάχιστον $1 - \delta$.

0.14 Εκτιμήσεις Διασποράς σε αλλοιωμένο περιβάλλον

Ο αλγόριθμος *BARBAR* προσομοιώνει την συμπεριφορά των αλγορίθμων *UCB, AAE* με εύρωστο τρόπο όταν υπάρχουν αλλοιώσεις. Σε αυτό το σημείο μελετάμε αν μπορούμε να επιτύχουμε παρόμοια συμπεριφορά για τον αλγόριθμο *UCBV* χρησιμοποιώντας εκτιμήσεις διασποράς. Διατυπώνουμε λοιπόν τον παρακάτω αλγόριθμο.

Algorithm 9 BARBAR-V

```
$\lambda = 2^12\ln\frac{10k\log^2 T}{\delta}$
$T_0 = 1$
for $a \in [k]$ do
    $\Delta_0(a) = 1$
    $V_0(a) = \frac{1}{4}$                                   ▷ Η μέγιστη διασπορά Τ.Μ. στο [0, 1] είναι 0.25.
for $m \in \{1, 2, ...\}$ do
    $\bar{n}_m(a) = \lambda\left(\frac{V_{m-1}(a)}{\Delta^2_{m-1}(a)} + \frac{1}{\Delta_{m-1}(a)}\right)$   ▷ Μια ενέργεια $a$ θα επιλεχθεί περίπου τόσες φορές.
    $N_m = \sum_{a\in[k]} \bar{n}_m(a)$
    $q_m(a) = \frac{\bar{n}_m(a)}{N_m}$
    $T_m = T_{m-1} + N_m$
    $E_m = [T_{m-1}, T_m]$
    for $t \in E_m \cap [T]$ do
        Επίλεξε την ενέργεια $a$ με πιθανότητα $q_m(a)$
    $\mu_m(a) =$ εμπειρική μέση ανταμοιβή $a$ την εποχή $m$
    $V_m(a) =$ εμπειρική διασπορά ανταμοιβής $a$ την εποχή $m$
    $a^*_m = \text{argmax}_{a\in[k]}\{\mu_m(a) - \frac{1}{16}\Delta_{m-1}(a) - \frac{1}{128}\Delta_{m-2}(a)\}$
    $\mu^*_m = \mu_m(a^*_m) - corr_m(a^*_m)$
    $\Delta_m(a) = \max\{2^{-m}, \mu^*_m - \mu_m(a)\}$
```

Ένα σημαντικό τεχνικό λήμμα για την ανάλυση του αλγορίθμου είναι το παρακάτω αποτέλεσμα για την εμπειρική εκτίμηση της διασποράς.

Λήμμα 0.14.1. *Για κάθε ενέργεια a και για κάθε εποχή m με πιθανότητα τουλάχιστον $1 - 3k\log^2 Te^{-x}$:*

Όταν $\sigma^2_a \geq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$ (υψηλή διασπορά):

$$-\frac{48C_m}{N_m} + \frac{1}{4}\sigma^2_a \leq V_m(a) \leq 4\sigma^2_a + \frac{96C_m}{N_m}$$

Όταν $\sigma^2_a \leq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$ (χαμηλή διασπορά):

$$V_m(a) \leq \frac{245x}{\lambda} \cdot \Delta_{m-1}(a) + \frac{96C_m}{N_m}$$

Αυτό μας οδηγεί στο παρακάτω λήμμα που ποσοτικοποιεί την εκτίμηση του χάσματος κάποιας ενέργειας, με βάση την διασπορά και την εποχή. Διαπιστώνουμε ότι λαμβάνουμε ουσιαστικά το αντίστοιχο λήμμα που διατύπωσαν οι Gupta κ.α. στο [36], με πολύ μικρές διαφορές.

Λήμμα 0.14.2. *Για κάθε εποχή m και ενέργεια a έχουμε με πιθανότητα τουλάχιστον $1 - \delta$:*

$$\Delta_m(a) \leq 2(\Delta(a) + 2^{-m} + \rho_m)$$

Όπως και:

$$\Delta_m(a) \geq \frac{1}{2}\Delta(a) - 2^{-m} - 6\rho_m$$

Όπου ορίζουμε:

$$\rho_m := \sum_{s=1}^{m} \frac{1}{8^{m-s-1}} \frac{C_s}{N_s}$$

Χρησιμοποιώντας το προηγούμενο λήμμα και διαχωρίζοντας περιπτώσεις για την πραγματική διασπορά κάθε ενέργειας, αποδεικνύουμε το παρακάτω θεώρημα για την εγγύηση του αλγορίθμου.

Θεώρημα 0.14.1 (Μετάνοια σε στοχαστικό περιβάλλον). *Ο αλγόριθμος BARBAR-V επιτυγχάνει την επόμενη εγγύηση μετάνοιας με πιθανότητα τουλάχιστον* $1-\delta$*, σε στοχαστικό περιβάλλον:*

$$R(T) \leq O\left(\ln \frac{k \log^2 T}{\delta} \cdot \sum_{a \neq a^*} \log T \left\{ \frac{\sigma_a^2}{\Delta(a)} + 1 \right\}\right)$$

Παρ'ότι το παραπάνω θεώρημα είναι ενθαρρυντικό, δυστυχώς η παρακάτω οικογένεια αντιπαραδειγμάτων δείχνει ότι ο αλγόριθμος δεν μπορεί να εγγυηθεί τα ζητούμενα φράγματα μετάνοιας σε αλλοιωμένο περιβάλλον.

Θεώρημα 0.14.2. *Αλλοίωση της τάξης* $\Theta(\sqrt{T})$ *μπορεί να οδηγήσει τον αλγόριθμο* BARBARV *σε μετάνοια της τάξης* $\Omega(T)$.

Απόδειξη.

Έστω ένα στιγμιότυπο δύο ενεργειών με την μη-βέλτιστη ενέργεια να επιδεικνύει χάσμα Δ. Έστω ακόμα $M = \log_4 T$. Θεωρούμε έναν ανταγωνιστή ο οποίος αλλοιώνει την εποχή $M/2 - 2$ με τέτοιο τρόπο, ώστε ο αλγόριθμος θεωρεί ότι οι δύο ενέργειες είναι ντετερμινιστικές. Ο ανταγωνιστής έχει την ίδια συμπεριφορά για όλες τις επόμενες εποχές μέχρι την εποχή $M-2$ (σημειωτέον: δε μας ενδιαφέρει ποια ενέργεια πιστεύει ο αλγόριθμος ότι είναι βέλτιστη).

Οι εποχές αυτές $m \in \{M/2-2, M/2-1, \ldots, M-2\}$ έχουν μήκος 2^m. Ο ανταγωνιστής στην εποχή $M-2$ αλλοιώνει με τέτοιο τρόπο ώστε ο αλγόριθμος να πιστεύει ότι η βέλτιστη ενέργεια είναι η μη βέλτιστη και μάλιστα ότι έχει μεγάλη (σταθερή) διασπορά. Η επόμενη εποχή τότε θα έχει μήκος 4^{M-1}.

Η συνολική αλλοίωση είναι:

$$\Theta(4^{M/2}) + \Theta\left(2^{M/2} + 2^{M/2+1} + \ldots + 2^{M-2}\right) = \Theta(\sqrt{T})$$

Ενώ ο αλγόριθμος υποκύπτει σε μετάνοια:

$$\Omega(\Delta \cdot 4^{M-1}) = \Omega(\Delta \cdot T)$$

Η οποία είναι γραμμική, αν το χάσμα Δ είναι σταθερό.

0.15 Επίλογος

Η φυσιολογική επέκταση του αλγορίθμου *BARBAR* για την χρήση εκτιμήσεων διασποράς (με σκοπό την βελτιωμένη μετάνοια σε πιο 'στατικά' υποκείμενα στοχαστικά περιβάλλοντα) αποτυγχάνει στο σκοπό της. Παρ'όλα αυτά η αποτυχία του αλγορίθμου δεν έγκειται στην εκτίμηση της διασποράς αυτή καθ'αυτή, καθώς η εκτίμηση της διασποράς είναι απροσδόκητα εύρωστη στην αλλοίωση (περαιτέρω από την εκτίμηση μέσης τιμής). Η αδυναμία του αλγορίθμου έγκειται στις μεταβαλλόμενου μήκους εποχές που χρησιμοποιεί. Συγκεκριμένα, στον αλγόριθμο αυτό μπορεί κάλλιστα να μην έχουμε την εκθετική αύξηση του μήκους που επιδεικνύει ο *BARBAR*, μάλιστα μπορεί να μην είναι καν αύξουσα συνάρτηση της εποχής. Παρ'όλα αυτά ο σχεδιασμός αλγορίθμων που αξιοποιούν εκτιμήσεις διασποράς σε αλλοιωμένα περιβάλλοντα, παραμένει μια ανοιχτή κατεύθυνση, η οποία ίσως χρειαστεί διαφορετικές προσεγγίσεις, για την επίτευξη βελτιωμένων εγγυήσεων μετάνοιας.

Κείμενο στα Αγγλικά

Chapter 1

The Multi Armed Bandit problem

1.1 Introduction

You are a gambler and really like slot machines. The casino you frequent has k machines, (or more coloquially known: one armed bandits). In a limited time frame, how can you maximize your profits, by learning which bandit is the 'best'?

This is the well known problem of *Multi Armed Bandits (MAB)*, which was originally proposed by Robbins in [50] (1952), but has its roots (specifically the widely used Thompson sampling algorithm) even earlier in the work of Thompson [56] (1933). The unique name of this problem comes from the setting we introduced and was first named so by Bush and Mosteller in [20].

The Multi Armed Bandits problem finds application in seemingly different areas of interest. One of the first such applications which also brought attention to it, was that of how to design efficient medical trials and choose the better course of treatment, as in [1]. More specifically, one can model a medical trial as a series of sequential patient arrivals and a variety of treatments (the arms) which can be chosen for each individual patient. The similarity with the aforementioned problem can easily be inferred then, assuming the treatments can be graded in a scale and the patients are "uniform" in the sense of their condition. Medical applications also extend to more usual settings, such as finding the correct dosage of a prescription (a somewhat lengthy process in various scenarios, such as thyroid issues or some mental disorders).

Another very famous application, which emerged more recently, is that of advertisement placement (for example in search engine results). Search engines host a variety of advertisements for many different clients in exhange for compensation. The bulk of the revenue is a function of the click through rate (in practical terms; how many times was an ad clicked), as a vendor would like to maximize their visibility. As such, the search engine is motivated to show each user an advertisement that maximizes the probability that the user clicks on it, elliciting their interest. This problem is also very similar to the classic MAB problem, albeit a bit more complicated, as each user has different interests and likes (a context is needed). Recommendation systems and ad placement applications have driven a great deal of the current research towards Multi Armed Bandits. Some indicative applied works in this direction can be found in [59, 44, 16] and for a more comprehensive survey on recommendation applications the reader can consult [54].

A non exhaustive list of other applications, range from partial feedback games, where an agent in a complicated repeated game is faced with the choice of various strategies only observing her utility, with some examples in [37, 15, 35, 42], to portfolios [53, 57] and dynamic pricing in auctions with clients arriving sequentially [47, 48], to packet routing [7, 55], social and communication networks [25, 23, 26, 19] and many others. The reader is referred to [12] for a more thorough survey.

1.2 Structure

We model the quintessential problem of Multi-Armed Bandits on the next subsection and on the next chapter we consider the classic stochastic variant [18]. We first study a naive *non-adaptive* algorithm coined *Explore then Commit*, which attains sublinear regret, which while not close to being optimal, is beneficial in gaining an intuition for the problem and later algorithms. The next algorithm we consider is an *adaptive* algorithm which is widely used. The algorithm, known as *Active Arm Elimination*, first described by [30], considers the history of the rewards to adapt its choices, attaining aconsiderably better regret guarantee. A quite similar algorithm (although not that clear at first) is the very well known *UCB* algorithm by Auer et al. in [4]. Both of these algorithms attain almost-optimal instance-dependent and instance-independent regret bounds (considering the information theoretic lower bounds on regret). We finish the chapter by presenting *UCBV* proposed by [3], an algorithm that differs from the above, as it also uses variance estimation in its decisions, which guarantees an even better regret bound when these variances are small.

The third chapter chapter 3 considers the *Adversarial MAB* or (or non-stochastic) problem. We present the celebrated algorithm *Exp3* by Auer et al. in [6], which draws inspiration from the full-feedback *Online Learning* problem and specifically the *Multiplicative Weights Algorithm* first described by Littlestone and Warmuth [45].

In the fourth chapter we describe a model of environments which are between these two extremes and specifically environments which can be described as stochastic in nature, but experience corruptions/perturbations by an adversary, a problem first considered by Lykouris et al. [46]. The first algorithm we study is by the same authors and attains the optimal regret bounds in the stochastic case, which deteriorate gracefully as corruption increases. We also study *BARBAR*, an algorithm proposed by Gupta et. al. [36], which achieves a substantially better regret bound dependence on the corruption that the adversary injects.

Our last chapter attempts to extend the results, by considering an algorithm that also uses variance estimates in its decisions, in order to obtain an improved regret bound when the instance allows for small variances. Even though variance estimations prove not to be too sensitive to corruption and our algorithm attains the desired regret bound in the stochastic case, a key family of problem instances shows that this direction might not be achievable with our current algorithm.

1.3 The model

Let's dive into more technical details. We will model our problem as follows. The agent (or learner) has k arms (or actions)[1] to choose from and T rounds available. Each arm a [2] when picked at round t, supplies the agent with a reward $r_t(a)$ and only that. We call this bandit feedback (as opposed to other scenarios, such as full feedback: revealing each and every action's reward at the round). The agent strives to maximize her reward in the span of the T rounds.

Definition 1.3.1. Multi Armed Bandit Problem (MAB)
1. The learner has k available actions at each round t and is given T rounds in total.
2. At each round t the learner chooses an action a_t.
3. The learner receives and observes the reward $r_t(a_t)$ of arm a_t.
4. The learner strives to maximize her reward.

There is an important detail we have glossed over though: how are the rewards chosen? Are they random, meaning that they are observations of a distribution? If so, is that distribution static throughout the course of time? What if we make no assumptions at all and allow an "adversary" to make up the rewards as they please? We will study two important cases.

Definition 1.3.2. Stochastic MAB
The reward $r_t(a)$ is sampled from a static distribution D_a, with expectation $\mu(a)$ and variance σ_a (obviously unknown to the learner). No other assumptions about the distribution are made.

Assumption 1.3.1. Adversarial MAB
There are no assumptions about the rewards at all. The rewards $r_t(a)$ could be adversarially (and even adaptively) constructed.

There is an obvious dilemma for the agent in both these cases. Should she strive to explore and learn more about the arms, possibly incuring a lot of regret, or exploit and choose the best action using the information she has collected, possibly not having learned the 'best' arm yet? This exploration-exploitation tradeoff is a hallmark of the MAB problem and is something that any algorithm hoping to tackle this problem has to juggle.

Alas, the methods employed for each environment differ quite a lot. In the case of the stochastic bandits, while the rewards are random, they are *stochastically* random, which means that after a sufficient amount of "pulls" (times that an arm is chosen), we will have a pretty good idea of how good is a particular arm, with high probability. The zero assumptions

[1] Both pairs of words used interchangeably. However, we will prefer to use 'learner' and 'arm'.

[2] a is an index in $[k] := \{1, \dots, k\}$

on the adversarial case though, leave no room for estimation and any confidence about the rewards of an arm. However, even though in general the stochastic case is drastically ”easier” in most cases, in the worst case both scenarios exhibit more or less the same difficulty (which we will quantify with asymptotic regret).

1.4 Regret

How can the learner measure her success? She obviously cannot maximize her profit in general due to stochasticity, even if she somehow knew which arm achieves the highest expected reward and she chose that one every time. A logical measure would be to compare her choices to that of someone that is a 'good player'. How do we define a good player though? The best player is the one that knows the sequence of the arms that provide the maximum cumulative reward. However, achieving results close enough seems to be an insourmountable task. A reasonable measure of success would be to compare herself to the agent that clairvoyantly knows the best arm in advance (maximum revenue of a fixed arm in hindsight). Let's formalize this.

Definition 1.4.1. Regret
We define regret as the quantity:

$$R(T) := \sum_{t=1}^{T} r_t(a^*) - \sum_{t=1}^{T} r_t(a_t)$$

Where $a^* := \underset{a\in[k]}{\operatorname{argmax}} \sum_{t=1}^{T} r_t(a)$ *(the 'best' arm).*

We call this quantity regret, as it measures how much the agent ”regrets” by not playing the 'best' arm. This is the usual definition of regret one may encounter in most sources in the literature. There exists another measure, which is sometimes called ”Pseudo-Regret” and applies to stochastic bandits.

Definition 1.4.2. Pseudo-Regret
We define Pseudo-Regret as the quantity:

$$\begin{aligned} R(T) &:= \sum_{t=1}^{T} \mu^* - \sum_{t=1}^{T} \mathbb{E}\left[r_t(a_t)\right] \\ &= T\mu^* - \sum_{t=1}^{T} \mu(a_t) \end{aligned}$$

Where $\mu^* := \max a \in [k] \mathbb{E}\left[r_t(a)\right]$ *and* a_t *is the arm the agent chose at round* t*.*

Another very useful way to see this definition is the below formulation:

$$R(T) = \sum_{a\in[k]} n_T(a)\Delta(a)$$

Where:

$$n_T(a) := \sum_{i \in [T]} \mathbb{I}\{a_t = a\} \qquad \text{(times that arm } a \text{ is played up to time } T\text{)}$$

$$\Delta(a) := \mu^* - \mu(a) \qquad \text{(the 'gap' of arm a)}$$

We note that in some literature the MAB model encases scenarios where each action is associated with a cost and as such the learner strives to minimize her total accumulated cost. The regret is still measured as the difference with respect to an oracle that plays the best fixed arm, but since we measure losses, the signs in the cumulative sums are opposite.

Chapter 2

Stochastic MAB

The first problem we are going to study is that of the Stochastic MAB. Each arm a is associated with a (static) distribution D_a and an expected reward $\mu(a)$. We denote with μ^* the highest such expected reward and with a^* the arm that corresponds to it (break ties arbitrarily).

2.1 Explore then Commit (ETC)

The stochasticity of the arms allows us to claim with some confidence - after a sufficient amount of pulls - that an arm is "better" or "worse" than some other. How would we go about in designing an algorithm that takes advantage of this? A logical (but naive as we will see) first idea would be to play each and every arm a predefined number of times and then exploit; choose the one that has the highest average reward and play it for the rest of the rounds. Let's formalize this algorithm, which is known as Explore then Commit in the literature (or Explore First). We note that the origins of the algorithm are unclear.

Algorithm 1 Explore then Commit

```
for $t \in [k \cdot M]$ do                          ▷ Every arm will be played $M$ times
    $a_t = t \mod k$
Define $\bar{\mu}(a)$ the $M$-sample average of $D_a$ from these samples.
for $t \in [T] \backslash [kM]$ do                  ▷ Play the 'best' arm from now on
    $a_t = \text{argmax}_{a \in [k]} \bar{\mu}(a)$
```

This algorithm is simple enough, but in its current state is incomplete. We haven't defined how many samples we will need from each arm, ie the quantity M. Recall that we measure our success compared to someone who knew beforehand which arm is best in expectation and we defined the notion of 'regret' to reflect that. We turn to revealing an "optimal" choice for M (and proving it later).

Theorem 2.1.1. (High probability regret guarantee)
Explore Then Commit *with* $M = \Theta(logT)$ *samples of each arm achieves regret* $O\left(T^{2/3}\sqrt[3]{k \ln \frac{k}{\delta}}\right)$ *with probability at least* $1-\delta$.

Our first course of action will be to bound the deviation of the empirical average of the rewards from their true expectation, in order to argue about how much stochasticity has affected our decision. This will be a recurring theme in the entirety of the thesis (and in general in the bandit literature). In order to do that we will have to use a so-called *concentration inequality* and specifically *Hoeffding's Inequality*. This deviation from the true expectation will basically decide how much we could "lose" with respect to the best arm (or how much we regret), when we chose the seemingly better arm after playing each one M times. The theorem's proof follows.

Proof. Fix an arm a and define $X_i(a)$ as its i-th stochastic reward. We assume that the environment is indifferent to our actions and as such these RVs are pairwise independent. We can easily infer that after M samples we have the following by Hoeffding:

$$\mathbb{P}\left(\left|\frac{1}{M}\sum_{i=1}^{M} X_i(a) - \mu(a)\right| \geq \epsilon\right) \leq 2\exp\left\{-2M\epsilon^2\right\}$$

As mentioned earlier, bounding the deviation from the expectation will aid us in analyzing our regret. However, we need such a concentration equality for each arm, so by union-bound:

$$\mathbb{P}\left(\exists a \in [k] : \left|\frac{1}{M}\sum_{i=1}^{M} X_i(a) - \mu(a)\right| \geq \epsilon\right) \leq 2k\exp\left\{-2M\epsilon^2\right\}$$

This concentration inequality is not very useful in its current form. We would like to have a bound that can hold with an arbitrary probability (which ofcourse will dictate the deviation). In other words, if we let $\epsilon = \sqrt{\frac{1}{2M}\ln\frac{k}{2\delta}}$ we have the following *high-probability* bound:

$$\mathbb{P}\left(\forall a \in [k] : \left|\frac{1}{M}\sum_{i=1}^{M} X_i(a) - \mu(a)\right| \leq \sqrt{\frac{1}{2M}\ln\frac{2k}{\delta}}\right) \geq 1-\delta$$

We are ready to analyze our algorithm's regret. We assume that the above event holds and as such our regret guarantee will hold with the same probability. In the exploration phase our algorithm tries each arm one by one k times and as such could suffer a regret of $(k-1)M$. In the exploitation phase the algorithm settles on the seemingly best arm and plays it for the remaining $(T-kM)$ rounds. How much worse can the seemingly best arm be from the true one? Well, we can answer that through the concentration inequalities we constructed.

Assume that the algorithm does not choose the best arm a^* and instead chooses another suboptimal arm a. By denoting $\bar{\mu}_a$ the average reward of arm a after M samples, we have that:

$$\begin{gathered}\bar{\mu}(a) \geq \bar{\mu}^* \\ \Rightarrow \mu(a) + \sqrt{\frac{1}{2M}\ln\frac{2k}{\delta}} \geq \bar{\mu}_a \geq \bar{\mu}^* \geq \mu^* - \sqrt{\frac{1}{2M}\ln\frac{2k}{\delta}} \\ \Rightarrow \mu^* - \mu(a) \leq 2\sqrt{\frac{1}{2M}\ln\frac{2k}{\delta}}\end{gathered}$$

Which means that the (pseudo)regret satisfies:

$$\begin{aligned} R(T) &\leq (k-1)M + (T-kM)\Delta(a) \\ &\leq kM + T \cdot \left(2\sqrt{\frac{1}{2M} \ln \frac{2k}{\delta}} \right) \end{aligned}$$

with probability at least $1-\delta$.

Now we can notice that the above expression is a sum of two functions of M with opposite monotonicity. We can approximately minimize the regret by choosing M such that the above summands are approximately equal.

If $M = \left(\frac{T}{k}\right)^{2/3} \sqrt[3]{\ln \frac{k}{\delta}}$ then ETC achieves asymptotic regret $O\left(T^{2/3} \sqrt[3]{k \ln \frac{k}{\delta}}\right)$ with probability at least $1-\delta$. □

Corollary 2.1.1. *ETC with $M = \left(\frac{T}{k}\right)^{2/3} \sqrt[3]{\ln (kT^2)}$ samples of each arm achieves* expected *regret of $O(T^{2/3} \sqrt[3]{k \ln (kT)})$*

Proof. This is easy enough to see if we let $\delta = 1/T^2$. Denote by $\mathcal{E}$ the event that all concentration inequalities hold (this happens with probability at least $1 - 1/T^2$). We have then:

$$\begin{aligned} \mathbb{E}\left[R(T)\right] &= \mathbb{P}\left(\mathcal{E}\right) \cdot R(T \mid \mathcal{E}) + \mathbb{P}\left(\neg\mathcal{E}\right) \cdot R(T \mid \neg\mathcal{E}) \\ &\leq (1 - 1/T^2) \cdot O(T^{2/3} \sqrt[3]{k \ln (kT^2)}) + 1/T^2 \cdot T \\ &\leq O(T^{2/3} \sqrt[3]{k \ln (kT)}) \end{aligned}$$

□

The expected regret is simpler in its formulation, but many applications exhibit small tolerance and require stricter guarantees which points to high probability regret guarantees. This is because while an algorithm can exhibit a small expected regret, this could very well be the result of small regret events balancing high regret ones.

2.2 Active Arm Elimination (AAE)

Explore then Commit is a very simple MAB algorithm that guarantees sublinear regret, but is inefficient. Explore then commit is a *non-adaptive* algorithm, it does not change its behaviour no matter the instance or the history of observed rewards. Assuming a sufficient amount of rounds the learner can possibly detect (depending on the instance) that some arm(s) are sub-optimal with high confidence, early in the exploration. This observation is the basis of the *adaptive* algorithm that is known as *Active Arm Elimination*, first introduced by Even-Dar et al. in [30]. A thorough study of the use of elimination in this setting and more generally in reinforcement learning can be found on [29]. We present the algorithm below.

Algorithm 2 Active Arm Elimination

```
$S \leftarrow [k]$
for $t \in [T]$ do
    $CB(a;t) = \left[\bar{\mu}_t(a) - \sqrt{\ln\frac{2kT}{\delta}\cdot\frac{1}{n_t(a)}},\ \bar{\mu}_t(a) + \sqrt{\ln\frac{2kT}{\delta}\cdot\frac{1}{n_t(a)}}\right]$
    If there exist arms $a, a'$ with $UCB(a') < LCB(a)$ , then $S \leftarrow S\backslash\{a'\}$
    Play the arm $a \in S$ that's been played the least (break ties arbitrarily)
```

AAE pulls each arm in a round-robin manner, just like ETC, but when it notices that some arm is sub-optimal with high confidence it ”deactivates” it and only considers the rest (in the end sticking with the last remaining). Intuitively, this ”adaptiveness” and preemptive decision-making should result in improved regret.

In order to argue about AAE's regret we will have to invoke the Azuma-Hoeffding inequality (see Theorem A.0.4), which is an extention of Hoeffding's concentration inequality for *martingales*. This is necessary because AAE is adaptive and the observed rewards affect the times the arm is sampled, which now forms a stochastic variable in itself.

Lemma 2.2.1. *The sequence of the random variables* $Y_t(a) = \mathbb{I}\{a_t = a\}(r_t(a) - \mu(a))$ *form a martingale difference sequence.*

Proof. Define $\mathcal{H}_{t-1} := \{Y_1(a) \dots Y_{t-1}(a)\}$ for brevity. The random variable representing that an arm is pulled at round t and its stochastic reward at that round are independent of eachother (in the classic *MAB problem*):

$$\begin{aligned}\mathbb{E}\left[Y_t(a) \mid \mathcal{H}_{t-1}\right] &= \mathbb{P}\left(\mathbb{I}\{a_t = a \mid \mathcal{H}_{t-1}\} = 1\right) \cdot \mathbb{E}\left[r_t(a) - \mu(a) \mid \mathcal{H}_{t-1}, a_t = a\right] + 0\\ &= \mathbb{P}\left(a_t = a \mid \mathcal{H}_{t-1}\right) \cdot 0\\ &= 0\end{aligned}$$

It is of course easy to verify that the unconditional expectation of any RV of the sequence is finite. These conditions are sufficient. □

Intuitively, although the samples taken from AAE aren't independent from each other, they only have a ”minor” dependency, which allows us to formulate concentration inequalities similar to ones that hold for fully independent sequences of variables. We can proceed to the regret analysis.

Theorem 2.2.1. *AAE achieves* instance-dependent *regret*

$$R(T) = O\left(\ln \frac{kT}{\delta} \cdot \sum_{\substack{a\in[k]:\\ \mu(a)<\mu^*}} \frac{1}{\Delta(a)}\right)$$

with probability at least $1-\delta$.

Proof. As proved earlier, the RV $Y_\tau(a) = \mathbb{I}\{a_\tau = a\}(r_\tau(a) - \mu(a))$ is a MDS. By Azuma-Hoeffding (Theorem A.0.4) we have that:

$$\mathbb{P}\left(\left|\sum_{\tau=1}^{t} Y_\tau(a)\right| \geq \epsilon\right) \leq 2\exp\left\{-\frac{2\epsilon^2}{\sum_{\tau=1}^{t}(b_\tau - a_\tau)^2}\right\}$$

Where $[a_\tau, b_\tau]$ is the smallest interval that $Y_\tau(a)$ lies in almost surely. If we denote by $n_t(a)$ the number of pulls of arm a until time t, then only this number of intervals are non-trivial (length not zero), since all the other RVs are almost surely 0. Which means that

$$\sum_{\tau=1}^{t}(b_\tau - a_\tau)^2 \leq 4n_t(a) \text{ (since } Y_\tau \text{ is in [-1, 1] almost surely)}$$

So the inequality becomes:

$$\mathbb{P}\left(\left|\sum_{\tau=1}^{t} Y_\tau(a)\right| \geq \epsilon\right) \leq 2\exp\left\{-\frac{\epsilon^2}{2n_t(a)}\right\}$$

By union bound we have that:

$$\mathbb{P}\left(\exists t\in[T], \exists a\in[k] : \left|\sum_{\tau=1}^{t} Y_\tau(a)\right| \geq \epsilon\right) \leq 2kT\cdot\exp\left\{-\frac{\epsilon^2}{2n_t(a)}\right\}$$

Note that the sum amounts to the realized accumulated reward minus the expected accumulated reward:

$$\sum_{\tau=1}^{t} Y_t(a) = n_t(a)\cdot(\bar{\mu}(a) - \mu(a))$$

We abuse notation by overloading $\bar{\mu}(a)$ to mean the $n_t(a)$-sample average of a, where t is given by the context. By simple manipulations we have that:

$$\mathbb{P}\left(\exists t\in[T], \exists a\in[k] : |\bar{\mu}(a) - \mu(a)| \geq \epsilon\right) \leq 2kT\cdot\exp\left\{-\frac{1}{2}n_t(a)\epsilon^2\right\}$$

Or the following concentration inequality:

$$\mathbb{P}\left(\forall t\in[T], \forall a\in[k] : |\bar{\mu}(a) - \mu(a)| \leq \sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_t(a)}}\right) \geq 1-\delta \tag{2.1}$$

Assuming this event is realized, then the first - but not the most obvious - observation is that the optimal arm will never get "deactivated". Intuitively, this is because LCBs are bounded above by the true expected reward and UCBs are lower bounded by the same quantity, or in other words, the confidence interval will only "close in" on the true expectation as time goes on and can't "detach" from it. All of this means that if the optimal arm were to be deactivated it could not have been optimal. We can make this argument formal and prove it rigorously, but we choose to keep the proof succint.

With this important observation we proceed to the regret analysis by bounding the number of pulls of a suboptimal arm. Each arm gets deactivated after $n_T(a)$ pulls by the definition of $n_t(a)$. Fix arm a, since the arms get pulled in a round robin fashion; at the time of a's deactivation, the optimal arm has been pulled either $n_T(a)$ times or $n_T(a) \pm 1$ times. Whichever is the case, arm a was not deactivated when both arms had been played only $n_T(a) - 1$ times:

$$\begin{aligned} UCB(a) &\geq LCB(a^*) \\ \Leftrightarrow \sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}} &\geq \bar{\mu}^* + \sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}} \\ \Leftrightarrow \bar{\mu}^* - \bar{\mu}(a) &\leq 2\sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}} \end{aligned} \tag{2.2}$$

By definition of the confidence bounds:[1]

$$\mu(a) \geq LCB(a) = \bar{\mu}(a) - \sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}} \tag{2.3}$$

$$\mu^* \leq UCB(a^*) = \bar{\mu}^* + \sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}} \tag{2.4}$$

Combining these three inequalities:

$$\Delta(a) = \mu^* - \mu(a) \leq 4\sqrt{\frac{2\ln\frac{2kT}{\delta}}{n_T(a)-1}}$$

Which means that:

$$n_T(a) = O\left(\ln\frac{2kT}{\delta} \cdot \frac{1}{\Delta(a)^2}\right) \tag{2.5}$$

So we have that our regret satisfies:

$$R(T) = \sum_{a\in[k]:\mu(a)<\mu^*} n_T(a) \cdot \Delta(a) = O\left(\ln\frac{kT}{\delta} \cdot \sum_{\substack{a\in[k]: \\ \mu(a)<\mu^*}} \frac{1}{\Delta(a)}\right)$$

□

Corollary 2.2.1. AAE *achieves an expected regret of*

$$\mathbb{E}\left[R(T)\right] = O\left(\ln(kT) \cdot \sum_{\substack{a\in[k]: \\ \mu(a)<\mu^*}} \frac{1}{\Delta(a)}\right)$$

[1] We silently assume here that we play each arm at least two times. We can make this a requirement in the algorithm by playing every arm twice in the beginning. By information theoretic bounds, the regret is $O(T)$ anyway when $k = O(T)$ [4].

This regret already seems like an improvement to the one that the naive *ETC* guarantees, it is exponentially better if we assume that the gaps are 'constant'. This regret is instance-dependent though, to be fully correct and compare both algorithms we should strive for an instance-agnostic regret formulation. We will prove the following.

Theorem 2.2.2. AAE *incurs an instance-independent regret*

$$R(T) = O\left(\sqrt{kT \cdot \ln \frac{kT}{\delta}}\right)$$

with probability at least $1 - \delta$.

Proof. Let's go back to Equation (2.5) and examine it from a different perspective. If we rearrange:

$$\Delta(a) = O\left(\sqrt{\ln \frac{kT}{\delta} \cdot \frac{1}{n_T(a)}}\right) \tag{2.6}$$

As usual, for the regret:

$$\begin{aligned}
R(T) &= \sum_{a \in [k]: \mu(a) < \mu^*} n_T(a)\Delta(a) \\
&\leq \sum_{a \in [k]: \mu(a) < \mu^*} n_T(a) \cdot O\left(\sqrt{\ln \frac{kT}{\delta} \cdot \frac{1}{n_T(a)}}\right) && \text{by Equation (2.6)} \\
&\leq O\left(\sqrt{\ln \frac{kT}{\delta}} \cdot \sum_{\substack{a \in [k]: \\ \mu(a) < \mu^*}} \sqrt{n_T(a)}\right) \\
&\leq O\left(\sqrt{\ln \frac{kT}{\delta}} \cdot \sqrt{k \sum_{a \in [k]} n_T(a)}\right) && \text{by Jensen's inequality and concavity of } \sqrt{\cdot} \\
&\leq O\left(\sqrt{\ln \frac{kT}{\delta}} \cdot \sqrt{kT}\right) && \text{since } \sum_{a \in [k]} n_T(a) = T
\end{aligned}$$

□

We proved that the intuition of eliminating arms that are behaving fairly sub-optimally earlier leads to better guarantees for regret. Both regret bound formulations are basically optimal, since information theoretic analysis shows that for instance-independent regret [4] :

Theorem 2.2.3. *No stochastic bandit algorithm can achieve expected regret*

$$\mathbb{E}\left[R(T)\right] = o(\sqrt{kT})$$

And for instance-dependent regret [41] (and also [18]):

Theorem 2.2.4. *No stochastic bandit algorithm can achieve expected regret*

$$\mathbb{E}\left[R(T)\right] = o(C \ln T)$$

where the constant C *depends only on the instance (and not the time horizon* T *).*

2.3 Upper Confidence Bound Algorithm (UCB)

There is another adaptive algorithm that (surprisingly to some) leads to the same guarantees as *AAE*. The *UCB* algorithm by Auer et al. in [4] is one of the most well known Stochastic MAB algorithms and is fairly simple: at every round it chooses the arm that has the highest upper confidence bound.

Algorithm 3 UCB

```
for $t \in [T]$ do
    Play arm $a$ that maximizes $UCB(a;t) := \bar{\mu}_t(a) + \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_t(a)}}$
```

Before presenting any regret gurantees and going in to analysis, it would be beneficial to understand why this algorithm *works*. A high *Upper Confidence Bound* can be a result of two things: either that particular arm has been pulled few times or it has a high expected reward. In both cases the learner would have incentive to play that particular arm (either to explore, or exploit respectively). *UCB* falls under a wider umbrella of sequential decision making policies in uncertain environments, namely *Optimism Under Uncertainty*. We continue with our claims.

Theorem 2.3.1. UCB *achieves regret*

$$R(T) = O\left(\ln \frac{kT}{\delta} \cdot \sum_{\substack{a \in [k]: \\ \mu(a) < \mu^*}} \frac{1}{\Delta(a)}\right)$$

with probability at least $1 - \delta$ *and an expected regret*

$$\mathbb{E}\left[R(T)\right] = O\left(\ln (kT) \cdot \sum_{\substack{a \in [k]: \\ \mu(a) < \mu^*}} \frac{1}{\Delta(a)}\right)$$

It also guarantees an instance-independent regret bound

$$R(T) = O\left(\sqrt{kT \cdot \ln \frac{kT}{\delta}}\right)$$

with probability at least $1 - \delta$.

Proof. Even though two pairs of 'active' arms (every arm is active in *UCB*) could have wildly different number of pulls, we can use the same argument as in the analysis of *AAE*, comparing a suboptimal arm to the optimal one. The relevant concentration inequality (2.1) holds as is and in the following analysis we condition on the event that they hold. Fix arm a and consider the last time it is played; it must be then:

$$UCB(a) \geq UCB(a^*) \geq \mu^*$$

The second inequality although simple, is crucial to the analysis. At the point a is pulled for the last time it had been previously played $n_T(a) - 1$ times, so we have that:[2]

$$UCB(a) = \bar{\mu}(a) + \sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_T(a) - 1}} \geq \mu^* \tag{2.7}$$

But we also have that $\mu(a) \geq LCB(a)$, so then:

$$\mu(a) + 2\sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_T(a) - 1}} \geq UCB(a) \tag{2.8}$$

By the two numbered equations we have then:

$$\mu^* - \mu(a) \leq 2\sqrt{\ln \frac{2kT}{\delta} \cdot \frac{1}{n_T(a) - 1}}$$

Or more simply:

$$n_T(a) = O\left(\ln \frac{kT}{\delta} \cdot \frac{1}{\Delta(a)^2}\right) \tag{2.9}$$

This is exactly the same as Equation (2.5), meaning that *AAE* and *UCB* will pull each arm roughly the same amount of times (asymptotically). As one can verify, intuitively and formally, this is a sufficient condition for the claimed regret bounds. □

[2] Again requiring that we play at least two times each arm at the beginning.

2.4 Upper Confidence Bound with Variance estimates (UCB-V)

In all the previous algorithms the only information used to make any decision is the empirical average reward (and the confidence interval around the true expected reward, by extension). Audibert et al. in [3] make an important observation; arms that behave fairly statically (meaning the observed rewards do not diverge a lot from each other) should ensure a higher confidence in the sampled average. In other words, an arm that has a small variance should have a smaller confidence interval and we should learn its gap quicker if we track its sample variance. This is encapsulated in the author's algorithm, called *UCB-V* below (slightly different than the one actually proposed):

Algorithm 4 UCB-V

```
for $t \in [T]$ do
    Play arm $a$ that maximizes $UCBV(a;t) := \bar{\mu}_t(a) + \sqrt{8V_t(a) \cdot \ln \frac{4kT}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT}{\delta} \cdot \frac{13}{n_t(a)}$
```

When the sample variance is high, this upper confidence bound is asymptotically the same as for classic UCB (since $x = o(\sqrt{x})$ when $x < 1$), specifically when this variance is asymptotically close to the average reward (for example this will happen on Bernoulli arms with high probability). However, when the variance is asymptotically smaller than its expected reward (for example an exponential variable), then the same holds for the empirical quantities with high probability (we will prove that later). In that case, the confidence interval around the expected reward shrinks considerably. [3] prove the following for their algorithm:

Theorem 2.4.1. UCB-V *achieves expected regret*

$$\mathbb{E}\left[R(T)\right] = O\left(\ln\left(kT\right) \cdot \sum_{\substack{a\in[k]:\\ \mu(a)<\mu^*}} \left\{\frac{\sigma_a^2}{\Delta(a)} + 1\right\}\right)$$

where σ_a is the variance of a's stochastic reward.

For a proof of this result the reader can follow [3]. The result shown here is slightly different than they one proved, since the authors also consider a slightly more general environment (the rewards are in an arbitrary finite interval). We will prove the following guarantee for $UCB - V$, using a much simpler argument than the one presented on the mentioned paper. This analysis is also in fashion with the analysis of previous bandits algorithms we've showed.

Theorem 2.4.2. UCB-V *achieves regret*

$$O\left(\ln\frac{kT}{\delta} \cdot \sum_{\substack{a\in[k]:\\ \mu(a)<\mu^*}} \left\{\frac{\sigma_a^2}{\Delta(a)} + 1\right\}\right)$$

with probability at least $1 - \delta$

Our first course of action is to bound the deviation of the sample average from the true expected reward, as always. As we are interested in incorporating the variance of the rewards a concentration inequality like Bernstein's would be ideal. However, as in UCB at each round the number of pulls for any arm is a random variable and each pull is not fully independent from the previous ones. As such, once again we will need a martingale argument steering us towards Freedman's Inequality: Corollary A.0.1.

First, we define the sequence of RVs $Y_t = \mathbb{I}\{a_t = a\}(r_t(a) - \mu(a))$. In the same way as in earlier sections it is easy to show that this is a *Martingale Difference Sequence*. Then, in order to use Freedman's Inequality on this sequence we need to bound the "predictable variation". That is:

$$\mathcal{V}_t := \sum_{\tau=1}^{t} \mathbb{E}\left[Y_\tau^2 \mid \mathcal{H}_{\tau-1}\right], \text{ where } \mathcal{H}_{\tau-1} := \{Y_1, \dots, Y_{\tau-1}\}$$

Lemma 2.4.1. *We have that:*

$$\mathcal{V}_t \leq t \cdot \sigma_a^2$$

Proof. Notice that since the algorithm's decision to pull an arm doesn't affect its stochastic reward (they are independent) we have that:

$$\begin{aligned}\mathbb{E}\left[Y_\tau^2 \mid \mathcal{H}_{\tau-1}\right] &= \underbrace{\mathbb{P}\left(a_t = a \mid \mathcal{H}_{\tau-1}\right)}_{\leq 1} \cdot \underbrace{\mathbb{E}\left[(r_\tau(a) - \mu(a))^2 \mid \mathcal{H}_{\tau-1}, a_\tau = a\right]}_{=\sigma_a^2} \\ &\leq \sigma_a^2\end{aligned}$$

Since rewards between rounds are independent. □

We can apply Freedman's lemma (see Corollary A.0.1) with the above bound then and get the below lemma.

Lemma 2.4.2.

$$|\bar{\mu}_t(a) - \mu(a)| \leq \frac{1}{n_t(a)}\sqrt{2t\sigma_a^2 \cdot \ln\frac{2kT}{\delta}} + \ln\frac{2kT}{\delta} \cdot \frac{4}{3n_t(a)}$$

for all arms a and rounds t with probability at least $1 - \delta$ where $\bar{\mu}_t(a) := \frac{1}{n_t(a)}\sum_{\tau=1}^{t} \mathbb{I}\{a_t = a\}r_t(a)$

However this is not exactly what we want. In the case that the arm is pulled too few times up to time t, ie when $n_t(a) << t$ this bound is bad. However wouldn't a stronger bound hold when it was last pulled? Before entertaining this obervation we present the above lemma in a way more suitable to our following analysis.

Lemma 2.4.3. *Let $U_1, \dots, U_t$ be iid samples/copies of a distribution with expectation μ and variance σ^2. Let n_t be a discrete random variable supported on $[1, t]$ (that can possibly depend on the realizations of X_i). We have the following concentration inequality:*

$$|\bar{\mu}_t - \mu| \leq \frac{1}{n_t}\sqrt{2t\sigma^2 \cdot \ln\frac{2kT}{\delta}} + \ln\frac{2kT}{\delta} \cdot \frac{4}{3n_t}$$

for all $1 \leq t \leq T$ with probability at least $1 - \delta$ where $\bar{\mu}_t := \frac{1}{n_t}\sum_{i=1}^{n_t} U_i$.

Note that in the above inequality n_t can very well depend on the realizations of the U_i. If n was known ahead of time (fixed) Bernstein's inequality would give the following bound:

$$|\bar{\mu} - \mu| \leq \sqrt{\frac{2}{n}\sigma^2 \ln \frac{2kT}{\delta}} + \ln \frac{2kT}{\delta} \cdot \frac{4}{3n}$$

Since it might be the case that $n_t << t$ our bound could be way worse than the above. The current lemma is not sufficient as we will see later.

Although, notice that the opposite could have happened as well, meaning that n_t could have been very close to t and as such their fraction would be at most a constant, basically gaining the Bernstein bound. If we telepathically knew a *constant* upper bound for n_t that is close to it (a constant factor away) we would be done. Alas, since n_t is a random variable in $[t]$ this is not possible.

However, we could "break" $[t]$ into smaller intervals such that each endpoint is at most a constant factor away from the other. In that case, there would exist an interval that contains n_t and we could use the upper endpoint as an upper bound, as it would only be at most a constant factor away from n_t! If we also factor that n_t could be in any of these intervals we are done. So with that in mind, we prove the next *seemingly* more powerful lemma.

Lemma 2.4.4.

$$|\bar{\mu}_t(a) - \mu(a)| \leq 2\sqrt{\frac{1}{n_t(a)}\sigma_a^2 \cdot \log \frac{2kT \log T}{\delta_1}} + \ln \frac{2kT \log T}{\delta_1} \cdot \frac{4}{3n_t(a)}$$

for all arms a and rounds t with probability at least $1 - \delta_1$ where $\bar{\mu}_t(a) := \frac{1}{n_t(a)} \sum_{i=1}^{n_t(a)} X_i(a)$ and $X_i(a)$ is the i-th pull of arm a.

Proof. We partition the interval in logarithmically many subsets as we mentioned

$$[t] := \{\{1, 2\}, \{3, \ldots, 6\}, \{7, \ldots, 14\}, \ldots, \{l_i, \ldots, r_i\}, \ldots\}$$

It is easy to see that there at most $\log t$ such intervals. Also notice that $r_i \leq 2l_i$ for any interval with index i. There always exists an interval i such that $l_i \leq n_t(a) \leq r_i$ and in that case, using the previous lemma we have that with probability at least $1 - \delta$

$$\begin{aligned}
|\bar{\mu}_t(a) - \mu(a)| &\leq \frac{1}{n_t(a)}\sqrt{2r_i \cdot \sigma^2 \cdot \ln \frac{2kT}{\delta}} + \ln \frac{2kT}{\delta_1} \cdot \frac{4}{3n_t(a)} \\
&\leq \sqrt{2\frac{1}{n_t(a)}\frac{r_i}{n_t(a)}\sigma^2 \cdot \ln \frac{2kT}{\delta}} + \ln \frac{2kT}{\delta} \cdot \frac{4}{3n_t(a)} \\
&\leq \sqrt{2\frac{1}{n_t(a)} \cdot 2 \cdot \sigma^2 \cdot \ln \frac{2kT}{\delta}} + \ln \frac{2kT}{\delta} \cdot \frac{4}{3n_t(a)} \quad \text{since } r_i \leq 2l_i \leq 2n_t(a)
\end{aligned}$$

Now use $\delta = \frac{1}{\log T}\delta_1$ and union bound the failure probability for any of the $\log t \leq \log T$ relevant realization sets of $n_t(a)$. □

We have bounded the sample average deviation from the true expected reward. However, we don't know anything a priori about the variance of the arms. That is why we construct a sample variance and attempt to approach it. We need to bound the deviation of the sample variance of an arm from its true one. In order to do that we will assume a 'weaker' notion of sample variance for reasons to be explained.

Definition 2.4.1. *We define the sample variance as:*

$$V_t(a) := \frac{1}{2\lfloor n_t(a)/2 \rfloor} \sum_{i=1}^{\lfloor n_t(a)/2 \rfloor} U_i(a)$$

Where

$$U_i(a) := (X_{2i-1}(a) - X_{2i}(a))^2$$

and $X_i(a)$ is the i-th pull of arm a.

Lemma 2.4.5. *$V_t(a)$ is an unbiased estimator for the variance, meaning*

$$\mathbb{E}\left[V_t(a)\right] = \sigma_a^2$$

Proof. It is easy to see that when the X_i are iid RVs, so are U_i and regarding their expectation we have:

$$\begin{aligned}
\mathbb{E}\left[U_i\right] &= \mathbb{E}\left[X_{2i-1}^2 - 2X_{2i-1}X_{2i} + X_{2i}^2\right] \\
&= \mathbb{E}\left[X_{2i-1}^2\right] - 2\mathbb{E}\left[X_{2i-1}X_{2i}\right] + \mathbb{E}\left[X_{2i}^2\right] \\
&= \mu^2 + \sigma^2 - 2\mathbb{E}\left[X_{2i-1}X_{2i}\right] + \mu^2 + \sigma^2 && \text{since } \mathbb{V}ar(X) = \mathbb{E}\left[X^2\right] - \mathbb{E}\left[X\right]^2 \\
&= 2(\mu^2 + \sigma^2) - 2\mathbb{E}\left[X_{2i-1}\right]\mathbb{E}\left[X_{2i}\right] && \text{by independency} \\
&= 2\sigma^2
\end{aligned}$$

It follows that $\mathbb{E}\left[V_m(a)\right] = \sigma_a^2$.

□

At this point we need to get a hold of the deviation from the true variance, as we have done throughout the text for sample averages. We would like to use Bernstein's inequality, but that cannot happen, since -again- we would like a concentration inequality that is uniform in time and holds for a random number of plays $n_t(a)$, that depends on the rewards. For that reason following the previous analysis we can use Lemma 2.4.4 and get the following lemma.

Lemma 2.4.6. *The following hold for all arms a and rounds t, with probability at least $1 - \delta_2$. If $n_t(a) \geq 64 \ln \frac{2kT \log T}{\delta_2} \cdot \frac{1}{\sigma_a^2}$:*

$$\frac{1}{2}\sigma_a^2 \leq V_t(a) \leq 2\sigma_a^2$$

And also if $n_t(a) \leq 64 \ln \frac{2kT \log T}{\delta_2} \cdot \frac{1}{\sigma_a^2}$:

$$V_t(a) \leq 227 \ln \frac{2kT \log T}{\delta_2} \frac{1}{n_t(a)}$$

Proof. First we note that:

$$\begin{aligned}
\mathbb{V}ar(U_i(a)) &= \mathbb{E}\left[U_i^2(a)\right] - \mathbb{E}\left[U_i(a)\right]^2 \\
&\leq \mathbb{E}\left[U_i^2(a)\right] \\
&\leq \mathbb{E}\left[U_i(a)\right] && \text{since } U_i(a) \in [0,1] \\
&= 2\sigma_a^2
\end{aligned}$$

So we can use Lemma 2.4.4 with $U_i(a)$ instead of $X_i(a)$ and σ_a^2 as an upper bound on the variance. Then with probabiity at least $1 - \delta_2$:

$$\left|V_t(a) - \sigma_a^2\right| \leq 2\sqrt{\sigma_a^2 \cdot \ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{\lfloor n_t(a)/2 \rfloor}} + \frac{4}{3\left\lfloor n_t(a)/2 \right\rfloor} \cdot \ln\frac{2kT\log T}{\delta_2}$$

First of all, we observe that $\lfloor n/2 \rfloor \geq n/3$ when $n \geq 2$ and using this crude approximation we get that:

$$\left|V_t(a) - \sigma_a^2\right| \leq 2\sqrt{3\sigma_a^2 \cdot \ln\frac{2kT\log T}{\delta_1} \cdot \frac{1}{n_t(a)}} + \frac{12}{3n_t(a)} \cdot \ln\frac{2kT\log T}{\delta_2}$$

We move on to splitting cases.

Case 1/: $n_t(a) \geq 64\ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{\sigma_a^2} \Rightarrow 64\ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{n_t(a)} \leq \sigma_a^2$

From the above bound:

$$\begin{aligned}
\left|V_t(a) - \sigma_a^2\right| &\leq 2\sqrt{3\sigma_a^2 \cdot \ln\frac{2kT\log T}{\delta_1} \cdot \frac{1}{n_t(a)}} + \frac{12}{3n_t(a)} \cdot \ln\frac{2kT\log T}{\delta_1} \\
&\leq 2\sqrt{3\sigma_a^2 \cdot \frac{\sigma_a^2}{64}} + \frac{12}{3}\frac{\sigma_a^2}{64} \\
&= \left(\frac{2\sqrt{3}}{8} + 12/192\right)\sigma_a^2 \leq \sigma_a^2/2
\end{aligned}$$

Which proves the first part.

Case 2/: $n_t(a) \leq 64\ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{\sigma_a^2} \Rightarrow \sigma_a^2 \leq 64\ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{n_t(a)}$

Again:

$$\begin{aligned}
\left|V_t(a) - \sigma_a^2\right| &\leq 2\sqrt{3\sigma_a^2 \cdot \ln\frac{2kT\log T}{\delta_1} \cdot \frac{1}{n_t(a)}} + \frac{12}{3n_t(a)} \cdot \ln\frac{2kT\log T}{\delta_2} \\
&\leq 16\sqrt{3 \cdot 64\left(\ln\frac{2kT\log T}{\delta_2} \cdot \frac{1}{n_t(a)}\right)^2} + \frac{16}{3n_t(a)} \cdot \ln\frac{2kT\log T}{\delta_1} \\
&\leq \frac{227}{n_t(a)} \cdot \ln\frac{2kT\log T}{\delta_2}
\end{aligned}$$

□

We move closer to proving the regret bound, we now present the final lemma needed before moving to the analysis of the regret guarantee.

Lemma 2.4.7.

$$UCBV(a;t) := \bar{\mu}_t(a) + \sqrt{8V_t(a) \cdot \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT \log T}{\delta} \cdot \frac{13}{n_t(a)}$$

is an $1 - \delta$ *upper confidence bound for the true expected reward* $\mu(a)$.

Proof. First of all both Lemma 2.4.4 and Lemma 2.4.2 hold with probability $1 - \delta$ by a simple union bound (choose $\delta_1 = \delta_2 = \delta/2$).
When $n_t(a) \leq 64 \ln \frac{4kT \log T}{\delta} \frac{1}{\sigma_a^2}$ we have from Lemma 2.4.6 that:

$$V_t(a) \geq 1/2\sigma_a^2$$

So then:

$$\begin{aligned} UCBV(a;t) &= \bar{\mu}_t(a) + \sqrt{8V_t(a) \cdot \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT \log T}{\delta} \cdot \frac{13}{n_t(a)} \\ &\geq \bar{\mu}_t(a) + 2\sqrt{\sigma_a^2 \cdot \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT \log T}{\delta} \cdot \frac{4}{3n_t(a)} \end{aligned}$$

The last quantity is a $1 - \delta$ UCB for $\mu(a)$ by Lemma 2.4.2.
Now when $n_t(a) \leq 64 \ln \frac{4kT \log T}{\delta} \frac{1}{\sigma_a^2}$, then for the same UCB:

$$\begin{aligned} UCB(a;t) = \bar{\mu}_t(a) &+ \sqrt{2\sigma_a^2 \cdot \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT \log T}{\delta} \cdot \frac{4}{3n_t(a)} \\ &\leq \bar{\mu}_t(a) + \sqrt{2\left(64 \ln \frac{4kT \log T}{\delta} \frac{1}{n_t(a)}\right) \cdot \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}} + \ln \frac{4kT \log T}{\delta} \cdot \frac{4}{3n_t(a)} \\ &\leq \bar{\mu}_t(a) + \ln \frac{4kT \log T}{\delta} \cdot \frac{13}{n_t(a)} \\ &\leq UCBV(a;t) \end{aligned}$$

□

We are finally ready to calculate the regret of the algorithm.

Proof of the **Regret** *guarantee: Theorem 2.4.2.* We will get a handle on the regret as in earlier chapters, by calculating the number of pulls for a suboptimal arm a. All the following arguments hold with probability at least $1 - \delta$. Define as τ_a the last time that such an arm is pulled. Then at that point it must be that:

$$UCBV(a;\tau_a) \geq UCBV(a^*;\tau_a) \tag{2.10}$$

But in the previous lemma we proved that $UCBV$ is a $1 - \delta$ upper confidence bound, so then for any t it holds that:

$$\mu^* \leq UCBV(a;t) = \bar{\mu}_t + rd(a;t) \tag{2.11}$$

Where we define:

$$rd(a;t) := \ln \frac{4kT \log T}{\delta} \cdot \frac{1}{n_t(a)}$$

Notice also that:

$$\bar{\mu}_t \leq \mu + rd(a;t) \tag{2.12}$$

And so combining the above numbered inequalities we have that:

$$\mu^* \leq \mu + 2rd(a;\tau_a) \Leftrightarrow \Delta(a) \leq 2rd(a;\tau_a) \tag{2.13}$$

This prompts us to upper bound the confidence radius $rd(a;t)$. We do this by considering the "high/low variance" cases as in the above lemmas.

Case 1/: $n_t(a) \geq 64\ln\frac{2kT\log T}{\delta_2}\cdot\frac{1}{\sigma_a^2} \Rightarrow 64\ln\frac{2kT\log T}{\delta_2}\cdot\frac{1}{n_t(a)} \leq \sigma_a^2$

Recall that from Lemma 2.4.6 we have:

$$V_t(a) \leq 2\sigma_a^2$$

Which means that:

$$\begin{aligned} rd(a;t) &\leq \sqrt{8\sigma_a^2\cdot\ln\frac{4kT\log T}{\delta}\cdot\frac{1}{n_t(a)}} + \ln\frac{4kT\log T}{\delta}\cdot\frac{13}{n_t(a)} \\ &= \sqrt{8\sigma_a^2\cdot\ln\frac{4kT\log T}{\delta}\cdot\frac{1}{n_{t_a}(a)}} + \left(\sqrt{\ln\frac{4kT\log T}{\delta}\cdot\frac{13}{n_t(a)}}\right)^2 \\ &\leq \sqrt{8\sigma_a^2\cdot\ln\frac{4kT\log T}{\delta}\cdot\frac{1}{n_t(a)}} + \sqrt{\sigma_a^2\ln\frac{4kT\log T}{\delta}\cdot\frac{169}{64n_t(a)}} \\ &\leq 5\sqrt{\sigma_a^2\ln\frac{4kT\log T}{\delta}\cdot\frac{1}{n_t(a)}} \end{aligned}$$

And from Equation (2.13):

$$\Delta(a) \leq O\left(\sqrt{\sigma_a^2\ln\frac{kT}{\delta}\cdot\frac{1}{n_{\tau_a}(a)}}\right)$$

Which when rearranged results in the following bound for the number of pulls:

$$n_T(a) = O\left(\sigma_a^2\ln\frac{kT}{\delta}\cdot\frac{1}{\Delta^2(a)}\right)$$

Now we move on in the "low variance" case below.

Case 2/: $n_t(a) \leq 64\ln\frac{2kT\log T}{\delta_2}\cdot\frac{1}{\sigma_a^2} \Rightarrow \sigma_a^2 \leq 64\ln\frac{2kT\log T}{\delta_2}\cdot\frac{1}{n_t(a)}$

$$n_{\tau_a}(a) \geq 16\ln\frac{2kT}{\delta_1}\cdot\frac{1}{\sigma_a^2}$$

which from Lemma 2.4.6 implies that:

$$V_{\tau_a}(a) \leq 227\ln\frac{4kT\log T}{\delta}\cdot\frac{1}{n_{\tau_a}(a)}$$

Which in turn by the assumption, implies that:

$$rd(a;\tau_a) \leq O\left(\frac{\ln \frac{kT}{\delta}}{n_{\tau_a}(a)}\right)$$

and by the previous analysis:

$$n_T(a) \leq O\left(\ln \frac{kT}{\delta} \cdot \frac{1}{\Delta(a)}\right)$$

Finally, recalling that:

$$R(T) = \sum_{t\in[T]} \Delta(a_t) = \sum_{a\in[k]} n_T(a)\Delta(a)$$

we have the result:

$$R(T) = O\left(\ln \frac{kT}{\delta} \cdot \sum_{\substack{a\in[k]:\\ \mu(a)<\mu^*}} \left\{\frac{\sigma_a^2}{\Delta(a)} + 1\right\}\right)$$

□

Chapter 3

Adversarial MAB (and Online Learning)

3.1 Introduction

What do you do when you can make no assumptions? In that scenario you might as well assume that there is an adversary that has one goal in mind: to ruin your plans. This very general setting is known as the *adversarial* (or non-stochastic) MAB problem, first considered by Auer et al in [6].

Let's for a minute assume that the learner is using a simple stochastic MAB algorithm, like Algorithm 1. In *ETC* the learner plays each arm $\tilde{O}\left(T^{2/3}\right)$ times, then chooses the seemingly best one and plays it for the remainder of the time. An adversary can very easily fool this type of learner by making sure that all the arms but one look terrible (rewarding 0) in the first stage and then switching this up. The learner will always suffer linear regret (with respect to the best-in-hindsight arm).

Actually in general, in a very similar fashion one can show that any deterministic algorithm [1] cannot guarantee sublinear regret in expectation. This can happen even with an adversary that is not responsive to the learner's actions, meaning they are *oblivious*. Such an adversary can be deterministic or randomized (a special case of the second is IID rewards, ie stochastic MAB).

An adversary that responds (adapts) to the learner's actions (the algorithm's choices) is called *adaptive*. An adaptive adversary encapsulates many environments where a user's actions alter the said environment. For example in game theoretic applications, self-interested parties adapt to others' choices to maximize their utility.

We will consider regret with respect to the best-in-hindsight arm, meaning:

$$a^* := \underset{a \in [k]}{\text{argmax}} \sum_{t=1}^{T} r_t(a)$$

Rather surprisingly we will prove the following theorem:

[1] Given a history it will always make the same choice

Theorem 3.1.1. *There exists an algorithm for the adversarial MAB problem that guarantees expected regret:*

$$\mathbb{E}\left[R(T)\right] = O\left(\sqrt{kTlogk}\right)$$

This seemingly much harder problem (than the simple stochastic MAB problem) admits an algorithm achieving a regret guarantee, almost matching the theoretical lower bound of the instance-independent regret for stochastic MAB. However in most scenarios well-known stochastic bandits algorithms will outperform this algorirm due to their strong instance-dependent logarithmic regret. The regret achieved by this algorithm [6] is also near-optimal for the adversarial problem as well, as shown in the same work.

Theorem 3.1.2. *The expected regret of any policy for an adversarial MAB problem is*

$$\mathbb{E}\left[R(T)\right] \geq \Omega\left(\sqrt{kT}\right)$$

3.2 Full Feedback

Before unveiling the algorithm, we will need to cover some prerequesite ground. Let's assume a slightly simpler problem: every arm reveals its reward at the end of the round. We will continue to assume nothing about the rewards (ie assume an adversary generating them). In its more general form, this is called the *Online Learning Problem* [39, 49].
There is a well known algorithm for the full-feedback case which is known as *Multiplicative Weights Update* or *Hedge* in the literature [22, 32], first described in its more well known form by Littlestone and Warmuth [32]. Hedge is based on the also well known *Weighted Majority Vote* [45] algorithm for the *Binary Prediction* problem, by Littlestone and Warmuth. We also note that due to these roots, *MWU* and its analysis follow a *cost* framework, where each arm instead incurs a loss on the learner, prompting them to instead *minimize* their cumulative losses.

Algorithm 5 MWU/Hedge

```
Parameter $\epsilon \in (0, 1/2)$
For all $a \in k$ : $w_1(a) = 1$
for $t \in [T]$ do
    $p_t(a) = \frac{w_t(a)}{\sum_{\tilde{a} \in [k]} w_t(\tilde{a})}$
    Play arm $a_t$ w.p. $p_t(a)$
    $w_{t+1}(a) = w_t(a) \cdot (1 - \epsilon)^{1 - r_t(a)}$
```

As we noted, to have a chance of sub-linear regret, an algorithm's choices would have to be random, such an algorithm is called *randomized. Multiplicative Weights Update* is a randomized algorithm, as at every round it chooses some arm probabilistically, sampling from the distribution defined by $p_t(\cdot)$. Intuitively, the algorithm associates a weight with each arm and plays that arm with probability proportional to that weight; an arm with a higher weight is more likely to be chosen. These weights are updated in a multiplicative fashion at each round, for every arm, as there is full feedback at the end of the round.

The multiplicative factor scales exponentially with the deviation from the optimal reward, so highly suboptimal arms get 'punished' quickly.

However as the multiplicative factor is never zero, the same holds for the pull probability and so even arms that have not performed well get recourse. We begin our analysis by arguing first about the multiplicative decrease in the total weight of all the arms at each round.

Lemma 3.2.1. *The* total weight $W_t := \sum_{a\in[k]} w_t(a)$ *at each round obeys the following properties:*

$$W_{T+1} > (1-\epsilon)^{T-R^*} \tag{3.1}$$

where:

$$R^* := \sum_{i=1}^{T} r_t(a^*) \text{ \textit{(ie the cumulative reward of the best in hindsight arm)}}$$

and:

$$\frac{W_{t+1}}{W_t} \leq 1 - \epsilon \cdot \mathbb{E}\left[1 - r_t(a_t) \mid \vec{w}_t\right] \tag{3.2}$$

Proof. The first property is easy enough, note that:

$$\begin{aligned} W_{T+1} &> w_{T+1}(a^*) \\ &= w_1(a)\Pi_{t=1}^{T}(1-\epsilon)^{1-r_t(a^*)} \\ &= (1-\epsilon)^{1-\sum_{t=1}^{T} r_t(a^*)} \\ &= (1-\epsilon)^{1-R^*} \end{aligned}$$

We proceed to the second property.

$$\begin{aligned} \frac{W_{t+1}}{W_t} &= \sum_{a\in[k]} \frac{w_t(a)}{W_t} \cdot (1-\epsilon)^{1-r_t(a)} \\ &= \sum_{a\in[k]} p_t(a) \cdot (1-\epsilon)^{1-r_t(a)} \\ &\leq \sum_{a\in[k]} p_t(a) \cdot (1 - \epsilon \cdot (1 - r_t(a))) \qquad \text{since } (1-\epsilon)^x \leq 1-\epsilon^x \text{ for } x \in [0,1] \\ &= 1 - \epsilon \cdot \mathbb{E}\left[1 - r_t(a_t) \mid \vec{w}_t\right] \end{aligned}$$

□

We are ready to prove the main result, *MWU*'s regret guarantee.

Theorem 3.2.1. Multiplicative Weight's Update *with parameter* $\epsilon = \sqrt{\frac{\ln k}{T}}$ *guarantees expected regret:*

$$\mathbb{E}\left[R(T)\right] \leq O\left(\sqrt{T \ln k}\right)$$

Proof. By Lemma 3.2.1 we have that:

$$\ln \frac{W_{t+1}}{W_t} \leq \ln\left(1 - \epsilon \cdot \mathbb{E}\left[1 - r_t(a_t) \mid \vec{w}_t\right]\right) \leq -\epsilon \cdot \mathbb{E}\left[1 - r_t(a_t) \mid \vec{w}_t\right]$$

Summing across the rounds we have:

$$\begin{aligned}\sum_{t\in[T]} \epsilon \cdot \mathbb{E}\left[1 - r_t(a_t) \mid \vec{w}_t\right] &\leq - \sum_{t\in[T]} \ln \frac{W_{t+1}}{W_t} \\ &= - \ln \frac{W_{T+1}}{W_1} \\ &\leq \ln k - \ln{(1-\epsilon)}^{T-R^*} && \text{by } \textit{Lemma } 3.2.1 \\ &= \ln k - \ln{(1-\epsilon)} \cdot (T - R^*)\end{aligned}$$

If we take expectations, we have that:

$$\epsilon T - \epsilon \cdot \mathbb{E}\left[\sum_{t\in[T]} r_t(a_t)\right] \leq \ln k - \ln{(1-\epsilon)} \cdot (T - \mathbb{E}\left[R^*\right])$$

We use the below identity:

$$-\frac{1}{\epsilon} \ln{(1-\epsilon)} \leq 1 + \epsilon, \text{ for } \epsilon \in [0, 1/2]$$

And rearanging we have that:

$$T - \mathbb{E}\left[\sum_{t\in[T]} r_t(a_t)\right] \leq \frac{1}{\epsilon} \ln k + (1+\epsilon) \cdot (T - \mathbb{E}\left[R^*\right])$$

Or:

$$\begin{aligned}\mathbb{E}\left[R^*\right] - \mathbb{E}\left[\sum_{t\in[T]} r_t(a_t)\right] &\leq \frac{1}{\epsilon} \ln k + \epsilon \cdot (T - \mathbb{E}\left[R^*\right]) \\ &\leq \frac{1}{\epsilon} \ln k + \epsilon T\end{aligned}$$

We identify that the left hand side is just the expected regret of the algorithm and we choose

$$\epsilon = O\left(\sqrt{\frac{\ln k}{T}}\right)$$

One can note that we have a requirement for ϵ already, namely that $\epsilon \in (0, 1/2)$. But $\ln k$ should be asymptotically smaller than T for meaningful regret guarantees (else no algorithm can guarantee sublinear regret) and we can guarantee that $\epsilon < 1/2$ by multiplying the above fraction by a suitable constant, satisfying the requirement.
By simple calculations, the expected regret then is:

$$\mathbb{E}\left[R(T)\right] \leq O\left(\sqrt{T \ln k}\right)$$

□

Notice that the proof does not assume anything about how the rewards are generated. We link the best-in-hindisight arm's performance to the algorithm's via the total weight's change at each round up until the last one, which is of course, dependent on the algorithm's choices. A crucial point here is that the algorithm will always get feedback by the 'best' arm. A final note is on the comparison with the 'loss' framework that we mentioned. The quantity $1 - r_t(a)$ and its cumulative sum through the rounds $T - \sum_{t\in[T]} r_t(a)$ can be immediately seen as instantaneous and cumulative *losses* respectively.

3.3 Bandit Feedback and Sublinear regret through EXP3

We return to the adversarial MAB problem and bandit feedback, having seen the full feedback equivalent (*online learning* with bounded at-most-unit rewards). An ingenious idea is to reduce bandit feedback to full feedback and employ the *MWU* algorithm that we just saw. This is captured in the *EXP3* algorithm (stands for "Exponential weight, Exploration, Exploitation)" introduced by Aurer et al. in [6].

Algorithm 6 Exp3

```
Parameters $\gamma, \eta \in (0, 1/2)$
For all $a \in k$ : $w_1(a) = 1$
for $t \in [T]$ do
    Play arm $a$ w.p. $p_t(a) = (1-\gamma)\frac{w_t(a)}{\sum_{a' \in [k]} w_t(a')} + \gamma/k$
    Receive $r_t(a)$
    Set $\hat{r}_t(a) = \mathbb{I}\{a_t = a\} \cdot \frac{r_t(a)}{p_t(a)}$
    Set $w_{t+1}(a) = w_t(a) \cdot \exp\{\{\eta \hat{r}_t(a)\}\}$
```

Let's analyze what the algorithm does step by step. First of all, we notice that at every round the probability that an arm is chosen is at least the constant γ/k. So, with probability γ the algorithm will *explore* at a given round. With probability $1-\gamma$ we have exploitation; the algorithm basically calls Hedge/MWU, tracking a weight for each arm, that never decreases. Another difference is the rewards that this variant of the *MWU* "sees". As the algorithm is in a bandit feedback environment, the only reward it knows at the end of the round is the one from the chosen arm, implying a *bias* on the experienced rewards. This bias is counteracted by dividing by the bias probability $p_t(a)$; the probability that a certain arm is chosen. To guarantee that $\hat{r}_t(a)$ is on expectation $r_t(a)$, it is set as zero when a is not chosen (and its reward not seen).
We begin the analysis by making the previous argument concrete.

Lemma 3.3.1. *$\hat{r}_t(a)$ is an unbiased estimator for $r_t(a)$, ie*

$$\mathbb{E}\left[\hat{r}_t(a)\right] = r_t(a)$$

Proof.

$$\mathbb{E}\left[\hat{r}_t(a)\right] = \mathbb{P}\left(a_t = a\right) \cdot \frac{r_t(a)}{p_t(a)} = r_t(a)$$

□

Define $W_t := \sum_{a \in [k]} w_t(a)$, we will attempt to upper bound the ratio of change from the previous round as in *MWU*. We will prove the following lemma:

Lemma 3.3.2. *When $\eta = \gamma/k$ holds that:*

$$\frac{W_{t+1}}{W_t} \le 1 + \frac{\gamma/k}{1-\gamma}\mathbb{E}\left[\hat{r}_t(a_t) \mid \vec{w}_t\right] + (e-2)\frac{(\gamma/k)^2}{1-\gamma}\mathbb{E}\left[\hat{r}_t(a_t)^2 \mid \vec{w}_t\right]$$

Proof.

$$\begin{aligned}\frac{W_{t+1}}{W_t} &= \sum_{a\in[k]} \frac{w_{t+1}(a)}{W_t} \\ &= \sum_{a\in[k]} \frac{w_t(a)}{W_t} \exp\{\eta \cdot \hat{r}_t(a)\} \\ &= \frac{p_t(a) - \gamma/k}{1-\gamma} \exp\{\eta \cdot \hat{r}_t(a)\}\end{aligned}$$

We would like to bound the exponential by a polynomial expression in $\eta \cdot \hat{r}_t(a)$. We have that:

$$e^x \leq 1 + x + (e-2)x^2 \text{ for } x \leq 1$$

So choosing $\eta = \gamma/k$ we guarantee that $\eta \cdot \hat{r}_t(a) \leq r_t(a) \leq 1$ and that:

$$\exp\{\eta \cdot \hat{r}_t(a)\} \leq 1 + \frac{\gamma}{k}\hat{r}_t(a) + (e-2)\left(\frac{\gamma}{k}\hat{r}_t(a)\right)^2$$

We have then that:

$$\begin{aligned}\frac{W_{t+1}}{W_t} &\leq \sum_{a\in[k]} \frac{p_t(a) - \gamma/k}{1-\gamma}\left[1 + \frac{\gamma}{k}\hat{r}_t(a) + (e-2)\left(\frac{\gamma}{k}\hat{r}_t(a)\right)^2\right] \\ &\leq \sum_{a\in[k]} \left\{\frac{w_t(a)}{W_t} + \frac{p_t(a)}{1-\gamma}\left(\frac{\gamma}{k}\hat{r}_t(a)\right) + \frac{p_t(a)}{1-\gamma}(e-2)\left(\frac{\gamma}{k}\hat{r}_t(a)\right)^2\right\} \\ &= 1 + \frac{\gamma/k}{1-\gamma}\sum_{a\in[k]} p_t(a)\hat{r}_t(a) + (e-2)\frac{(\gamma/k)^2}{1-\gamma}\sum_{a\in[k]} p_t(a)\hat{r}_t(a)^2\end{aligned}$$

□

Lemma 3.3.3. *The following hold:*

$$\mathbb{E}\left[\hat{r}_t(a_t) \mid \vec{w}_t\right] = r_t(a_t)$$

and:

$$\mathbb{E}\left[\hat{r}_t(a_t)^2 \mid \vec{w}_t\right] \leq \hat{r}_t(a_t) = \sum_{a\in[k]} \hat{r}_t(a)$$

Proof. Notice that:

$$\mathbb{E}\left[\hat{r}_t(a_t) \mid \vec{w}_t\right] = \sum_{a\in[k]} p_t(a)\hat{r}_t(a) = \sum_{a\in[k]} p_t(a) \cdot \mathbb{I}\{a_t = a\}\frac{r_t(a)}{p_t(a)} = r_t(a_t)$$

On the other hand:

$$\mathbb{E}\left[\hat{r}_t(a_t)^2 \mid \vec{w}_t\right] = \sum_{a\in[k]} p_t(a)\hat{r}_t(a)^2 = \sum_{a\in[k]} p_t(a)\mathbb{I}\{a_t = a\}\frac{r_t(a)}{p_t(a)}\hat{r}_t(a) = r_t(a_t) \cdot \hat{r}_t(a_t) \leq \hat{r}_t(a_t)$$

Finally:

$$\sum_{a\in[k]} \hat{r}_t(a) = \sum_{a\in[k]} \mathbb{I}\{a_t = a\}\frac{r_t(a)}{p_t(a)} = \hat{r}_t(a_t)$$

□

We introduce the algorithm's performance and relate it to the previous quantities (bound it from below) through the following lemma:

Lemma 3.3.4. *Exp3's cumulative reward $G_{exp3} := \sum_{t\in[t]} r_t(a_t)$ obeys:*

$$G_{exp3} \geq \sum_{t\in[T]} \hat{r}_t(a^*) - \frac{(e-1)\gamma}{k} \sum_{t\in[T]} \sum_{a\in[k]} \hat{r}_t(a) - \frac{k \ln k}{\gamma}$$

Proof. By Lemma 3.3.2 and Lemma 3.3.3 we have that:

$$\frac{W_{t+1}}{W_t} \leq 1 + \frac{\gamma/k}{1-\gamma} r_t(a_t) + \frac{(e-2)(\gamma/k)^2}{1-\gamma} \sum_{a\in[k]} \hat{r}_t(a)$$

Since $\ln x \leq x-1, x>0$:

$$\ln \frac{W_{t+1}}{W_t} \leq \frac{W_{t+1}}{W_t} - 1 \leq \frac{\gamma/k}{1-\gamma} r_t(a_t) + \frac{(e-2)(\gamma/k)^2}{1-\gamma} \sum_{a\in[k]} \hat{r}_t(a)$$

And using the usual trick, summing over t:

$$\ln \frac{W_{T+1}}{W_1} \leq \frac{\gamma/k}{1-\gamma} G_{exp3} + \frac{(e-2)(\gamma/k)^2}{1-\gamma} \sum_{t\in[T]} \sum_{a\in[k]} \hat{r}_t(a)$$

To get "rid of" the total weights, we lower bound them, as for any arm a it holds that:

$$\ln \frac{W_{T+1}}{W_1} \geq \ln \frac{w_{T+1}(a)}{W_1} = \frac{\gamma}{k} \sum_{t\in[T]} \hat{r}_t(a) - \ln k$$

And so using this lower bound in the immediately previous inequality:

$$\frac{\gamma/k}{1-\gamma} G_{exp3} + \frac{(e-2)(\gamma/k)^2}{1-\gamma} \sum_{t\in[T]} \sum_{a\in[k]} \hat{r}_t(a) \geq \frac{\gamma}{k} \sum_{t\in[T]} \hat{r}_t(a) - \ln k$$

Rearranging and noting that the arm a was arbitrary:

$$G_{exp3} \geq (1-\gamma) \sum_{t\in[T]} \hat{r}_t(a^*) - (e-2)\frac{\gamma}{k} \sum_{t\in[T]} \sum_{a\in[k]} \hat{r}_t(a) - \frac{k \ln k}{\gamma}$$

□

We are ready to move to the big theorem, Exp3's regret.

Theorem 3.3.1. *Exp3 attains the below regret-like bound:*

$$G_{max} - \mathbb{E}\left[G_{exp3}\right] \leq \gamma(e-1)G_{max} + \frac{1}{\gamma} k \ln k$$

Also, choosing $\gamma = \min\left\{1/2, \sqrt{\frac{k \ln k}{(e-1)T}}\right\}$ we have that:

$$G_{max} - \mathbb{E}\left[G_{exp3}\right] = O\left(\sqrt{kT \ln k}\right)$$

Where we define $G_{max} := \max_{a\in[k]} \sum_{t\in[T]} r_t(a)$.

Note that the above difference is the expected regret that *Exp3* attains if we assume a deterministic oblivious adversary. Even though we prove this slightly weaker bound, the expected regret satisfies the same bound, for any adversary [6]. We move on the proof of the theorem.

Proof. From Lemma 3.3.4 we have that:

$$G_{exp3} \geq (1-\gamma) \sum_{t \in [T]} \hat{r}_t(a^*) - (e-2)\frac{\gamma}{k} \sum_{t \in [T]} \sum_{a \in [k]} \hat{r}_t(a) - \frac{k \ln k}{\gamma}$$

Taking expectations with respect to the history $\mathcal{H}_T := \{a_1, \ldots, a_T\}$ and noting that $\mathbb{E}\left[\hat{r}_t(a) \mid \mathcal{H}_{t-1}\right] = r_t(a)$ we have that:

$$\mathbb{E}\left[G_{exp3} \mid \mathcal{H}_t\right] \geq (1-\gamma)G_{max} - (e-2)\frac{\gamma}{k} \sum_{t \in [T]} \sum_{a \in [k]} r_t(a) - \frac{k \ln k}{\gamma}$$

Now noting that $\sum_{t \in [T]} r_t(a) \leq G_{max}$ and that the history is arbitrary, we have the below:

$$\mathbb{E}\left[G_{exp3}\right] \geq (1-\gamma)G_{max} - (e-2)\frac{\gamma}{k} \sum_{a \in [k]} G_{max} - \frac{k \ln k}{\gamma}$$

We've proved the first part. Now choosing $\gamma = \min\left\{1/2, \sqrt{\frac{k \ln k}{(e-1)T}}\right\}$ it is a matter of calculations to prove the second part as well. □

Chapter 4

Stochastic Bandits with Adversarial Corruptions

4.1 Introduction

So far we've seen *Stochastic* and *Adversarial* Bandits and the stark contrast between these environments, the methods and algorithms used to tackle these problems and the corresponding results (regret guarantees). We have seen algorithms that provide strong logarithmic (in the time horizon) instance-dependent guarantees in the first case and guarantees of order $O(\sqrt{T})$ in the second case (which also is an instance-independent bound for algorithms in the Stochastic case). These two settings are extremely orthogonal, with one being overoptimistic in assuming that all rewards are sampled from the same distribution, while the other is too pessimistic in order to protect from any adversary.

Stochastic Bandits with Adversarial Corruptions as a problem was first introduced by Lykouris et al. in [46] to capture environments where the underlying stochastic structure of the rewards is corrupted by adversarial attacks. The goal in this problem is to construct algorithms that take advantage of the (mostly) stochastic underlying structure and are robust to these adversarial attacks, gracefully transitioning from the optimal instance-dependent regret bounds to ”worse” guarantees, as the corruption increases.

The study of this environment was motivated by the increasing phenomenon of click fraud in search-engine advertisement allocation scenarios. It was observed that a large group of users (likely *bots*) would engage in very similar actions targeting specific ad banners. Another motivating real-life example cited in [46] is that of *spam* and malicious reviews in recommendation systems.

A similar, but different line of work is that of designing *Best of Both Worlds* algorithms. This direction necessitates the design of algorithms that behave well when the environment is stochastic or adversarial in nature. The works of [60, 61, 17, 5] achieve almost-optimal (meaning up to logarithmic factors) pseudo-regret guarantees for the stochastic case and the optimal regret guarantee in the adversarial case. However, these algorithms are overly pessimistic in an environment such as the one we are studying, where the stochastic nature is corrupted by a fair amount.

Another active area of research is that of extending and improving guarantees for stochastic settings. For example, Audibert and Bubeck in [2] provide and algorithm that attains the optimal non-distribution-based upper bound for stochastic bandits while retaining the optimal distribution-based stochastic guarantee.

4.2 Adversarial corruptions

We begin by modeling the environment and some definitions.

Definition 4.2.1. Stochastic MAB with Adversarial Corruptions
1. The learner has k available actions at each round t and is given T rounds in total.
2. At each round t the learner chooses an action a_t with stochastic reward $\tilde{r}_t(a_t)$.
3. The adversary "corrupts" the reward by injecting corruption $c_t(a_t)$.
4. The learner observes the reward $r_t(a_t) = \tilde{r}_t(a_t) + c_t(a_t)$.
5. The learner strives to minimize her pseudo-regret.

The amount of corruption C is measured in the following sense.

Definition 4.2.2. Corruption C

$$C := \sum_{t \in [T]} \max_{a \in [k]} |c_t(a)|$$

Stochastic algorithms like *AAE* or *UCB* do not fare well at this problem, as even with moderate corruption (logarithmic in time horizon) they could "eliminate" the optimal arm very quickly from their future choices, as shown below.

Observation 4.2.1. *Logarithmic corruption can make* AAE *suffer linear regret in expectation.*

Proof. Consider a MAB problem with two arms and expected rewards $\mu_1 = 1$ and $\mu_2 = \epsilon$ respectively. Now consider an adversary that injects corruption only in the first $128 \log(2kT)$ rounds in the following way:

$$c_t(a_1) = -\tilde{r}_t(a_1) \text{ and } c_t(a_2) = 1 - \tilde{r}_t(a_2)$$

The algorithm alternates the arms until some arm is *confidently* worse than the other, meaning that if $LCB(a) > LCB(a')$, then arm a' is 'deactivated' and the algorithm plays only arm a afterwards. *AAE* with $\delta = 1/T^2$ guarantees sublinear expected regret in a fully stochastic environment, but after these $128 \log(2kT)$ rounds we have that:

$$UCB(a_1) = \sqrt{\frac{2 \log(2kT/T^{-2})}{n_t(a_1)}} = 1/4$$

and for arm a_2:

$$LCB(a_2) = 1 - \sqrt{\frac{2 \log(2kT/T^{-2})}{n_t(a_2)}} = 3/4$$

So arm a_1 will have been deactivated by this time (arm a_2 will have UCB strictly greater than a_1's LCB through the course of the algorithm). *AAE* will only play arm a_2 from then on, accruing regret:

$$R(T) \geq (1-\epsilon)(T - 128\log(2kT) = \Omega((1-\epsilon)T)$$

which is linear for constant $\epsilon \in [0,1)$. □

On the other hand an adversarial algorithm like *Exp3* would work (have sublinear regret) and attain its regret bound, but it would not take advantage of the underlying stochastic nature of the problem in moderate or light corruption scenarios. For example if the above corruption was very 'spread out' across the rounds, *AAE* would fare very well. Evidently these observations further motivate the construction of new algorithms.

4.3 Multi Layer Active Elimination Race

Lykouris et al. note that if the learner somehow knew the corruption C or at least a close upper bound, then she could "play safe" and enlarge the confidence radius. Specifically:

Theorem 4.3.1. AAE *with confidence radius*

$$rad(a;t) = \sqrt{\frac{2\ln(2kT/\delta)}{n_t(a)}} + \frac{C}{n_t(a)}$$

has regret $O\left(\sum_{a \neq a^*} \frac{\ln\frac{kT}{\delta} + C}{\Delta(a)}\right)$, *where* C *is an upper bound for the corruption.*

We will not prove this theorem rigorously, but we will provide some intuition as to why it holds. By Azuma Hoeffding the expected reward lies in the confidence interval centered on the stochastic part of the average reward with radius the first addend of $rad(a;t)$ with probability at least $1-\delta$. Since the realized average reward might be corrupted by at most C, the expected reward lies in the confidence interval of radius $rad(a;t)$ centered on the realized average reward (and vice versa, which is what we care about). All of this means that this modified version of *AAE* will work correctly and will never deactivate the optimal arm (since the algorithm's CB is an actual 1 - δ CB). The proof is finished by showing that every suboptimal arm a will be eliminated after $O\left(\frac{\ln\frac{kT}{\delta} + C}{\Delta(a)^2}\right)$ pulls with probability at least $1-\delta$.

However, even if there is no corruption at all *AAE* with the above enlarged radius would attain the same regret bound. Another idea which is a precursor to the main algorithm of [46] is to maintain two instances of AAE. One would be meant for the purely stochastic case and be quick and decisive in its decisions, while the other would be slower and correct even in a corrupted scenario. Lykouris et al achieve that by maintaining a 'fast' instance of *AAE* which has the typical confidence radius and a 'slow' instance which is chosen with a small probability and has an enlarged radius. An algorithm sketch follows.

Algorithm 7 Fast-Slow AAE

```
for $t \in [T]$ do
    $rd_F(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^F(a)}}$
    $rd_S(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^S(a)}} + \frac{2\ln\frac{2kT}{\delta}}{n_t^S(a)}$
    With probability $1/C$ use the 'slow' AAE ($S$), else use the 'fast' ($F$).
    If $S$ eliminates an arm $a$, eliminate the same arm in $F$.
    If $F$ has no arms to play, pull a random one from the ones still active in $S$. ▷ Without updating anything in S
```

Theorem 4.3.2. *Algorithm 7 attains regret* $O\left(\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$ *in the purely stochastic case and* $O\left(kC\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$ *in the* C*-corrupted case, with probability at least* $1-\delta$.

Again, we will provide intuition in to why the algorithm works, instead of proving the above theorem rigorously. Since the slow instance is chosen with probability $1/C$ it will experience constant corruption in expectation, but to have high probability guarantees we need to correct for the deviation and as such the radius is enlarged by $\frac{2\ln\frac{2kT}{\delta}}{n_t^F(a)}$ (by a martingale concentration inequality the experienced corruption would be at most $2\ln\frac{2kT}{\delta}$). In a purely stochastic scenario it is easy to show that the regret is at most $O\left(\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$. To bound the regret in the C-corrupted case we need to bound how many times a suboptimal arm can be played in the 'fast' instance. In expectation it is played at most kC times more than in the 'slow' instance and to have a high probability guarantee another logarithmic factor is needed, leading to the $O\left(kC\ln\frac{kT}{\delta}\sum_a \frac{1}{\Delta(a)}\right)$ regret bound.

We move to the main algorithm in [46], which combines all the above ideas in the case of agnostic corruption level C. Since C is unknown one could perform an 'upwards' binary search for its value. In their algorithm, Lykouris et al maintain $\log T$ instances of AAE with enlarged confidence intervals bar the first one, as in the 'fast-slow' algorithm. Each instance is chosen with a probability that gets exponentially smaller (as such combating corruption up to the reciprocal of the probability). To be clearer we define a sketch of the algorithm below.

Algorithm 8 Multi-layer Active Arm Elimination Race

```
for $t \in [T]$ do
    $rad_0(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^0(a)}}$
    $rad_l(a;t) \leftarrow \sqrt{\frac{2\ln\frac{4kT}{\delta}}{n_t^\ell(a)}} + \frac{2\ln\frac{2kT}{\delta}}{n_t^\ell(a)}$ for $\ell \in \{1,\dots,\lfloor\log T\rfloor\}$
    With probability $(1/2)^{\ell+1}$ choose the $\ell$-th layer algorithm and with the remaining probability choose $\ell = 0$.
    If layerl $\ell$ eliminates an arm $a$, eliminate the same arm in all layers $\ell' < \ell$.
    If a chosen layer has no arms, play an active arm from the closest higher layer that has active arms.
```

Theorem 4.3.3. Multi-layer Active Arm Elimination Race *attains regret*

$$O\left(\ln\frac{kT}{\delta}\sum_{a\neq a^*}\frac{kC\ln\frac{kT}{\delta}+\log T}{\Delta(a)}\right)$$

with probability at least $1-\delta$.

We will provide a sketch of the proof. As there will exist a layer with $2^\ell \geq C$ (since $2^{logT} = T \geq C$, we might be off only by a factor of 2), that layer and all the ones above it will behave stochastically and accrue regret at most $\ln\frac{kT}{\delta}\sum_a\frac{1}{\Delta(a)}$ with probability at least $1-\delta$. All the layers are at most $\log T$ and so that takes care of the second addend in the regret bound. The first addend is a result of bounding the number of pulls of suboptimal arms in the 'fast' and inaccurate layers below ℓ. This is done in a similar fashion as previously in the 'slow-fast' algorithm.

4.4 BARBAR algorithm

While the algorithm Lykouris et al present is beautiful and intuitive, the regret it accrues has a multiplicative dependence on the corruption, which results in linear worst-case regret for corruption $C = \Omega(\sqrt{T})$. Gupta et al in their work [36] provide an algorithm that trades the multiplicative dependence on kC for an additive one, tolerating significantly more corruption: that is up to $o(T)$. The algorithm is agnostic to the corruption level and is fairly simple as well.

Their algorithm works in epochs, which are logarithmically many (in the time horizon). Each epoch m uses an estimation $\Delta_m(a)$ of each arm's gap using the previous epoch's results and pulls each arm according to this estimation (in expectation as many times as UCB would if that gap were the true one $\Delta_m(a)^{-2}$). No arm is pulled more than 2^{2m} times and each decision is probabilistic giving seemingly 'bad' arms some recourse (hence the name *Bandit Algorithm with Robustness: Bad Arms get Recourse*). Finally, arguably the most important ingredient of the algorithm is that each epoch uses information only from the preceding epoch, which means that corruption has a bounded effect. The algorithm follows.

Algorithm 9 BARBAR

```
$\lambda = 1024 \ln(\frac{8k}{\delta} \log T)$
$T_0 = 1$
for $a \in [k]$ do
    $\Delta_0(a) = 1$
for $m \in \{1, 2, ...\}$ do
    $\bar{n}_m(a) = \lambda \cdot \Delta_{m-1}^{-2}(a)$
    Set $N_m = \sum_{a \in [k]} \bar{n}_m(a)$ and $q_m(a) = \frac{\bar{n}_m(a)}{N_m}$
    Set $T_m = T_{m-1} + N_m$ and $E_m = [T_{m-1}, T_m]$
    for $t \in E_m \cap [T]$ do
        play arm $a$ w.p. $q_m(a)$
    $\mu_m(a) = \frac{1}{\bar{n}_m(a)} \sum_{t \in E_m} \mathbb{I}\{a_t = a\} r_t(a)$
    $a_m^* = \text{argmax}_{a \in [k]} \{\mu_m(a) - \frac{1}{16}\Delta_{m-1}(a)\}$
    $\mu_m^* = \mu_m(a_m^*) - \frac{1}{16}\Delta_{m-1}(a)$
    $\Delta_m(a) = \max\{2^{-m}, \mu_m^* - \mu_m(a)\}$
```

Gupta et al go on to prove the following regret guarantee for their algorithm.

Theorem 4.4.1. BARBAR *attains regret*

$$R(T) \leq O\left(kC + \log(T) \cdot \log\left(\frac{k}{\delta} \log T\right) \sum_{a \neq a^*} \frac{1}{\Delta(a)}\right)$$

with probability at least $1 - \delta$

It is evident that this is a much improved regret guarantee than the one in [46].

We will follow a slightly different trajectory towards proving the algorithm's regret bound compared to the original work. The first course of action is to investigate the deviation of the empirical reward $\mu_m(a)$ from the actual expected reward $\mu(a)$. We will prove the following lemma.

Lemma 4.4.1. *For a fixed arm a and epoch m we have that:*

$$\mathbb{P}\left(|\mu_m(a) - \mu(a)| \geq \sqrt{\frac{4\ln 4/\beta}{\bar{n}_m(a)}} + \frac{2C_m}{N_m}\right) \leq \beta/2$$

where $C_m := \max_a \sum_{t\in E_m} |c_t(a)|$.

In other words the deviation from the true expectation is a sum that depends on the previous epoch's empirical gap and basically the "average" corruption at this epoch. When the corruption is low and the previous epoch is accurate, then the empirical gap of the epoch will be very close to the true one (may be off by a constant factor only).

Proof. The proof begins by noting that:

$$\mu_m(a) = \frac{1}{\bar{n}_m(a)} \sum_{t\in E_m} \mathbb{I}\{a_t = a\}(\tilde{r}_t(a) + c_t(a))$$

Then Gupta et al break the sum, analyzing the stochastic reward and the corruption separately, defining:

$$A_m(a) := \sum_{t\in E_m} \mathbb{I}\{a_t = a\}\tilde{r}_t(a) \text{ and } B_m(a) := \sum_{t\in E_m} \mathbb{I}\{a_t = a\}c_t(a)$$

At this point we condition on the previous epoch's quantities, making $n_m(a), N_m, T_{m-1}$ deterministic.
By a simple Chernoff-Hoeffding bound Theorem A.0.2 the reader can verify that:

$$\mathbb{P}\left(\left|\frac{A_m(a)}{\bar{n}_m(a)} - \mu(a)\right| \geq \sqrt{\frac{3\ln\frac{4}{\beta}}{\bar{n}_m(a)}}\right) \leq \beta/4$$

In order to bound the experienced corruption we consider the sequence $Y_t = (\mathbb{I}\{a_t = a\} - q_m(a))\, c_t(a)$. Assuming that we have a deterministic adversary, then proving that this sequence is an MDS is easy enough. To apply the Freedman-type inequality Theorem A.0.2 we bound the predictable variation:

$$\begin{aligned}
\mathcal{V} &:= \sum_{t\in E_m} \mathbb{E}\left[Y_t^2 \mid \mathcal{H}_{t-1}\right] \\
&\leq \sum_{t\in E_m} |c_t(a)| \mathbb{V}ar\left(\mathbb{I}\{a_t = a\}\right) \\
&= (q_m(a) - q_m^2(a)) \sum_{t\in E_m} |c_t(a)| \\
&\leq q_m(a) \sum_{t\in E_m} |c_t(a)|
\end{aligned}$$

Applying the Freedman-type equality Theorem A.0.6 we have that with probability at least $1 - \beta/4$:

$$\frac{1}{\bar{n}_m(a)} B_m(a) \leq \frac{q_m(a)}{n_m(a)} \sum_{t \in E_m} c_t(a) + \frac{1}{\bar{n}_m(a)} \left(\mathcal{V} + \ln \frac{4}{\beta} \right) \leq 2 \frac{q_m(a)}{\bar{n}_m(a)} \sum_{t \in E_m} |c_t(a)| + \frac{\ln \frac{4}{\beta}}{\bar{n}_m(a)}$$

Noting that $q_m(a) = \bar{n}_m(a)/N_m$, $\sum_{t \in E_m} |c_t(a)| \leq C_m$ and given $\bar{n}_m(a) >= \lambda > \ln \frac{4}{\beta}$ which means that $\frac{\ln \frac{4}{\beta}}{\bar{n}_m(a)} \leq \sqrt{\frac{\ln \frac{4}{\beta}}{\bar{n}_m(a)}}$ we have that:

$$\mathbb{P}\left(\left| \frac{B_m(a)}{\bar{n}_m(a)} \right| \leq \frac{2C_m}{N_m} + \sqrt{\frac{\ln \frac{4}{\beta}}{\bar{n}_m(a)}} \right) \geq 1 - \beta/4$$

And in a similar argument $-B_m(a)/n_m(a)$ satisfies the same bound expect with probability $\beta/4$.
So then:

$$\left| \frac{1}{\bar{n}_m(a)} B_m(a) \right| \leq 2 \frac{q_m(a)}{\bar{n}_m(a)} \sum_{t \in E_m} |c_t(a)| + \frac{\ln \frac{4}{\beta}}{\bar{n}_m(a)}$$

By a simple union bound and removing the conditioning on the arbitrary values (as they hold for any arbitrary value, they also hold for the RV representing that value) we have the claimed. □

We would like a similar deviation bound to hold for all the arms and all epochs, using union bound. But first we need to calculate some bounds for the number of epochs to do that.

Lemma 4.4.2. *The length N_m of each epoch m satisfies:*

$$\lambda 2^{2(m-1)} \leq N_m \leq k \lambda 2^{2(m-1)}$$

and the number M of epochs is at most $\log_2 T$

Proof. The lower bound is an easy consequence of the existence of a "best" arm in an epoch and the algorithm assigning $\Delta_m = 2^{-m}$ in that case. On the other hand, since Δ_m is at least 2^{-m} for any arm, we have the upper bound. The bound on the number of epoch's is a consequence of the lower bound on the length of each epoch. □

By a union bound on the $\log T$ epochs and k arms then, and a simple Hoeffding bound on the actual number of plays $n_m(a)$ we have the following important lemma.

Lemma 4.4.3. *Define*

$$\mathcal{E} := \left\{ \forall m, i : |\mu_m(a) - \mu(a)| \leq \frac{2C_m}{N_m} + \frac{\Delta_{m-1}(a)}{16} \text{ and } n_m(a) \leq 2\bar{n}_m(a) \right\}$$

. It holds that $\mathbb{P}(\mathcal{E}) \geq 1 - \delta$

Proof. Choose $\beta = \frac{\delta}{2k \log T}$, we have then that $\bar{n}_m(a) = \frac{\lambda}{\Delta^2_{m-1}(a)} = 4 \cdot 16^2 \ln \frac{4}{\beta} \cdot \frac{1}{\Delta^2_{m-1}(a)}$ and by a union bound on the arms, epochs and the two inequalities we have the claimed bound. □

Gupta et. al go on to prove that the empirical gaps Δ_m are not too far off the true gaps and get closer as the epochs move on, if not for the corruption. They prove the following:

Lemma 4.4.4.

$$\Delta_m(a) \leq 2(\Delta(a) + 2^{-m} + \rho_m)$$

and

$$\Delta_m(a) \geq \frac{1}{2}\Delta(a) - \frac{3}{4}2^{-m} - 3\rho_m$$

where $\rho_m := \sum_{e=1}^{m} \frac{2C_e}{8^{m-e}N_e}$

We can think of the value ρ_m as a kind of 'discounted corruption rate' as the authors name it. We notice that the empirical gap is contaminated not only by the current epoch's corruption but previous one's as well. However, corruption from earlier epochs is less relevant as time goes on, as the quantity suggests.

We provide a sketch of the proof of the algorithm's regret bound. The authors note that:

$$R(T) = \sum_{m\in[M]} \sum_{a\in[k]} \Delta(a)n_m(a) \leq 2 \sum_{m\in[M]} \sum_{a\in[k]} \Delta(a)\bar{n}_m(a)$$

Then they consider three cases for the true gap and the corruption rate.

First, consider the case that the true gap is very small. In that case the algorithm cannot hope to have a good estimation of the gap, but surely it could not have pulled considerably more than it "deserved" too, since we have a lower bound on the empirical gap and as such an upper bound on the expected number of plays. Assuming that $\Delta(a) < 4 \cdot 2^{-m}$, then from the algorithm's upper bound $\bar{n}_m(a) \leq \lambda 2^{2(m-1)} \leq \frac{\lambda}{\Delta^2(a)}$ and

$$\bar{n}_m(a)\Delta(a) \leq \frac{4\lambda}{\Delta(a)}$$

Note that this holds whatever the corruption rate!

Now consider a case where the gap is considerable and there is a small corruption rate. Specifically when $\Delta(a) > 4{\cdot}2^{-m}$ and $\rho_{m-1} < \frac{\Delta(a)}{32}$. By Lemma 4.4.4 we have that $\Delta_{m-1}(a) \geq \frac{1}{32}\Delta(a)$ which implies that $\bar{n}_m(a)\Delta(a) \leq \frac{32^2\lambda}{\Delta(a)}$.

The last case to consider is when the gap is considerable but there is also considerable corruption. Specifically when $\rho_{m-1} > \frac{\Delta(a)}{32} \rightarrow \Delta(a) < 32\rho_{m-1}$. In this case we charge the regret to the corruption $\bar{n}_m(a)\Delta(a) \leq 8\lambda\rho_{m-1}2^{2m}$. By carefully summing over the epochs and arms the claimed regret bound is proven.

Chapter 5

Variance Estimates in Corrupted Bandits

5.1 Intro

Adversarially corrupted bandits were first studied by Lykouris et al. in [46] with their seminal paper. They provided an algorithm agnostic to the corruption level C and robust to corruption up to $o(\sqrt{T})$. The algorithm's central idea is to enhance its confidence bounds by the uncertainty that the corruption brings forth. This is easy enough when the corruption level is known; in the general case one can run multiple instances of UCB with enhanced confidence bounds implementing something similar to a binary search on the corruption, as Lykouris et al. showed.

The work of Gupta et al. in [36] improved upon this by devising an algorithm that is robust to corruption level $C \leq o(T)$. This algorithm works in epochs, where each one makes decisions based on the previous one's estimations. Given the estimated gap of an arm by a previous epoch, one can mimick the behaviour of UCB and similar algorithms, by playing an arm according to that gap. A crucial point is that the epochs keep increasing in length (roughly doubling each time) and as such the adversary has to continue injecting exponentially more corruption (and in a continuous manner) to strongly impact the learner.

Based on this work, we construct an algorithm that mimics UCB-V behaviour and assuming a fully stochastic scenario, essentially regains the regret bound that UCB-V satisfies (seen below):

Lemma 5.1.1. *[3] The expected regret attained by UCB-V is:*

$$\mathbb{E}\left[R(T)\right] = O\left(\ln T \cdot \sum_{a \neq a^*} \left[\frac{\sigma_a^2}{\Delta(a)} + 1\right]\right)$$

This lays promising ground for future work in extending these results to a corrupted case, and attaining similar bounds to [36]. However, as we will see, this algorithm suffers from a key property that makes it differ from *BARBAR*; corruption can alter an epoch's length in a significant matter. The BARBAR algorithm works in epochs, with exponentially increasing length. Each epoch utilizes the previous epoch's estimates of the arms' gaps, to influence its decisions. This ensures that no matter how corruption is shared throughout the rounds it has a "contained"/bounded effect. At epoch m, an arm is played with probability such, that in expectation, it is pulled roughly $\Theta\left(\frac{\ln T}{\Delta^2_{m-1}(a)}\right)$ times, as many as the upper bound of pulls UCB makes for an arm a with gap $\Delta_{m-1}(a)$. We will modify BARBAR to behave like UCB-V (in the same sense). Our algorithm follows.

Algorithm 10 BARBAR-V

```
$\lambda = f(k, T, ...)$                                  ▷ TBD
$T_0 = 1$
for $a \in [k]$ do
    $\Delta_0(a) = 1$
    $V_0(a) = \frac{1}{4}$                                ▷ Max variance of a RV w. support [0, 1] is 0.25.
for $m \in \{1, 2, ...\}$ do
    $\bar{n}_m(a) = \lambda\left(\frac{V_{m-1}(a)}{\Delta^2_{m-1}(a)} + \frac{1}{\Delta_{m-1}(a)}\right)$    ▷ An arm $a$ will be played roughly this many times.
    $N_m = \sum_{a\in[k]} \bar{n}_m(a)$
    $q_m(a) = \frac{\bar{n}_m(a)}{N_m}$
    $T_m = T_{m-1} + N_m$
    $E_m = [T_{m-1}, T_m]$
    for $t \in E_m \cap [T]$ do
        play arm $a$ w.p. $q_m(a)$
    $\mu_m(a) =$ empirical mean reward of $a$ at epoch $m$
    $V_m(a) =$ empirical variance of $a$'s reward at epoch $m$
    $a^*_m = \text{argmax}_{a\in[k]} \{\mu_m(a) - corr_m(a)\}$ ▷ We use a pessimistic estimation for the best arm,
    for reasons to be explained later, $corr_m(a)$ TBD
    $\mu^*_m = \mu_m(a^*_m) - corr_m(a^*_m)$
    $\Delta_m(a) = \max\{2^{-m}, \mu^*_m - \mu_m(a)\}$
```

Before getting into the analysis it would be wise to justify intuitively our choices. Simplifying the setting, we consider a purely stochastic scenario at first. By the law of large numbers we can convince ourselves that Δ_m will be approaching Δ and V_m will be approaching σ^2. Let's first see how many times are sub-optimal meta-arms played. If such an arm's variance is sufficiently small such that the second term dominates, then this arm will be played roughly $O(\frac{\lambda}{\Delta})$ times which is small (less than the the original BARBAR or UCB). However, as we will see this will be enough to "learn" the arm, as it will behave fairly statically due to the low variance. On the other hand, if the variance is big, then one cannot avoid playing roughly $O(\frac{\sigma^2}{\Delta})$, which is at most what UCB/BARBAR would do (but considerably less when the variance is asymptotically smaller than the gap of the arm). One maybe undesired property of this algorithm is that it tends to prefer "riskier" good arms (with higher variance) than more "stable" ones.

5.2 Preliminary Analysis

We begin by getting a handle on the length N_m of each epoch m and the number of epochs M, by the below lemma.
First of all by an easy application of Chernoff-Hoeffding (see Theorem A.0.2) we have that:

Lemma 5.2.1. *For all arms a and all epochs m the event*

$$\mathcal{E}_0 := \{|n_m(a) - \bar{n}_m(a)| \leq \bar{n}_m(a)/2\}$$

holds with probability at least $1 - 2k \log T e^{-\lambda/2}$

Proof. Apply the cited inequality and note that $\bar{n}_m(a) \geq \lambda$ and that the number of epochs is at most $\log T$ as in [36]. □

We can decompose arm a's reward $r_t(a)$ as $r_t(a) = \tilde{r}_t(a) + c_t(a)$, where $\tilde{r}_t$ is the stochastic reward and c_t the injected corruption.
The next result follows directly from the Freedman-type inequality we proved in chapter 2:

Lemma 5.2.2. *We have that for any arm and epoch*

$$\left| \frac{1}{n_m(a)} \sum_{t \in E_m} \mathbb{I}\{a_t = a\} \tilde{r}_t(a) - \mu(a) \right| \leq 2\sqrt{\frac{2x\sigma_a^2}{\bar{n}_m(a)}} + \frac{8}{3} \frac{x}{\bar{n}_m(a)}$$

or equivalently:

$$\left| \frac{1}{n_m(a)} \sum_{t \in E_m} \mathbb{I}\{a_t = a\} \tilde{r}_t(a) - \mu(a) \right| \leq 2\sqrt{2} \cdot \sqrt{\frac{x}{\lambda}} \cdot \sqrt{\frac{\sigma_a^2}{V_{m-1}(a)}} \cdot \Delta_{m-1}(a) + \frac{8}{3} \frac{x}{\lambda} \Delta_{m-1}(a)$$

with probability at least $1 - 2k \log^2 T e^{-x}$, assuming that $\mathcal{E}_0$ holds.

Proof. First of all, we condition on the epoch, which means that $\bar{n}_m(a), N_m(a), n_m(a)$ are deterministic quantities. Verify that from the concentration inequality of Lemma 2.4.4, the following hold for fixed a and epoch m with probability at least $1 - 2 \log T e^{-x}$:

$$\left| \frac{1}{n_m(a)} \sum_{t \in E_m} \mathbb{I}\{a_t = a\} r_t(a) - \mu(a) \right| = \left| \frac{1}{n_m(a)} \sum_{i=1}^{n_m(a)} X_i(a; m) - \mu(a) \right| \leq 2\sqrt{\frac{x\sigma_a^2}{n_m(a)}} + \frac{4}{3} \frac{x}{n_m(a)}$$

We define $X_i(a; m)$ as the i-th pull of arm a at epoch m. Event $\mathcal{E}_0$ from Corollary A.0.1 was defined such that $|n_m(a) - \bar{n}_m(a)| \leq \bar{n}_m(a)/2$ and for the first addend on the RHS use the fact that $n_m(a) \geq \frac{\lambda V_{m-1}(a)}{\Delta_{m-1}^2(a)}$, while for the second use that $n_m(a) \geq \frac{\lambda}{\Delta_{m-1}(a)}$.

We lift the conditioning on the epoch's quantities, as the above hold for any realization of those values. □

If we can manage to prove that $V_{m-1}(a)$ is a good approximation to the true variance (in particular that it is at least a constant fraction of it), then choosing λ appropriately our estimation of the reward of arm a, that is $r_m(a)$, will be very close to the true expected reward. In that case, we will have that the deviation of the empirical reward from the true one, will be within a constant factor of the previous epoch's empirical gap. As this empirical gap will tend to the true one (assuming little corruption), our algorithm's regret follows classical optimal regret bounds in the stochastic case. Simply, in a more practical sense, we can follow similar arguments as [36] to bound the algorithm's regret and reach very similar results.

Unfortunately, this will not happen in general. Let's investigate how close we can get to the actual variance. First of all, we need to define $V_m(a)$ explicitly.

5.3 Bounding the empirical variance

Consider the classical estimation of the variance of an RV:

$$V = \frac{1}{n-1}\sum_{i=1}^{n} U_i, \text{ where } U_i := \left(X_i - \frac{1}{n}\sum_{j=1}^{n} X_j\right)^2$$

For a second, let's assume that the rewards are stochastic. This estimation is not desirable for the following reason: U_i are not independent and they neither form a martingale sequence (or can be manipulated into one). We would want at least one of those properties to be able to achieve strong concentration guarantees of $V_m(a)$ around its expectation, σ^2.

We define the following estimator:

$$V_m(a) := \frac{1}{\lfloor n_m(a)/2 \rfloor}\sum_{i=1}^{\lfloor n_m(a)/2 \rfloor} U_i, \text{ where } U_i := (X_{2i-1} - X_{2i})^2$$

Note: we simplify notation, by considering X_i as the i-th pull of arm a at epoch m.

We had proven in Chapter 2 the following properties:

$$\mathbb{E}\left[U_i\right] = \sigma^2 \text{ and } \mathbb{V}ar(U_i) \leq \sigma^2$$

Lemma 5.3.1. *Concerning the "stochastic variance", ie*

$$\tilde{V}_m(a) := \frac{1}{\lfloor n_m(a)/2 \rfloor}\sum_{i=1}^{\lfloor n_m(a)/2 \rfloor} (\tilde{r}_{2i-1}(a) - \tilde{r}_{2i}(a))^2$$

the following concentration inequality holds, for all arms and epochs with probability at least $1 - 2k\log^2 T e^{-x}$, assuming $\mathcal{E}_0$.

$$\left|\tilde{V}_m(a) - \sigma_a^2\right| \leq 2\sqrt{6\sigma_a^2 \cdot \frac{x}{\bar{n}_m(a)}} + \frac{8x}{\bar{n}_m(a)}$$

Proof. From Chapter 2 and specifically in Lemma 2.4.6 we have that, for a fixed arm a, with probability at least $1 - 2\log T e^{-x}$:

$$\left|\tilde{V}_m(a) - \sigma_a^2\right| \leq 2\sqrt{3\sigma_a^2 \cdot x \cdot \frac{1}{n_m(a)}} + \frac{4x}{n_m(a)}$$

Now use that $n_m(a) \geq 1/2\bar{n}_m(a)$ from event $\mathcal{E}_0$ and union bound over the number of epochs and arms. □

Lemma 5.3.2. *Assuming $\mathcal{E}_0$ holds, for all arms a, epochs m we have with probability at least $1 - k\log Te^{-x}$:*

$$\tilde{V}_m(a) - \frac{48C_m}{N_m} - \frac{24x}{\bar{n}_m(a)} \leq V_m(a) \leq 2\tilde{V}_m(a) + \frac{96C_m}{N_m} + \frac{48x}{\bar{n}_m(a)}$$

Where C_m is defined as in [36], ie $C_m := \sum_{t\in E_m} \max_a |c_t(a)|$

Proof. For the upper bound we have that (abusing notation):

$$\begin{aligned}
\left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot V_m(a) &= \sum_{i=1}^{n_m(a)/2} (X_{2i-1} - X_{-i})^2 \\
&= \sum_{i=1}^{n_m(a)/2} (r_{2i-1}(a) - r_{2i}(a) + c_{2i-1}(a) - c_{2i}(a))^2 \\
&\leq 2 \sum_{i=1}^{n_m(a)/2} \left\{ (r_{2i-1}(a) - r_{2i}(a))^2 + (c_{2i-1}(a) - c_{2i}(a))^2 \right\} \\
&\leq 2 \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) + 2 \sum_{t\in E_m} \mathbb{I}\{a_t = a\} \cdot (2|c_t(a)|)^2 \\
&\leq 2 \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) + 8 \sum_{t\in E_m} \mathbb{I}\{a_t = a\} \cdot |c_t(a)| \qquad \text{as } |c_t(\cdot)| \leq 1 \\
&\leq 2 \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) + 32C_m \cdot \frac{\bar{n}_m(a)}{N_m} + 16x
\end{aligned}$$

For the lower bound we have that:

$$\begin{aligned}
\left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot V_m(a) &= \sum_{i=1}^{n_m(a)/2} (X_{2i-1} - X_{2i})^2 \\
&\geq \sum_{i=1}^{n_m(a)/2} (r_{2i-1}(a) - r_{2i}(a))^2 - 2 \sum_{i=1}^{n_m(a)/2} |r_{2i-1}(a) - r_{2i}(a)| \cdot |c_{2i-1}(a) - c_{2i}(a)| \\
&\geq \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) - 2 \sum_{i=1}^{n_m(a)} |c_{2i-1}(a) - c_{2i}(a)| \text{ since } r_t(a) \in [0,1] \\
&\geq \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) - 4 \sum_{t\in E_m} \mathbb{I}\{a_t = a\} \cdot |c_t(a)| \\
&\geq \left\lfloor \frac{n_m(a)}{2} \right\rfloor \cdot \tilde{V}_m(a) - 16C_m \cdot \frac{\bar{n}_m(a)}{N_m} - 8x
\end{aligned}$$

In both cases use $\lfloor n/2 \rfloor \geq n/3$ for $n \geq 2$ (since for $n < 2$ we don't have a variance estimation anyways). For the last step the reader can look into the next lemma. □

Lemma 5.3.3. *With probability at least $1 - k\log Te^{-x}$ for all arms and epochs:*

$$\frac{1}{n_m(a)} \sum_{t\in E_m} \mathbb{I}\{a_t = a\}|c_t(a)| \leq 4C_m/N_m + 2x/\bar{n}_m(a)$$

assuming that $n_m(a) \geq 1/2\bar{n}_m(a)$

Proof. As in BARBAR's proof, use Beygelzimer's Freedman-like concentration inequality on the MDS:

$$X_t = (\mathbb{I}\{a_t = a\} - q_m(a))|c_t(a)|$$

It is of course easy to see that this is an MDS:

$$\mathbb{E}\left[X_t \mid \mathcal{H}_{t-1}\right] = \mathbb{E}\left[\mathbb{I}\{a_t = a\} - q_m(a) \mid \mathcal{H}_{t-1}\right] \cdot \mathbb{E}\left[|c_t(a)| \mid \mathcal{H}_{t-1}\right] = 0$$

Define $X = \sum_{t \in E_m} X_t$ and $\mathcal{V} = \sum_{t \in E_m} \mathbb{E}\left[X_t^2 \mid \mathcal{H}_{t-1}\right]$
Then assuming a deterministic adversary:

$$\mathcal{V} \leq \sum_{t \in E_m} |c_t(a)| \mathbb{V}ar\left(\mathbb{I}\{a_t = a\}\right) \leq q_m(a) \sum_{t \in E_m} |c_t(a)|$$

By Theorem A.0.6 we have that:

$$X = \sum_{t \in E_m} \mathbb{I}\{a_t = a\}|c_t(a)| - q_m(a) \sum_{t \in E_m} |c_t(a)| \leq \mathcal{V} + x$$

Using that $\mathcal{V} \leq q_m(a) \sum_{t \in E_m} |c_t(a)| = \frac{\bar{n}_m(a)}{N_m} C_m$, we have that:

$$\sum_{t \in E_m} \mathbb{I}\{a_t = a\}|c_t(a)| \leq 2 \frac{\bar{n}_m(a)}{N_m} C_m + x$$

Dividing by $n_m(a)$ and using the assumption that $n_m(a) \geq 1/2\bar{n}_m(a)$ we have the result. □

We can now reason about how close is the empirical variance $V_m(a)$ to the actual variance and prove the following lemma.

Lemma 5.3.4. *For all arms a and epochs and with probability at least $1 - 3k \log^2 T e^{-x}$ we have that:*
When $\sigma_a^2 \geq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$:

$$-\frac{48 C_m}{N_m} + \frac{1}{4}\sigma_a^2 \leq V_m(a) \leq 4\sigma_a^2 + \frac{96 C_m}{N_m}$$

And when $\sigma_a^2 \leq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$:

$$V_m(a) \leq \frac{295x}{\lambda} \cdot \Delta_{m-1}(a) + \frac{96 C_m}{N_m}$$

Proof. From the above lemmas by a simple union bound we have that for all arms and epochs, with probability at least $1 - 3k \log^2 T e^{-x}$:

$$V_m(a) \geq \sigma_a^2 - 2\sqrt{\sigma_a^2 \frac{6x}{\bar{n}_m(a)}} - \frac{32x}{\bar{n}_m(a)} - \frac{48 C_m}{N_m}$$

And simultaneously:

$$V_m(a) \leq 2\sigma_a^2 + 4\sqrt{\sigma_a^2 \frac{6x}{\bar{n}_m(a)}} + \frac{56x}{\bar{n}_m(a)} + \frac{96 C_m}{N_m}$$

Now note that $\bar{n}_m(a) = \lambda \left(\frac{V_{m-1}(a)}{\Delta_{m-1}^2(a)} + \frac{1}{\Delta_{m-1}(a)} \right) \geq \frac{\lambda}{\Delta_{m-1}}(a)$ so that: $\bar{n}_m(a)^{-1} \leq 1/\lambda \cdot \Delta_{m-1}(a)$
We note the following cases.

Case 1/ High variance: $\sigma_a^2 \geq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$

We have that:

$$\begin{aligned}
V_m(a) &\geq \sigma_a^2 - 2\sqrt{6}\sqrt{\sigma_a^2 \frac{x}{\bar{n}_m(a)}} - \frac{32x}{\bar{n}_m(a)} - \frac{48C_m}{N_m} \\
&\geq \sigma_a^2 - 2\sqrt{6}\sqrt{\sigma_a^2 \frac{x}{\lambda} \cdot \Delta_{m-1}(a)} - \frac{16x}{\lambda} \cdot \Delta_{m-1}(a) - \frac{48C_m}{N_m} \\
&\geq \sigma_a^2 - 2\sqrt{6}\sqrt{\sigma_a^2 \frac{1}{128}\sigma_a^2} - 16\frac{\sigma_a^2}{128} - \frac{48C_m}{N_m} \\
&\geq \frac{1}{4}\sigma_a^2 - \frac{48C_m}{N_m}
\end{aligned}$$

And similarly:

$$V_m(a) \leq 4\sigma_a^2 + \frac{32C_m}{N_m}$$

Case 2/ Small variance: $\sigma^2 \leq \frac{128x}{\lambda} \cdot \Delta_{m-1}(a)$

In this case we don't have a good lower bound for $V_m(a)$, but we don't need one.

For the upper bound, which we will need to bound the number of plays of arm a:

$$\begin{aligned}
\tilde{V}_m(a) &\leq 2\sigma^2 + 4\sqrt{6} \cdot \sqrt{\sigma_a^2 \cdot \frac{x}{\lambda}\Delta_{m-1}(a)} + \frac{56x}{\lambda} \cdot \Delta_{m-1}(a) + \frac{96C_m}{N_m} \\
&\leq \frac{295x}{\lambda} \cdot \Delta_{m-1}(a) + \frac{96C_m}{N_m}
\end{aligned}$$

□

5.4 Guaranteeing a good estimation of the rewards

Lemma 5.4.1. *With high probability, for appropriate λ, we have that for all arms a and all epochs m:*

$$|\mu_m(a) - \mu_a| \leq \frac{1}{16}\Delta_{m-1}(a) + \frac{4C_m}{N_m} + \frac{1}{128}\Delta_{m-2}(a) + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}}$$

(Define $\Delta_i(a) = 1, C_i = 0$ for $i \leq 0$ for conciseness of statements).

Proof. Using x such that $x = \ln \frac{10k^2 \log T}{\delta}$ and choosing $\lambda = 2^{12}x$ all the relevant events (Lemma 5.2.2, Lemma 5.3.2 etc) hold with probability at least $1 - \delta$ simultaneously, for all arms and epochs (via union bound). All the following approximations in the analysis are justified as well, as one could verify.

We begin by following the analysis of [36] and breaking the deviation by the triangle inequality as:

$$|\mu_m(a) - \mu_a| \leq |\tilde{\mu}_m(a) - \mu_a| + \frac{1}{n_m(a)} \sum_{t \in E_m} \mathbb{I}(a_t = a)|c_t(a)|$$

The corruption is bounded in a similar fashion through Lemma 5.3.3. So the rest of the analysis focuses on bounding the stochastic sample mean deviation.

Note that the case of $m = 1$ is simple, since $V_0(a) = 1/4$ and $\Delta_0(a) = 1$ and we have that:

$$\bar{n}_m(a) = \frac{\lambda V_0(a)}{\Delta_0^2(a)} + \frac{\lambda}{\Delta_0(a)} = \frac{5\lambda}{4\Delta_0^2(a)}$$

Which means that the result follows from a simple application of a concentration inequality such as Hoeffding's.

Next, we focus on any epoch beyond the first.

Case 1/ Small variance: $\sigma_a^2 \leq 128\frac{x}{\lambda}\Delta_{m-2}(a)$

From Lemma 5.2.2 we have that:

$$\begin{aligned}
|\tilde{\mu}_m(a) - \mu_a| &\leq 2\sqrt{2} \cdot \sqrt{\frac{x\sigma_a^2}{\bar{n}_m(a)}} + \frac{8}{3}\frac{x}{\bar{n}_m(a)} \\
&\leq 2\sqrt{2} \cdot \sqrt{\frac{x \cdot \left(128\frac{x}{\lambda}\Delta_{m-2}(a)\right)}{\bar{n}_m(a)}} + \frac{8}{3}\frac{x}{\bar{n}_m(a)} && \text{by the assumption} \\
&\leq 32\sqrt{\frac{x}{\lambda}\Delta_{m-2}(a) \cdot \frac{x}{\bar{n}_m(a)}} + \frac{8x}{\bar{n}_m(a)} \\
&\leq 32\frac{x}{\lambda}\sqrt{\Delta_{m-2}(a) \cdot \Delta_{m-1}(a)} + 0.68\frac{x}{\lambda}\Delta_{m-1}(a) \\
&\leq \frac{x}{\lambda}\left(32\Delta_{m-2}(a) + 32\Delta_{m-1}(a) + 0.68\Delta_{m-1}(a)\right) && \text{since } xy \leq \frac{1}{2}(x^2 + y^2) \\
&\leq \frac{1}{128}\Delta_{m-2}(a) + \frac{1}{128}\Delta_{m-1}(a) && \text{since } \lambda \geq 2^{12}x
\end{aligned}$$

Case 2/ High variance: $\sigma_a^2 \geq 128\frac{x}{\lambda}\Delta_{m-2}(a)$

a) Small corruption $\frac{C_{m-1}}{N_{m-1}} \leq \frac{1}{192}\sigma_a^2$

In this case from Lemma 5.3.4 we have that:

$$V_m(a) \geq \frac{1}{8}\sigma_a^2$$

So then:

$$\begin{aligned}
|\tilde{\mu}_m(a) - \mu_a| &\leq 2\sqrt{2}\sqrt{\frac{x}{\lambda}} \cdot \sqrt{\frac{\sigma_a^2}{V_{m-1}(a)}} \cdot \Delta_{m-1}(a) + \frac{8}{3}\frac{x}{\lambda}\Delta_{m-1}(a) \\
&\leq 2\sqrt{2}\sqrt{\frac{x}{\lambda}} \cdot \sqrt{\frac{\sigma_a^2}{\frac{1}{8}\sigma_a^2}} \cdot \Delta_{m-1}(a) + \frac{8}{3}\frac{x}{\lambda}\Delta_{m-1}(a) && \text{using Lemma 5.3.4} \\
&\leq \frac{1}{16}\Delta_{m-1}(a) && \text{since } \lambda \geq 2^{12}x
\end{aligned}$$

b) High corruption $\frac{C_{m-1}}{N_{m-1}} \geq \frac{1}{192}\sigma_a^2$

From Lemma 5.2.2 we have that:

$$\begin{aligned}
|\tilde{\mu}_m(a) - \mu_a| &\leq 2\sqrt{2}\sqrt{\frac{x\sigma_a^2}{\bar{n}_m(a)}} + \frac{8}{3}\frac{x}{\bar{n}_m(a)} \\
&\leq 2\sqrt{2}\sqrt{\frac{x}{\lambda}\sigma_a^2\Delta_{m-1}(a)} + \frac{8x}{3\lambda}\Delta_{m-1}(a) \\
&\leq 2\sqrt{2}\sqrt{\frac{x}{\lambda}}\sqrt{\frac{1}{192}\frac{C_{m-1}}{N_{m-1}} \cdot \Delta_{m-1}(a)} + \frac{8x}{3\lambda}\Delta_{m-1}(a) \\
&\leq 2\sqrt{2}\sqrt{\frac{x}{\lambda}}\left(\frac{1}{96}\frac{C_{m-1}}{N_{m-1}} + \Delta_{m-1}(a)\right) + \frac{8x}{3\lambda}\Delta_{m-1}(a) && xy \leq \frac{1}{2}(x^2 + y^2) \\
&\leq \frac{1}{16}\Delta_{m-1}(a) + \frac{1}{16}\frac{C_{m-1}}{N_{m-1}} && \text{since } \lambda \geq 2^{12}x
\end{aligned}$$

□

5.5 Bounding estimated gaps

If we choose $corr_m(a) = \frac{1}{128}\Delta_{m-2}(a) + \frac{1}{16}\Delta_{m-1}(a)$, we can reach an almost identical result as Lemma 5 in [36], using the above lemma and following the same analysis.

We would like to bound $\Delta_m(a)$ from above to ensure a good estimation for $r_m(a)$, but also bound it from below so as to bound the number of plays $n_m(a)$ as well. In the following results, we will assume all the previous events hold.

Remark 5.5.1.

$$\begin{aligned}
\Delta_m(a) &= \mu_m^* - \mu_m(a) \\
&= \underbrace{\mu_m^* - \mu^*}_{\text{we would like it to be } \leq 0} + \underbrace{\mu^* - \mu_a}_{=\Delta(a)} + \underbrace{\mu_a - \mu_m(a)}_{\text{bounded by prev. lemmas}}
\end{aligned}$$

It is clear that to bound $\Delta_m(a)$ we need to bound how far away is the estimated best reward at epoch m from the real optimal reward. That is what we do in the following lemma.

Lemma 5.5.1. *Choosing* $corr_m(a) = \frac{1}{128}\Delta_{m-2}(a) + \frac{1}{16}\Delta_{m-1}(a)$ *it holds that:*

$$-\frac{4C_m}{N_m} - \frac{1}{2}\frac{C_{m-1}}{N_{m-1}} - \frac{1}{64}\Delta_{m-2}(a^*) - \frac{1}{8}\Delta_{m-1}(a^*) \leq \mu_m^* - \mu^* \leq \frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}}$$

Where with a^* we denote the optimal arm (remember that μ_m^* doesn't necessarily refer to the optimal arm).

Proof. The upper bound is trivial since:

$$\begin{aligned}\mu_m^* - \mu^* &= \mu_m(a^*) - corr_m(a^*) - \mu^* \\ &\overset{Lemma\ 5.4.1}{\leq} \mu_{a_m^*} + \frac{1}{128}\Delta_{m-2}(a^*) + \frac{1}{16}\Delta_{m-1}(a^*) - corr_m(a_m^*) - \mu^* + \frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}} \\ &= -\Delta(a_m^*) + \frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}}\end{aligned}$$

For the lower bound, we have by definition:

$$\mu_m^* = \max_a\{\mu_m(a) - corr_m(a)\} \geq \mu_m(a^*) - corr_m(a^*)$$

So then:

$$\begin{aligned}\mu_m^* - \mu^* &\geq \mu_m(a^*) - corr_m(a^*) - \mu^* \\ &\overset{Lemma\ 5.4.1}{\geq} \mu^* - \frac{1}{128}\Delta_{m-2}(a^*) + \frac{1}{16}\Delta_{m-1}(a^*) - \left(\frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}}\right) - corr_m(a^*) - \mu^* \\ &= -\frac{4C_m}{N_m} - \frac{1}{2}\frac{C_{m-1}}{N_{m-1}} - \frac{1}{64}\Delta_{m-2}(a^*) - \frac{1}{8}\Delta_{m-1}(a^*)\end{aligned}$$

□

Having bounds for $\Delta_m(a)$ independent of previous estimations is crucial to analyzing the algorithm's regret. First, we will provide an upper bound.

Lemma 5.5.2.

$$\Delta_m(a) \leq 2(\Delta(a) + 2^{-m} + \rho_m)$$

Where we define:

$$\rho_m := \sum_{s=1}^{m} \frac{1}{8^{m-s-1}}\frac{C_s}{N_s}$$

Proof. Since $2 \cdot 2^{-1} = 1$ the statement holds for $m \leq 1$ trivially. Assuming it holds for epochs $m' \leq m-1$ we will prove it also holds for any epoch $m \geq 2$. We have that:

$$\begin{aligned}\mu_m^* - \mu_m(a) &= (\mu_m^* - \mu^*) + (\mu^* - \mu(a)) + (\mu(a) - \mu_m(a)) \\ &\leq \left(\frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}}\right) + \Delta(a) + \left(\frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}} + \frac{1}{16}\Delta_{m-1}(a) + \frac{1}{128}\Delta_{m-2}(a)\right)\end{aligned}$$

$$\begin{aligned}
\mu_m^* - \mu_m(a) &\leq \Delta(a) + \frac{1}{128}\Delta_{m-2}(a) + \frac{1}{16}\Delta_{m-1}(a) + \frac{C_{m-1}}{N_{m-1}} + \frac{8C_m}{N_m} \\
&\leq \Delta(a) + \frac{2}{128}(\Delta(a) + 2^{-m+2} + \rho_{m-2}) + \frac{2}{16}(\Delta(a) + 2^{-m+1} + \rho_{m-1}) + \frac{C_{m-1}}{N_{m-1}} + \frac{8C_m}{N_m} \\
&\leq 2(\Delta(a) + 2^{-m}) + \frac{1}{8}\left(\frac{1}{8}\rho_{m-2} + \frac{8C_{m-1}}{N_{m-1}}\right) + \frac{1}{8}\rho_{m-1} + \frac{8C_m}{N_m} \\
&\leq 2(\Delta(a) + 2^{-m}) + \frac{1}{8}\rho_{m-1} + \rho_m \\
&\leq 2(\Delta(a) + 2^{-m}) + \rho_m + \rho_m
\end{aligned}$$

Where in the second line we used the inductive hypothesis. The proof is finished by noting that:

$$\Delta_m(a) = \max\left\{2^{-m}, \mu_m^* - \mu_m(a)\right\}$$

□

Lemma 5.5.3. *We also have that*

$$\Delta_m(a) \geq \frac{1}{2}\Delta(a) - 2^{-m} - 6\rho_m$$

Proof. First we note that:

$$\Delta_m(a) \geq \mu_m^* - \mu_m(a)$$

We can bound μ_m^* by Lemma 5.5.1:

$$\mu_m^* \geq \mu^* - \frac{4C_m}{N_m} - -\frac{1}{2}\frac{C_{m-1}}{N_{m-1}} - \frac{1}{8}\Delta_{m-1}(a^*) - \frac{1}{64}\Delta_{m-2}(a^*)$$

We also bound $\mu_m(a)$ below by the main concentration inequality we constructed, ie Lemma 5.4.1:

$$\mu_m(a) \geq \mu(a) + \frac{4C_m}{N_m} + \frac{1}{2}\frac{C_{m-1}}{N_{m-1}} + \frac{1}{16}\Delta_{m-1}(a) + \frac{1}{128}\Delta_{m-2}(a)$$

Which means that we have:

$$\begin{aligned}
\Delta_m(a) &\geq \Delta(a) - \frac{8C_m}{N_m} - \frac{C_{m-1}}{N_{m-1}} - \left(\frac{1}{8}\Delta_{m-1}(a^*) + \frac{1}{16}\Delta_{m-1}(a)\right) - \left(\frac{1}{64}\Delta_{m-2}(a^*) + \frac{1}{128}\Delta_{m-2}(a)\right) \\
&\geq \Delta(a) - \frac{8C_m}{N_m} - \frac{C_{m-1}}{N_{m-1}} - \left(\frac{3}{8}\rho_{m-1} + \frac{3}{8}2^{-(m-1)} + \frac{1}{8}\Delta(a)\right) - \frac{1}{8}\left(\frac{3}{8}\rho_{m-2} + \frac{3}{8}2^{-(m-2)} + \frac{1}{8}\Delta(a)\right) \\
&\geq \frac{1}{2}\Delta(a) - 2^{-m} - \left(\frac{8C_m}{N_m} + \frac{3}{8}\rho_{m-1}\right) - \frac{1}{8}\left(\frac{8C_{m-1}}{N_{m-1}} + \frac{3}{8}\rho_{m-2}\right) \\
&\geq \frac{1}{2}\Delta(a) - 2^{-m} - 3\rho_m - \frac{3}{8}\rho_{m-1} \\
&\geq \frac{1}{2}\Delta(a) - 2^{-m} - 3\rho_m - 3\rho_m
\end{aligned}$$

□

5.6 Regret

Theorem 5.6.1. *BARBAR-V with choices* $corr_m(a) = \frac{1}{128}\Delta_{m-2}(a)+\frac{1}{16}\Delta_{m-1}(a)$, $\lambda = 2^{12} \ln \frac{10k \log^2 T}{\delta}$ *attains the below regret bound with probability at least* $1-\delta$*, for the classic Stochastic MAB problem:*

$$R(T) \le O\left(\ln \frac{k \log^2 T}{\delta} \cdot \sum_{a \neq a^*} \log T \left\{\frac{\sigma_a^2}{\Delta(a)} + 1\right\}\right)$$

Proof. Let's first focus on the first epoch.

$$R_1(a) \le \frac{3}{2}\bar{n}_1(a)\Delta(a) = \frac{3}{2}\frac{5\lambda}{4}\Delta(a) = O(\lambda)$$

Next, we focus on epochs beyond the first.

Case 1/ High Variance: $\sigma_a^2 \ge 128 \cdot 2^{-12}\Delta_{m-2}(a)$

From Lemma 5.3.4 we have that $V_{m-1}(a) \le 4\sigma_a^2$ (since we assume a fully stochastic scenario) so that:

$$\begin{aligned}\bar{n}_m(a) &= \lambda\left(\frac{V_{m-1}(a)}{\Delta_{m-1}^2(a)} + \frac{1}{\Delta_{m-1}(a)}\right)\\ &\le O\left(\frac{\lambda\sigma_a^2}{\Delta_{m-1}^2(a)} + \frac{\lambda}{\Delta_{m-1}(a)}\right)\end{aligned}$$

Case 2/ Small variance $\sigma_a^2 \le 128 \cdot 2^{-12}\Delta_{m-2}(a)$

In that case, from Lemma 5.3.4 we have that $V_{m-1}(a) \le O\left(\Delta_{m-2}(a)\right)$

Which means that:

$$\begin{aligned}\bar{n}_m(a) &= \lambda\left(\frac{V_{m-1}(a)}{\Delta_{m-1}^2(a)} + \frac{1}{\Delta_{m-1}(a)}\right)\\ &\le O\left(\lambda\left[\frac{\Delta_{m-2}(a)}{\Delta_{m-1}(a)} + 1\right]\frac{1}{\Delta_{m-1}(a)}\right)\end{aligned}$$

Now we just need to show that $\Delta_{m-2}(a)$ and $\Delta_{m-1}(a)$ are close enough to each other, when corruption is small, such that $\frac{\Delta_{m-2}(a)}{\Delta_{m-1}(a)}$ is close to a constant. We note the following two cases.

a) $\Delta(a) \ge 8 \cdot 2^{-m}$

From Lemma 5.5.3 we have that $\Delta_{m-1}(a) \ge \frac{1}{4}\Delta(a)$ and from Lemma 5.5.2 we have $\Delta_{m-2}(a) \le 4\Delta(a)$, which means that $\frac{\Delta_{m-2}(a)}{\Delta_{m-1}(a)} = O(1)$.

In both the above cases:

$$R_m(a) = n_m(a) \cdot \Delta(a) \le O\left(\frac{\lambda\sigma_a^2}{\Delta^2(a)} + \frac{\lambda}{\Delta(a)}\right) \cdot \Delta(a)$$

b) $\Delta(a) \le 8 \cdot 2^{-m}$

In this case by the strict lower bound in the estimation of $\Delta(a)$, in particular by

$\Delta_m(a) \geq^{-m}$ and from Lemma 5.5.2 $\Delta_{m-2}(a) \leq 2(\Delta(a) + 2^{-m-2} + \rho_{m-1}/8) < 9 \cdot 2^{-m}$ we bound the number of plays in both cases:

$$n_m(a) \leq \frac{3}{2}\bar{n}_m(a) \leq O\left(\lambda(\sigma_a^2 \cdot 2^{2m} + 2^m)\right) \leq O\left(\frac{\lambda\sigma_a^2}{\Delta^2(a)} + \frac{\lambda}{\Delta(a)}\right)$$

Summing over all epochs and arms we get the result when no corruption is present. □

In contrast to the above positive result, we present the below negative result, suggesting that the algorithm in its current state cannot attain the regret bounds we wished.

Theorem 5.6.2. *A corruption of $\Theta(\sqrt{T})$ can make the learner suffer $\Omega(T)$ regret.*

Proof. Consider a two arm instance, where both arms have a high variance (constant) and the suboptimal arm has gap Δ. Denote by $M = \log_4(T)$. Now consider an adversary that corrupts epoch $M/2 - 2$ such that the learner thinks both arms are almost deterministic, in that case the next epoch will have length $\Theta(2^{M/2-1})$ (notice we do not care which arm is the 'best' at this epoch). The adversary continues to corrupt in the same manner all the forthcoming epochs up until epoch $M - 2$. The adversary then corrupts in the following way: they make the learner think that the optimal arm is the suboptimal one, while also making the learner think it has a high (constant) variance. The next epoch is epoch $M - 1$ and will have length $\Theta(4^{M-1})$.
The adversary had to use corruption:

$$O(4^{M/2}) + O\left(2^{M/2} + 2^{M/2+1} + \ldots + 2^{M-2}\right) = O(\sqrt{T})$$

But the learner suffers regret:

$$\Omega(\Delta \cdot 4^{M-1}) = \Omega(\Delta \cdot T)$$

Assuming that the instance is such that Δ is a constant, then the regret is linear. □

5.7 Future work

The most important question is that of the possibility of achieving a regret bound that deteriorates from the almost-optimal regret that *UCB-V* achieves, as one goes from a fully stochastic scenario to a corrupted one. Our work which recovers key lemmas with minor expected differences, and the fact that the algorithm does achieve the expected regret bound in the fully stochastic case, is a positive step in that direction. However, the key difference between our algorithm and the one in [36] is that epochs can very well not scale exponentially as the time goes on, when the adversary can corrupt the variances heavily. This imposes a challenge on analyzing the effect that corruption on previous epochs has on the next ones and also suggests that the algorithm cannot hope to recover these bounds, as our previous negative result showcases.

Appendix A

Concentration Inequalities

Theorem A.0.1. *Suppose $X_i \in [a_i, b_i]$ are pairwise independent, then Hoeffding's inequality [38] states:*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i\in[n]} X_i - E\left[\frac{1}{n}\sum_{i\in[n]} X_i\right]\right| \geq \delta\right) \leq 2exp\left\{-2\frac{n^2\delta^2}{\sum_{i\in[n]}(b_i-a_i)^2}\right\}$$

Theorem A.0.2. *[27] Suppose $\{X_i\}_{i=1}^n$ is a sequence of independent $[0,1]$ RVs and let $X = \sum_{i=1}^n X_i$. For any $\epsilon > 0$:*

$$\mathbb{P}\left(|X - \mathbb{E}\left[X\right]| \geq \epsilon\mathbb{E}\left[X\right]\right) \leq 2\exp\left\{-\frac{\epsilon^2}{3}\mathbb{E}\left[X\right]\right\}$$

Equivalently:

$$\mathbb{P}\left(|X - \mathbb{E}\left[X\right]| \leq \sqrt{3\mathbb{E}\left[X\right]\ln\frac{2}{\delta}}\right) \geq 1-\delta$$

Theorem A.0.3 (Bernstein's Inequality [9]). *A well known result is the celebrated Bernstein's concentration inequality. For X_i i.i.d with $|X_i| \leq M$, $\mathbb{E}\left[X_i\right] = \mu$ and $\mathbb{V}ar(X_i) = \sigma^2$, the following holds:*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^n X_i - \mu\right| \leq \sqrt{2\sigma^2 x/n} + \frac{2xM}{3n}\right) \geq 1 - 2e^{-x}$$

Definition A.0.1. *A martingale difference sequence (MDS) is a sequence of RVs Y_i such that $|Y_i - Y_{i-1}|$ is bounded and $\mathbb{E}\left[Y_i \mid Y_1, \ldots Y_{i-1}\right] = 0$.*

Theorem A.0.4 (Azuma-Hoeffding inequality [8]). *Suppose Y_i is an MDS and $Y_i \in [a_i, b_i]$ it almost surely,* Azuma-Hoeffding *inequality then states that:*

$$\mathbb{P}\left(\left|\sum_{i=1}^t Y_i\right| \geq \epsilon\right) \leq 2\exp\left\{-\frac{2\epsilon^2}{\sum_{\tau=1}^t (b_i - a_i)^2}\right\}$$

Theorem A.0.5 (Freedman's inequality *(implicit in [27])*, original form in [31]). *Let X_i be an arbitrary RV sequence and $f = f(X_1, \ldots, X_n)$. Define:*

$$D_i := \mathbb{E}\left[f \mid X_1, \ldots, X_i\right] - \mathbb{E}\left[f \mid X_1, \ldots, X_{i-1}\right], \text{ with } D_0 = 0$$

and

$$b = \max_i \{\sup\{|D_i - D_{i-1}| \mid X_1, \ldots, X_{i-1}\}\}$$

and also

$$V := \sum_{i=1}^{n} \sup_{X_{i-1}} \mathbb{V}ar\left(D_i \mid X_1, \ldots, X_{i-1}\right)$$

Then:

$$\mathbb{P}\left(|f - \mathbb{E}\left[f\right]| \geq t\right) \leq 2\exp\left\{-\frac{t^2}{2V + 2bt/3}\right\}$$

Corollary A.0.1. *Let Z_i be an MDS with $|Z_i - Z_{i-1}| \leq b$ almost surely, then we have that:*

$$\mathbb{P}\left(\left|\sum_{i=1}^{n} Z_i\right| \geq \sqrt{\frac{2}{b}\mathcal{V}\log\frac{2}{\delta}} + \frac{4b\log\frac{2}{\delta}}{3}\right) \leq \delta$$

where $\mathcal{V} := \sum_{i=1}^{n} \mathbb{E}\left[Z_i^2 \mid \mathcal{H}_{i-1}\right]$

Theorem A.0.6 (Beygelzimer et al. 2011 in [10])). *Suppose that $\{X_i\}_{i=1}^{n}$ is an MDS and let $X = \sum_{i\in[n]} X_i$. Assuming that $|X_i| \leq b$, then if we define $\mathcal{V} := \sum_{i\in[n]} \mathbb{E}\left[X_i^2 \mid \mathcal{H}_{i-1}\right]$, for any $\delta > 0$:*

$$\mathbb{P}\left(|X| \leq \frac{\mathcal{V}}{b} + b\ln\left(2/\delta\right)\right)$$

Bibliography

[1] F. J. Anscombe. "Sequential Medical Trials." In: *Journal of the American Statistical Association* 58.302 (1963), pp. 365–383. ISSN: 01621459, 1537274X. URL: http://www.jstor.org/stable/2283272 (visited on 09/01/2024).

[2] Jean-Yves Audibert and Sébastien Bubeck. "Minimax Policies for Adversarial and Stochastic Bandits." In: *Annual Conference Computational Learning Theory*. 2009. URL: https://api.semanticscholar.org/CorpusID:216051277.

[3] Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. "Exploration–exploitation tradeoff using variance estimates in multi-armed bandits." In: *Theoretical Computer Science* 410.19 (2009). Algorithmic Learning Theory, pp. 1876–1902. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2009.01.016. URL: https://www.sciencedirect.com/science/article/pii/S030439750900067X.

[4] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. "Finite-time Analysis of the Multiarmed Bandit Problem." In: *Machine Learning* 47 (2002), pp. 235–256. URL: https://api.semanticscholar.org/CorpusID:207609497.

[5] Peter Auer and Chao-Kai Chiang. *An algorithm with nearly optimal pseudo-regret for both stochastic and adversarial bandits*. 2016. arXiv: 1605.08722 [cs.LG]. URL: https://arxiv.org/abs/1605.08722.

[6] Peter Auer et al. "The Nonstochastic Multiarmed Bandit Problem." In: *SIAM Journal on Computing* 32.1 (2002), pp. 48–77. DOI: 10.1137/S0097539701398375. eprint: https://doi.org/10.1137/S0097539701398375. URL: https://doi.org/10.1137/S0097539701398375.

[7] Pranjal Awasthi et al. "Congested Bandits: Optimal Routing via Short-term Resets." In: *ArXiv* abs/2301.09251 (2023). URL: https://api.semanticscholar.org/CorpusID:250340869.

[8] Kazuoki Azuma. "Weighted sums of certain dependent random variables." In: *Tohoku Mathematical Journal* 19.3 (1967), pp. 357–367. DOI: 10.2748/tmj/1178243286. URL: https://doi.org/10.2748/tmj/1178243286.

[9] S. N. Bernstein. *On a modification of Chebyshev's inequality and on the error in Laplace formula*. 1964.

[10] Alina Beygelzimer et al. *Contextual Bandit Algorithms with Supervised Learning Guarantees*. 2011. arXiv: 1002.4058 [cs.LG]. URL: https://arxiv.org/abs/1002.4058.

[11] Aditya Bhaskara et al. *Online Learning and Bandits with Queried Hints*. 2022. arXiv: 2211.02703 [cs.DS]. URL: https://arxiv.org/abs/2211.02703.

[12] Djallel Bouneffouf, Irina Rish, and Charu Aggarwal. "Survey on Applications of Multi-Armed and Contextual Bandits." In: *2020 IEEE Congress on Evolutionary Computation (CEC)*. Glasgow, United Kingdom: IEEE Press, 2020, pp. 1–8. DOI: 10.1109/CEC48606.2020.9185782. URL: https://doi.org/10.1109/CEC48606.2020.9185782.

[13] Djallel Bouneffouf, Irina Rish, and Guillermo A. Cecchi. "Bandit Models of Human Behavior: Reward Processing in Mental Disorders." In: *Artificial General Intelligence*. Ed. by Tom Everitt, Ben Goertzel, and Alexey Potapov. Cham: Springer International Publishing, 2017, pp. 237–248. ISBN: 978-3-319-63703-7.

[14] Djallel Bouneffouf et al. "Contextual Bandit for Active Learning: Active Thompson Sampling." In: *Neural Information Processing*. Ed. by Chu Kiong Loo et al. Cham: Springer International Publishing, 2014, pp. 405–412. ISBN: 978-3-319-12637-1.

[15] Mario Bravo, David Leslie, and Panayotis Mertikopoulos. "Bandit learning in concave N-person games." In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS'18. Montréal, Canada: Curran Associates Inc., 2018, pp. 5666–5676.

[16] Guy Bresler, Devavrat Shah, and Luis Filipe Voloch. "Collaborative Filtering with Low Regret." In: *SIGMETRICS Perform. Eval. Rev.* 44.1 (June 2016), pp. 207–220. ISSN: 0163-5999. DOI: 10.1145/2964791.2901469. URL: https://doi.org/10.1145/2964791.2901469.

[17] Sebastien Bubeck and Aleksandrs Slivkins. *The best of both worlds: stochastic and adversarial bandits*. 2012. arXiv: 1202.4473 [cs.LG]. URL: https://arxiv.org/abs/1202.4473.

[18] Sébastien Bubeck and Nicolò Cesa-Bianchi. *Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems*. 2012. arXiv: 1204.5721 [cs.LG]. URL: https://arxiv.org/abs/1204.5721.

[19] Swapna Buccapatnam, Atilla Eryilmaz, and Ness B. Shroff. "Multi-armed bandits in the presence of side observations in social networks." In: *52nd IEEE Conference on Decision and Control*. 2013, pp. 7309–7314. DOI: 10.1109/CDC.2013.6761049.

[20] Robert R Bush and Frederick Mosteller. "A stochastic model with applications to learning." In: *The Annals of Mathematical Statistics* (1953), pp. 559–585.

[21] Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. USA: Cambridge University Press, 2006. ISBN: 0521841089.

[22] Nicolo Cesa-Bianchi et al. "How to use expert advice." In: *J. ACM* (1997).

[23] Konstantina Christakopoulou and Arindam Banerjee. *Learning to Interact with Users: A Collaborative-Bandit Approach*. May 2018. DOI: 10.1137/1.9781611975321.69.

[24] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2006.

[25] Jessica Dai et al. *Can Probabilistic Feedback Drive User Impacts in Online Platforms?* 2024. arXiv: 2401.05304 [cs.LG]. URL: https://arxiv.org/abs/2401.05304.

[26] Kaize Ding, Jundong Li, and Huan Liu. "Interactive Anomaly Detection on Attributed Networks." In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. WSDM '19. Melbourne VIC, Australia: Association for Computing Machinery, 2019, pp. 357–365. ISBN: 9781450359405. DOI: 10.1145/3289600.3290964. URL: https://doi.org/10.1145/3289600.3290964.

[27] Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

[28] Audrey Durand et al. "Contextual Bandits for Adapting Treatment in a Mouse Model of de Novo Carcinogenesis." In: *Proceedings of the 3rd Machine Learning for Healthcare Conference*. Ed. by Finale Doshi-Velez et al. Vol. 85. Proceedings of Machine Learning Research. PMLR, 17–18 Aug 2018, pp. 67–82. URL: https://proceedings.mlr.press/v85/durand18a.html.

[29] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. "Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems." In: *J. Mach. Learn. Res.* 7 (2006), pp. 1079–1105. ISSN: 1532-4435.

[30] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. "PAC Bounds for Multi-armed Bandit and Markov Decision Processes." In: *Computational Learning Theory*. Ed. by Jyrki Kivinen and Robert H. Sloan. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 255–270. ISBN: 978-3-540-45435-9.

[31] David A. Freedman. "On Tail Probabilities for Martingales." In: *The Annals of Probability* 3.1 (1975), pp. 100–118. ISSN: 00911798. URL: http://www.jstor.org/stable/2959268 (visited on 09/09/2024).

[32] Yoav Freund and Robert E. Schapire. "A decision-theoretic generalization of on-line learning and an application to boosting." In: *Journal of Computer and System Sciences* (1997).

[33] Yoav Freund and Robert E. Schapire. "Adaptive game playing using multiplicative weights." In: *Games and Economic Behavior* (1999).

[34] Pratik Gajane, Tanguy Urvoy, and Emilie Kaufmann. *Corrupt Bandits for Preserving Local Privacy*. 2017. arXiv: 1708.05033 [cs.LG]. URL: https://arxiv.org/abs/1708.05033.

[35] Ramki Gummadi, Ramesh Johari, and Jia Yuan Yu. "Mean field equilibria of multi armed bandit games." In: *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 2012, pp. 1110–1110. DOI: 10.1109/Allerton.2012.6483342.

[36] Anupam Gupta, Tomer Koren, and Kunal Talwar. "Better Algorithms for Stochastic Bandits with Adversarial Corruptions." In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. PMLR, 25–28 Jun 2019, pp. 1562–1578. URL: https://proceedings.mlr.press/v99/gupta19a.html.

[37] Amélie Heliou, Johanne Cohen, and Panayotis Mertikopoulos. "Learning with Bandit Feedback in Potential Games." In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017.

[38] Wassily Hoeffding. *Probability Inequalities for Sums of Bounded Random Variables*. 1963. DOI: https://doi.org/10.2307/2282952.

[39] Steven C.H. Hoi et al. "Online learning: A comprehensive survey." In: *Neurocomput.* 459.C (Oct. 2021), pp. 249–289. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2021.04.112. URL: https://doi.org/10.1016/j.neucom.2021.04.112.

[40] Branislav Kveton et al. "Cascading bandits: learning to rank in the cascade model." In: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*. ICML'15. Lille, France: JMLR.org, 2015, pp. 767–776.

[41] T.L Lai and Herbert Robbins. "Asymptotically efficient adaptive allocation rules." In: *Advances in Applied Mathematics* 6.1 (1985), pp. 4–22. ISSN: 0196-8858. DOI: https://doi.org/10.1016/0196-8858(85)90002-8. URL: https://www.sciencedirect.com/science/article/pii/0196885885900028.

[42] Tor Lattimore and Csaba Szepesvári. *Partial Monitoring*. July 2020. DOI: 10.1017/9781108571401.046.

[43] Huitian Lei et al. *An Actor-Critic Contextual Bandit Algorithm for Personalized Mobile Health Interventions*. 2022. arXiv: 1706.09090 [stat.ML]. URL: https://arxiv.org/abs/1706.09090.

[44] Shuai Li, Alexandros Karatzoglou, and Claudio Gentile. "Collaborative Filtering Bandits." In: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '16. Pisa, Italy: Association for Computing Machinery, 2016, pp. 539–548. ISBN: 9781450340694. DOI: 10.1145/2911451.2911548. URL: https://doi.org/10.1145/2911451.2911548.

[45] Nick Littlestone and Manfred K. Warmuth. "The weighted majority algorithm." In: *Information and Computation* (1994).

[46] Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. "Stochastic bandits robust to adversarial corruptions." In: STOC 2018. Los Angeles, CA, USA: Association for Computing Machinery, 2018, pp. 114–122. ISBN: 9781450355599. DOI: 10.1145/3188745.3188918. URL: https://doi.org/10.1145/3188745.3188918.

[47] Kanishka Misra, Eric M. Schwartz, and Jacob Abernethy. *Dynamic Online Pricing with Incomplete Information Using Multi-Armed Bandit Experiments*. 2019.

[48] Jonas W. Mueller, Vasilis Syrgkanis, and Matt Taddy. "Low-rank Bandit Methods for High-dimensional Dynamic Pricing." In: *Neural Information Processing Systems*. 2018. URL: https://api.semanticscholar.org/CorpusID:26832096.

[49] Francesco Orabona. *A Modern Introduction to Online Learning*. 2023. arXiv: 1912.13213 [cs.LG]. URL: https://arxiv.org/abs/1912.13213.

[50] Herbert E. Robbins. "Some aspects of the sequential design of experiments." In: *Bulletin of the American Mathematical Society* 58 (1952), pp. 527–535. URL: https://api.semanticscholar.org/CorpusID:15556973.

[51] Yevgeny Seldin and Gábor Lugosi. *An Improved Parametrization and Analysis of the EXP3++ Algorithm for Stochastic and Adversarial Bandits*. 2017. arXiv: 1702.06103 [cs.LG]. URL: https://arxiv.org/abs/1702.06103.

[52] Yevgeny Seldin and Aleksandrs Slivkins. "One practical algorithm for both stochastic and adversarial bandits." In: *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*. ICML'14. Beijing, China: JMLR.org, 2014, pp. II–1287–II–1295.

[53] Weiwei Shen et al. "Portfolio choices with orthogonal bandit learning." In: *Proceedings of the 24th International Conference on Artificial Intelligence*. IJCAI'15. Buenos Aires, Argentina: AAAI Press, 2015, pp. 974–980. ISBN: 9781577357384.

[54] Nícollas Silva et al. "Multi-Armed Bandits in Recommendation Systems: A survey of the state-of-the-art and future directions." In: *Expert Systems with Applications* 197 (Feb. 2022), p. 116669. DOI: 10.1016/j.eswa.2022.116669.

[55] Gen Tabei et al. "Multi-Armed Bandit-based Routing Method for In-network Caching." In: *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2021, pp. 1899–1902.

[56] W. R. Thompson. *On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples*. 1933. DOI: https://doi.org/10.2307/2332286.

[57] Huo X and Fu F. *Risk-aware multi-armed bandit problem with application to portfolio selection*. 2017. DOI: 10.1098/rsos.171377.

[58] Yisong Yue et al. "The K-armed dueling bandits problem." In: *Journal of Computer and System Sciences* 78.5 (2012). JCSS Special Issue: Cloud Computing 2011, pp. 1538–1556. ISSN: 0022-0000. DOI: https://doi.org/10.1016/j.jcss.2011.12.028. URL: https://www.sciencedirect.com/science/article/pii/S0022000012000281.

[59] Qian Zhou et al. "Large-Scale Bandit Approaches for Recommender Systems." In: *Neural Information Processing*. Ed. by Derong Liu et al. Cham: Springer International Publishing, 2017, pp. 811–821. ISBN: 978-3-319-70087-8.

[60] Julian Zimmert, Haipeng Luo, and Chen-Yu Wei. *Beating Stochastic and Adversarial Semi-bandits Optimally and Simultaneously*. 2019. arXiv: 1901.08779 [cs.LG]. URL: https://arxiv.org/abs/1901.08779.

[61] Julian Zimmert and Yevgeny Seldin. *Tsallis-INF: An Optimal Algorithm for Stochastic and Adversarial Bandits*. 2022. arXiv: 1807.07623 [cs.LG]. URL: https://arxiv.org/abs/1807.07623.