ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΛΟΓΙΚΗΣ ΚΑΙ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

Fairness Constraints and Reward Manipulation in Multi-Armed Bandits

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΑΡΙΝΑ
ΚΟΝΤΑΛΕΞΗ

Επιβλέπων: Δημήτριος Φωτάκης
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2024

Εθνικο Μετσοβιο Πολυτεχνειο
Τμημα Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων
Εργαστηριο Λογικης και Επιστημης Υπολογιστων

Fairness Constraints and Reward Manipulation in Multi-Armed Bandits

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΑΡΙΝΑ ΚΟΝΤΑΛΕΞΗ

Επιβλέπων: Δημήτριος Φωτάκης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 12η Ιουνίου 2024.

.....	.....	.....
Δημήτριος Φωτάκης Καθηγητής Ε.Μ.Π.	Χαρά Ποδηματά Επικ. Καθηγήτρια Μ.Ι.Τ.	Αριστείδης Παγουρτζής Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2024.

.....

Κονταλέξη Μαρίνα
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η παρούσα διπλωματική μελετά το multi-armed bandit πρόβλημα με στοχαστικές ανταμοιβές, όπου ένας learner παίζει ένα σειριακό παιχνίδι με ένα περιβάλλον για T γύρους. Σε κάθε γύρο, ο learner διαλέγει ένα από τα K "χέρια" μίας μηχανής slot και λαμβάνει μία ανταμοιβή που προέρχεται από κάποια στοχαστική κατανομή. Ο στόχος του learner είναι να παίξει όσο καλά θα έπαιζε η καλύτερη στρατηγική (δηλαδή η βέλτιστη γνωρίζοντας όλες τις κατανομές μέχρι τον τρέχοντα γύρο). Οι βέλτιστοι αλγόριθμοι εγγυώνται πως το regret του learner είναι φραγμένο από $\widetilde{O}(\sqrt{KT})$, το οποίο είναι το καλύτερο δυνατό φράγμα σύμφωνα με τη θεωρία πληροφορίας. Οι Joseph et al. [1] επιβάλλουν έναν επιπλέον περιορισμό δικαιοσύνης στον learner, που δεν του επιτρέπει να ευνοήσει ένα "χέρι" έναντι ενός άλλου εκτός εάν είναι σίγουρος για τη σχετική τους σύγκριση. Η εργασία μας προτείνει μία ε-χαλάρωση του ορισμού τους και έναν δίκαιο αλγόριθμο που πετυχαίνει $\widetilde{O}(\sqrt{\frac{1}{\varepsilon}}\sqrt{KT})$ regret. Οι εφαρμογές όπου έχει νόημα αυτός ο περιορισμός (όπως τα recommendation systems) είναι ευαίσθητες σε ανταγωνιστικές επιθέσεις (π.χ., ψεύτικες κριτικές), γι' αυτόν τον λόγο παρουσιάζουμε πώς συμπεριφέρονται γνωστοί αλγόριθμοι σε αυτό το μοντέλο και φιλοδοξούμε να καταλάβουμε τη σχέση ανάμεσα στους δίκαιους αλγόριθμους και σε αυτούς που είναι ανεκτικοί στις παραπάνω επιθέσεις.

Λέξεις κλειδιά: άμεση μάθηση, regret, multi-armed bandits, δικαιοσύνη, strategic manipulation, adversarial corruption.

Abstract

This thesis studies the stochastic multi-armed problem, where a *learner* plays a sequential game with an environment for T rounds. In each round the learner chooses one of the K available arms to pull and receives a stochastically generated reward. The goal of the learner is to perform as the best policy *in hindsight*. Optimal algorithms can guarantee that the learner's regret is bounded by $\widetilde{O}(\sqrt{KT})$, which matches the lower bound obtained by information theory. Joseph et al. [1] imposed a fairness constraint on the learner's actions, that restricts her from favoring an arm (i.e., pull it with higher probability) unless the arm is of greater merit. Our work proposes an ε-relaxation of their fairness definition and a *fair* algorithm that achieves $\widetilde{O}(\sqrt{\frac{1}{\varepsilon}}\sqrt{KT})$ regret. Applications where fairness is sought after (like recommendation systems) are vulnerable to adversarial attacks (e.g., fake reviews) thus we present the behaviour of known algorithms in the mixed model and aspire to connect *fair* algorithms with *robustness* to adversarial corruption.

Key words: online learning, regret, multi-armed bandits, fairness, strategic manipulation, adversarial corruption.

Ευχαριστίες

Ολοκληρώνοντας αυτή τη διπλωματική θα ήθελα να ευχαριστήσω τον επιβλέποντά μου κύριο Δημήτρη Φωτάκη που μου έδωσε την ευκαιρία να ασχοληθώ με ένα πραγματικά ενδιαφέρον και επίκαιρο πρόβλημα, για τις συμβουλές του και για την εμπιστοσύνη του, η οποία με τιμάει ειλικρινά. Θα ήθελα να πω ευχαριστώ στην κυρία Χαρά Ποδηματά και στον κύριο Κωνσταντίνο Καραμανή για τις ώρες που αφιέρωσαν δουλεύοντας για το project και για τη διάθεση να μου μου δείξουν πώς λειτουργεί μία ερευνητική ομάδα. Ξεχωριστό ευχαριστώ στον Αποστόλη Τσορβαντζή που ήταν δίπλα μου από την πρώτη στιγμή και με βοηθούσε σε ό,τι δυσκολία είχα, πολλές φορές πριν καν προλάβω να το ζητήσω η ίδια.

Σε προσωπικό επίπεδο, το κλίμα που χτίστηκε όλη τη χρονιά στο CoReLab ήταν πολύ φιλόξενο και χαίρομαι πολύ που γνώρισα όλα τα άτομα που συντέλεσαν σε αυτό. Κλείνοντας, η στήριξη των φίλων μου και της οικογένειάς μου ήταν καθοριστική σε όλα τα χρόνια των σπουδών μου και τους ευχαριστώ έναν προς έναν.

Contents

Chapter 1

Εκτεταμένη Ελληνική Περίληψη

Στο παρόν κεφάλαιο ακολουθεί μία εκταταμένη ελληνική παρουσίαση του περιεχόμενου αυτής της διπλωματικής. Τα υποκεφάλαια έχουν την ίδια δομή με αυτή της αγγλικής εκδοχής και ο αναγνώστης παραπέμπεται στα αντίστοιχα σημεία της για ορισμένες αποδείξεις που έχουν παραληφθεί.

1.1 Το Στοχαστικό Multi-Armed Bandit Πρόβλημα

Το Multi-Armed Bandit (MAB) Πρόβλημα είναι ένα απλό εργαλείο μοντελοποίησης ενός διαδοχικού παιχνιδιού ανάμεσα σε έναν **learner** και ένα **περιβάλλον**, που συμβαίνει ως κάποιον **χρονικό ορίζοντα**. Πριν προχωρήσουμε στον ορισμό των παραπάνω εννοιών, θα παρουσιάσουμε ορισμένες εφαρμογές που το μοντέλο αυτό μπορεί να φανεί χρήσιμο.

- **Καζίνο** Ένας τζογαδόρος παίζοντας διαφορετικές μηχανές slot ελπίζει να μεγιστοποιήσει το κέρδος του μέσω της εκμάθησης κάποιας στατιστικής πληροφορίας για τις ανταμοιβές κάθε μηχανής. Οι μηχανές slot καλούνται *one-armed bandits* και από εκεί προέρχεται το όνομα του προβλήματος.
- **Συστήματα Προτάσεων** Ένα σύστημα προτάσεων (recommendation system) μοντελοποιείται με έναν learner που επιθυμεί να μάθει τις προτιμήσεις των χρηστών όσο παρατηρεί την αλληλεπίδρασή τους (π.χ. κλικς ή likes) με διαφορετικές προτάσεις.
- **Δρομολόγηση** Μία εφαρμογή πλοήγησης επιθυμεί να μάθει τη συντομότερη διαδρομή μεταξύ δύο κόμβων ανάμεσα σε ένα εκθετικά μεγάλο σύνολο από συνδυασμούς δρόμων μέσω της ανάδρασης που λαμβάνει όταν ένας χρήστης επιλέγει να κάνει μία συγκεκριμένη διαδρομή.

Στις παραπάνω εφαρμογές εμφανίζεται το δίλημμα ανάμεσα στην εξερεύνηση επιλογών που δεν φαίνονται βέλτιστες και στην εκμετάλλευση των τρέχοντων βέλτιστων επιλογών. Η απάντηση στο παραπάνω είναι η κάταλληλη επιλογή γύρων εξερεύνησης

ώστε ο learner να μπορεί με (σχετική) βεβαιότητα να εκμεταλλευτεί τις πραγματικά βέλτιστες επιλογές.

Στη γενική περίπτωση το παιχνίδι παίζεται σε T γύρους. Σε κάθε γύρο t, ένας αλγόριθμος `ALG` (learner) επιλέγει μία ενέργεια A_t από ένα ορισμένο σύνολο δυνατών ενεργειών $\mathcal{A}$ και το περιβάλλον αποκαλύπτει την ανταμοιβή r_{A_t}. Η επιλογή A_t εξαρτάται από τις επιλογές του learner στους προηγούμενους γύρους και από τις αντίχτοιχες ανταμοιβές που έλαβε. Η αντιστοίχηση της ιστορίας $H_{t-1} = ((A_1, r_1), \ldots, (A_{t-1}, r_{t-1}))$ στην επιλογή A_t ονομάζεται **πολιτική** του learner και συμβολίζεται $\pi_t^{\texttt{ALG}} : \mathcal{H}_{t-1} \to \mathcal{A}$. Στην περίπτωση που ο learner μπορεί να διαλέξει μεταξύ K διαφορετικών ενεργειών/"χεριών". Το μοντέλο MAB εισήχθηκε απο τον Thompson [2] για χρήση σε ιατρικές δοκιμές και έκτοτε έχει μελετηθεί σε διάφορες μορφές του. Μεγάλο μέρος της έρευνας πάνω στο πρόβλημα μπορεί να βρεθεί στα συγγράμματα των Bubeck and Cesa-Bianchi [3], Lattimore and Szepesvári [4], Slivkins [5]. Ανάλογα με την εκάστοτε εφαρμογή το πρόβλημα μελετάται με πλήρη, μερική ή bandit ανάδραση (πληροφορία που αποκαλύπτεται από το περιβάλλον) και με ανταμοιβές που προκύπτουν από στοχαστικές κατανομές ή από κάποιον ανταγωνιστή (adversary).

1.1.1 Μοντέλο

Το μοντέλο που μελετάται στην παρούσα διπλωματική είναι το στοχαστικό MAB (SMAB) με bandit ανάδραση. Σε κάθε γύρο $t \in [T]$ ο learner διαλέγει μία ενέργεια $A_t \in [K]$ και παρατηρεί ανταμοιβή $r_{A_t} \sim \mathcal{D}_{A_t}$, όπου $\mathcal{D}_{A_t}$ είναι η κατανομή που ακολουθούν οι ανταμοιβές της ενέργειας A_t. Χωρίς βλάβη της γενικότητας, υποθέτουμε φραγμένες ανταμοιβές $r_{\cdot,t} \in [0, 1]$ για όλες τις ενέργειες και όλους τους γύρους $t \in [T]$.

Στην ανάλυση θα χρησιμοποιούμε τον συμβολισμό $\mu_i = \mathbb{E}[\mathcal{D}_i]$ για τη μέση τιμή της κατανομής $\mathcal{D}_i$. Ορίζουμε την ενέργεια με τη μέγιστη μέση ανταμοιβή $\mu^* = \max_{i\in[K]} \mu_i$ ως τη βέλτιστη ενέργεια και συμβολίζουμε $i^* = \arg\max_{i\in[K]} \mu_i$. Επιπλέον, θα χρειαστούμε τον συμβολισμού $\Delta_i = \mu^* - \mu_i \geq 0$ με την ισότητα να ισχύει μόνο για $i = i^*$.

Για να μετρήσουμε την απόδοση ενός learner χρησιμοποιούμε την έννοια του regret, δηλαδή της απόστασής της συνολικής ανταμοιβής του learner από τη βέλτιστη ανταμοιβή εκ των υστέρων (γνωρίζοντας τις κατανομές $\mathcal{D}_i$). Έτσι έχουμε την παρακάτω έκφραση για το regret.

$$R(T) = \sum_{t=1}^{T} \mathbb{E}[r_{i^*,t}] - \sum_{t=1}^{T} \mathbb{E}[r_{A_t,t}] = \sum_{t=1}^{T} (\mu^* - \mu_{A_t}) = \sum_{t=1}^{T} \Delta_{A_t}.$$

Συμβολίζουμε με $n_{i,t}$ το σύνολο των φορών που παίξαμε την ενέργεια i. Τότε το regret γραφεται ως:

$$R(T) = \sum_{i=1}^{K} n_{i,T} \cdot \Delta_i.$$

1.1.2 Κλασικοί Bandit Αλγόριθμοι

Η θεωρία πληροφορίας μας δίνει το εξής **κάτω φράγμα**:

Θεώρημα 1.1.1. *Έστω T ένας χρονικός ορίζοντας και K ο αριθμός των πιθανών ενεργειών. Για οποιονδήποτε bandit αλγόριθμο υπάρχει ένα στιγμιότυπο τέτοιο ώστε*

$$\mathbb{E}[R(T)] \geq \Omega(\sqrt{KT}).$$

Η απόδειξη βασίζεται σε επιχειρήματα που χρησιμοποιούν την Kullback-Leibler (KL) απόκλιση (ή σχετική εντροπία) μεταξύ των κατανομών δύο διαφορετικών ενεργειών και υπολογίζουν τον ελάχιστο αριθμό δοκιμών ώστε να μπορούμε με βεβαιότητα να ξεχωρίσουμε τις κατανομές μεταξύ τους. Οι βασικές ιδιότητες της απόκλισης KL που χρειάζονται για την απόδειξη βρίσκονται στο A.1.4 και η πλήρης απόδειξη στο 3.5.1.

Στη βιβλιογραφία έχουν προταθεί αλγόριθμοι που πετυχαίνουν regret που συμπίπτει με το άνωθεν φράγμα. Θα παρουσιάσουμε στον αλγόριθμο `Successive Elimination (SAE)` των Even-Dar et al. [6]. Ανήκει στην κατηγορία των προσαρμοστικών αλγορίθμων, οι οποίοι προσαρμόζουν τις επιλογές τους μετά από κάθε γύρο (ή ομάδα γύρων) εξερεύνησης. Παρακάτω έχουμε την εκδοχή του αλγορίθμου που παρουσιάζεται στο [5]. Κατά την εκτέλεσή του, ο αλγόριθμος ανανεώνει διαστήματα εμπιστοσύνης για τα μ_i της μορφής $[l_{i,t}, u_{i,t}]$ για κάθε επιλογή i και κάθε γύρο t επιλέγοντας κατάλληλη ακτίνα εμπιστοσύνης $\texttt{rad}_t(i) = \sqrt{2\log T / n_{i,t}}$. Συγκεκριμένα,

$$u_{i,t} = \widehat{\mu}_{i,t} + \texttt{rad}_t(i),$$

$$l_{i,t} = \widehat{\mu}_{i,t} - \texttt{rad}_t(i),$$

όπου $\widehat{\mu}_{i,t}$ είναι ο μέσος όρος των ανταμοιβών από την επιλογή i ως τον γύρο t.

Algorithm 1: `Successive Elimination`

Είσοδος: K, T
`/* Αρχικοποίηση active set */`
$S \leftarrow [K]$
while $t \leq T$ **do**
  `/* Παίξε κάθε active arm μία φορά */`
  for $i \in S$ **do**
    Παίξε arm i
    Ανανέωσε u_i, l_i
  `/* Πολιτική εξάλειψης */`
  for $i \in S$ **do**
    if $\exists j \in S$ *τέτοιο ώστε* $l_j > u_i$ **then**
      $S \leftarrow S \setminus i$

Θεώρημα 1.1.2. *Ο αλγόριθμος `Successive Elimination` πετυχαίνει regret*

$$\mathbb{E}[R(t)] \leq O(\sqrt{Kt\log T}),$$

για κάθε γύρο $t \leq T$.

Η απόδειξη βρίσκεται στο 3.3.1.

Συνεχίζουμε κάνοντας αναφορά στον αλγόριθμο `Upper Confidence Bound (UCB)`, έναν κομψό και αποδοτικό αλγόριθμο που πετυχαίνει βέλτιστο regret. Η αρχική εκδοχή του είναι αυτή του `UCB1` από το Auer et al. [7].

Algorithm 2: UCB

Είσοδος: K, T, δ
for $t \in [T]$ **do**
  $A_t \leftarrow \arg\max_{i \in [K]}(u_{i,t})$ // $u_{i,t} = \widehat{\mu}_{i,t} + \texttt{rad}_t(i)$

Ο UCB ακολουθεί την αρχή **"Αισιοδοξία υπό Αβεβαιότητα"**, δηλαδή εμπιστεύεται πάντα την πιο αισιόδοξη πρόβλεψη κατά την επιλογή του.

Θεώρημα 1.1.3. *Ο αλγόριθμος UCB πετυχαίνει regret*

$$\mathbb{E}[R(t)] \leq O(\sqrt{Kt \log T}),$$

για κάθε γύρο $t \leq T$.

Η απόδειξη βρίσκεται στο 3.3.2.

1.2 Δίκαιοι Αλγόριθμοι στο MAB Πρόβλημα

Πλήθος εφαρμογών αλγοριθμικής λήψης αποφάσεων απαιτούν την τήρηση περιορισμών που εγγυώνται την *"δικαια"* απόφαση ενός αλγορίθμου, όπως η προσωποποιημένη διαφήμηση, οι ιατρικές δοκιμές, οι διαδικασίες πρόσληψης ή δανεισμού και πολλές άλλες. Παραθέτουμε ενδεικτικά τη σειρά εργασιών [8, 9, 10] όπου αναλύονται περιστατικά όπου αλγόριθμοι λήψης αποφάσεων αναπαράγουν biases. Φυσικά, ο ορισμός της δίκαιης απόφασης δεν είναι προφανής γι' αυτό το λόγο έχουν προταθεί διάφοροι ορισμοί δικαιοσύνης για το υπό μελέτη μοντέλο. Θα εστιάσουμε στον ορισμό δ-fairness των Joseph et al. [1].

Ορισμός 1.2.1. *Ένας αλγόριθμος είναι* δ*-fair αν για όλες τις ακολουθίες από ανταμοιβές* $r_{A_1}, r_{A_2}, \ldots, r_{A_t}$ *και όλες τις κατανομές* $\mathcal{D}_1, \ldots, \mathcal{D}_K$ *με πιθανότητα τουλάχιστον* $1 - \delta$ *πάνω στην ιστορία* h*, για κάθε γύρο* $t \in [T]$ *και όλα τα ζεύγη επιλογών* $j, j' \in [K]$*,*

$$\pi_t(j|h) > \pi_t(j'|h) \text{ only if } \mu_j > \mu_{j'}.$$

Ο παραπάνω ορισμός εξασφαλίζει πως ένας learner δεν μπορεί να ευνοήσει ένα arm έναντι ενός άλλου αν το πρώτο δεν έχει υψηλότερη μέση ανταμοιβή. Οι συγγραφείς προτείνουν τον αλγόριθμο FairBandits ο οποίος είναι μία δ-fair παραλλαγή του Successive Elimination και πετυχαίνει $\widetilde{O}(\sqrt{K^3 T \ln TK/\delta})$ regret. Το αποτέλεσμα είναι tight με το αντίστοιχο κάτω φράγμα που υπολογίζουν. Στην περιγραφή του αλγορίθμου χρησιμοποιείται η διμερής σχέση της σύνδεσης (link), που αναφέρεται σε arms των οποίων τα διαστήματα εμπιστοσύνης επικαλύπτονται και η διμερής σχέση της αλυσίδας (chain) που συσχετίζει arms που βρίσκονται στην ίδια κλάση της μεταβατικής κλειστότητας της σχέσης σύνδεσης.

Algorithm 3: `FairBandits`

Είσοδος: K, δ.
/* Αρχικοποίηση active set και διαστημάτων εμπιστοσύνης */
$S_0 \leftarrow \{1, \dots, K\}$.
for $i \in [K]$ **do**
  $\widehat{\mu}_{i,0} \leftarrow 1/2, u_{i,0} \leftarrow 1, l_{i,0} \leftarrow 0, n_{i,0} \leftarrow 0$
for $t \in [T]$ **do**
  $i_t^* \leftarrow \arg\max_{i \in S_{t-1}} u_{i,t}$
  $S_t \leftarrow \{j | j \text{ chains to } i_t^*, j \in S_{t-1}\}$
  Παίξε arm $j \in S_t$ ομοιόμορφα
  Παρατήρησε ανταμοιβή $r_{j,t}$
  /* Ανανέωσε τις στατιστικές πληροφορίες για το arm j */
  $n_{j,t} \leftarrow n_{j,t-1} + 1$
  $\widehat{\mu}_{j,t} \leftarrow \frac{1}{n_{j,t}}(\widehat{\mu}_{j,t-1} \cdot n_{j,t-1} + r_{j,t})$
  $\texttt{rad}_t(j) \leftarrow \sqrt{\frac{\ln((\pi \cdot (t+1))^2)/3\delta}{2n_{j,t}}}$
  $[l_{j,t}, u_{j,t}] \leftarrow [\widehat{\mu}_{j,t} - \texttt{rad}_t(j), \widehat{\mu}_{j,t} + \texttt{rad}_t(j)]$
  for $i \in S_t, i \neq j$ **do**
    $\widehat{\mu}_{i,t} \leftarrow \widehat{\mu}_{i,t-1}, u_{i,t} \leftarrow u_{i,t-1}, l_{i,t} \leftarrow l_{i,t-1}, n_{i,t} \leftarrow n_{i,t-1}$

1.3 (ε, δ)-Fairness

Στο παρόν υποκεφάλαιο βρίσκεται η βασική συνεισφορά αυτής της διπλωματικής. Συγκεκριμένα, προτείνουμε την παρακάτω ε-χαλάρωση του δ-fair ορισμού.

Ορισμός 1.3.1. *Ένας αλγόριθμος είναι (ε, δ)-fair αν για όλες τις ακολουθίες από ανταμοιβές $r_{A_1}, r_{A_2}, \dots, r_{A_t}$ και όλες τις κατανομές $\mathcal{D}_1, \dots, \mathcal{D}_K$ με πιθανότητα τουλάχιστον $1 - \delta$ πάνω στην ιστορία h, για κάθε γύρο $t \in [T]$ και όλα τα ζεύγη επιλογών $j, j' \in [K]$,*

$$\pi_t(j|h) > \pi_t(j'|h) + \varepsilon \text{ only if } \mu_j > \mu_{j'}.$$

Η διαφορά των δύο ορισμών έγκειται στο τι θεωρούν ως "δείχνω εύνοια" απέναντι σε ένα arm. Στο πλαίσιο του (ε, δ)-fairness, θεωρείται πως ο learner ευνοεί ένα arm έναντι ενός άλλου αν τα παίζει με πιθανότητες που απέχουν τουλάχιστον ε. Αυτή η χαλάρωση επιτρέπει στον learner να "σπάει" γρηγορότερα τις αλυσίδες και να παίζει με μεγαλύτερη πιθανότητα τα arms με καλύτερο μέσο όρο ανταμοιβών. Για $\varepsilon = 0$, οι δύο ορισμοί ταυτίζονται, ενώ για $\varepsilon = 1$ ο περιορισμός είναι τόσο ελαστικός που μία παραλλαγή του `Successive Elimination` θα αρκούσε για την επίλυση του προβλήματος. Παρακάτω παρουσιάζεται ο αλγόριθμος `FairTruthful (FT)` και η υπο-ρουτίνα `Grouping` που χρησιμοποιείται από τον `FT`.

Algorithm 4: `Fair algorithm for Truthful agents` (FT)

Είσοδος: K, ε, δ.
/* Αρχικοποίηση active set και διαστημάτων εμπιστοσύνης */
$S_0 \leftarrow \{1, \dots, K\}$.
for $i \in [K]$ **do**
  $\widehat{\mu}_{i,0} \leftarrow 1/2, u_{i,0} \leftarrow 1, l_{i,0} \leftarrow 0, n_{i,0} \leftarrow 0$
for $t \in [T]$ **do**
  /* υπολόγισε την κατανομή π_t πάνω στα active arms. */
  $\pi_t \leftarrow \texttt{Grouping}(S_{t-1}, \varepsilon)$ (13)
  Παίξε arm $j \sim \pi_t$.
  Παρατήρησε ανταμοιβή $r_{j,t}$.
  /* Ανανέωσε τις στατιστικές πληροφορίες για το arm j */
  $n_{j,t} \leftarrow n_{j,t-1} + 1$
  $\widehat{\mu}_{j,t} \leftarrow \frac{1}{n_{j,t}}(\widehat{\mu}_{j,t-1} \cdot n_{j,t-1} + r_{j,t})$
  $\texttt{rad}_t(j) \leftarrow \sqrt{\frac{\ln((\pi \cdot (t+1))^2)/3\delta}{2n_{j,t}}}$
  $[l_{j,t}, u_{j,t}] \leftarrow [\widehat{\mu}_{j,t} - \texttt{rad}_t(j), \widehat{\mu}_{j,t} + \texttt{rad}_t(j)]$
  $i_t^\star \leftarrow \arg\max_{i \in S_{t-1}} u_{i,t}$
  $S_t \leftarrow \{i | i \text{ chains to } i_t^\star\}$
  for $i \in S_t, i \neq j$ **do**
    $\widehat{\mu}_{i,t} \leftarrow \widehat{\mu}_{i,t-1}, u_{i,t} \leftarrow u_{i,t-1}, l_{i,t} \leftarrow l_{i,t-1}, n_{i,t} \leftarrow n_{i,t-1}$

Algorithm 5: Grouping

Είσοδος: Active set S_t, ε.
/* Αρχικοποίησε τον αριθμό των groups και το σύνολο των arm που δεν έχουν τοποθετηθεί ακόμα */
$M \leftarrow 0$
$\text{NA} \leftarrow S_t$
/* Χώρισε τα active arms σε groups */
while $NA \neq \emptyset$ **do**
  Ανανέωσε group counter M: $M \leftarrow M + 1$
  Pivot arm για το group M: $j^\star \leftarrow \arg\max_{i \in \text{NA}} u_{i,t}$.
  M: $G_{M,t} \leftarrow \{i \in \text{NA} : u_{i,t} \geq l_{j^\star,t} \text{ or } u_{i,t} \geq l_{k,t}, k \in G_{M-1,t}\}$.
  $\text{NA} \leftarrow \text{NA} \setminus G_{M,t}$.
Λύσε το παρακάτω LP: // Υπολόγισε την κατανομή

$$
\begin{aligned}
&\text{μεγιστοποίησε} && \tilde{\pi}_{1,t} \\
&\text{subject to} && \sum_{i=1}^{M} |G_{i,t}| \tilde{\pi}_{i,t} = 1, \\
& && \tilde{\pi}_{i,t} \leq \tilde{\pi}_{i+1,t} + \varepsilon, \quad i = 1, \dots, M-1 \\
& && \tilde{\pi}_{i,t} \geq 0 \qquad\qquad i = 1, \dots, M
\end{aligned}
$$

Επιστροφή: Κατανομή π_t: $\{\pi_{i,t} \leftarrow \tilde{\pi}_{j,t} \text{ s.t. } i \in G_{j,t}, \forall j \in [M]\}$

Η πλήρης απόδειξη του άνω φράγματος του regret του FT μπορεί να βρεθεί στο 5. Παρακάτω παρουσιάζουμε τα βασικά λήμματα και το σχετικό θεώρημα.

Λήμμα 1.3.1. *Για κάθε γύρο $t \in [T]$, ο αριθμός των group με μη μηδενική πι-*

θανότητα $\tilde{\pi}_{\cdot,t}$ *είναι άνω φραγμένος από* $m = O(\min\{\sqrt{1/\varepsilon}, K\}$.

Λήμμα 1.3.2. *Έστω ένα ζεύγος arms* i, j *τέτοιο ώστε* $\mu_i > \mu_j$ *και γύρος* $t \in [T]$. *Τότε,* $\mathbb{E}[n_{i,t}] \geq \mathbb{E}[n_{j,t}]$.

Θεώρημα 1.3.1. *Ο αλγόριθμος* `FT` *πετυχαίνει regret*

$$R(T) = O\left(\min\left(\sqrt{\frac{1}{\varepsilon}}, K\right)\sqrt{KT\log\frac{KT}{\delta}}\right).$$

1.4 Ανταγωνιστικές Επιθέσεις σε Στοχαστικά Bandits

Το τελευταίο κεφάλαιο πραγματεύεται το συνδυαστικό μοντέλο στοχαστικού MAB με ανταγωνιστικές (adversarial) αλλοιώσεις στις ανταμοιβές, που εισάγεται από τους Lykouris et al. [11]. Το μοντέλο αυτό είναι χρήσιμο σε εφαρμογές όπου η ανταμοιβή μίας ενέργειας μπορεί να παραποιηθεί από κάποιον ανταγωνιστή με ή χωρίς κάποιον σαφή στρατηγικό στόχο, όπως τα fake reviews, το click-fraud κ.ά. Το πρωτόκολλο μεταξύ learner και περιβάλλοντος έχει ως εξής:

1. Ο learner επιλέγει δημόσια μία κατανομή π_t.
2. Το περιβάλλον θέτει τη στοχαστική ανταμοιβή $r_{i,t}$ για κάθε ενέργεια i.
3. Ο ανταγωνιστής παρατηρεί τις στοχαστικές ανταμοιβές, την ιστορία h_{t-1} και την κατανομή π_t και επιστρέφει τις παραποιημένες ανταμοιβές $(\widehat{r}_{i,t})_{i\in[K]}$
4. Ο learner παίζει μία ενέργεια σύμφωνα με την τυχαιοποίηση της κατανομής του και παρατηρεί την παραποιημένη της ανταμοιβή.

Η αλλοιωμένες ανταμοιβές έχουν τη μορφή $\widehat{r}_{a,t} = r_{a,t} + c_{a,t}$, όπου το $c_{a,t}$ επιλέγεται από τον ανταγωνιστή. Χαρακτηρίζουμε έναν ανταγωνιστή C-corrupted αν

$$\sum_t \max_a |\widehat{r}_{a,t} - r_{a,t}| \leq C.$$

Εναλλακτικά, οι Gupta et al. [12] ορίζουν το επίπεδο αλλοίωσης C ως

$$C = \sum_{t\in[T]} \left\| \widehat{R}_t - R_t \right\|.$$

Οι κλασικοί bandit αλγόριθμοι που είδαμε στο προηγούμενο κεφάλαιο δεν είναι robust απέναντι σε τέτοιες αλλοιώσεις. Ωστόσο, είναι ενδιαφέρον ότι αν ο ανταγωνιστής μπορεί μόνο να παραποιήσει την ανταμοιβή ώστε $\widehat{r}_{i,t} \geq r_{i,t}$, τότε ο αλγόριθμος `UCB` διατηρεί το regret του απέναντι σε $\widetilde{O}(\sqrt{KT})$-corrupted ανταγωνιστή. Το πλήρες θεώρημα ακολουθεί.

Θεώρημα 1.4.1. *Στο συνδυαστικό μοντέλο με* $c_{\cdot,t} \geq 0$, *ο αλγόριθμος* `UCB` *πετυχαίνει regret*

$$R(t) = O\left(\sum_{i\in[K], i\neq i^*}\left[3\Delta_i + \frac{16\log T}{\Delta_i}\right] + 4KC\right),$$

σε κάθε γύρο t.

Ο όρος του αθροίσματος είναι το instance-dependent regret του `UCB` στο απλό στοχαστικό MAB.

Οι Lykouris et al. [11] προτείνουν τον αλγόριθμο *Multi-layer Active Arm Elimination Race* (14) που πετυχαίνει regret εξαρτώμενο πολλαπλασιαστικά από την τιμή C. Ο αλγόριθμος `BARBAR` των Gupta et al. [12] ρίχνει την εξάρτηση σε γραμμική και τον παρουσιάζουμε παρακάτω.

Algorithm 6: `BARBAR`

Παράμετροι: εμπιστοσύνη $\delta \in (0,1)$, T.
Initialize $T_0 = 0$ and $\Delta_i^0 = 1$ για κάθε $i \in [K]$.
Θέσε $\lambda = 1024 \ln(\frac{8K}{\delta} \log_2 T)$.
for $m = 1, 2, \ldots$ **do**
  Set $n_i^m = \lambda(\Delta_i^{m-1})^{-2}$ για κάθε $i \in [K]$.
  Θέσε $N_m = \sum_{i=1}^K n_i^m$ και $T_m = T_{m-1} + N_m$.
  for $t = T_{m-1} + 1$ *to* T_m **do**
    Παίξε το arm i με πιθανότητα n_i^m / N_m.
  Θέσε $r_i^m = S_i / n_i^m$ όπου S_i είναι η συνολική ανταμοιβή από το arm i στην εποχή m.
  Θέσε $r_\star^m = \max_i \{r_i^m - \frac{1}{16}\Delta_i^{m-1}\}$
  Θέσε $\Delta_i^m = \max\{2^{-m}, r_\star^m - r_i^m\}$

Θεώρημα 1.4.2. *Με πιθανότητα τουλάχιστον $1-\delta$, το regret του Algorithm 15 είναι φραγμένο από*

$$O\left(KC + \sum_{i \neq i^*} \frac{\log T}{\Delta_i} \log\left(\frac{K}{\delta} \log T\right)\right).$$

Chapter 2

Introduction

2.1 Motivation

Reinforcement learning (RL) is a computational framework to model a goal-oriented learning procedure between a learner and an environment. What differentiates RL from other types of learning is that learning stems from the interaction between a decision-making agent and an environment without any supervision or exemplary data. That is to say, a numerical reward function that maps a learner's actions to rewards is sufficient learning feedback. The goal in RL applications is not to uncover a hidden structure behind the environment but rather to find the decision-making policy that maximizes the cumulative reward.

The non-associative setting, where the learner only needs to make one type of decision is modeled with **multi-armed bandits** (MAB). In this model, in each round t the learner is faced with K arms of a multi-armed slot machine and must decide which one to pull. The rewards may be generated stochastically or adversarially depending on the application. The main metric to evaluate a learner's performance is **regret**, i.e., the difference between the cumulative reward of the learner's policy and the cumulative reward of the optimal policy *in hindsight*. Our work focuses on stochastic MAB and the metric used is pseudo-regret, i.e., the expectation of the difference above.

In this context, the **Exploration-Exploitation** trade-off arises. The learner must choose whether she will explore more options to get more information upon their merit or whether she should commit to the best performing option so far. Such dilemmas dominate most decision-making procedures and have been studied by a plethora of disciplines such as mathematics and behavioral science. Chapter 3 formally presents the model and some of the standard algorithms used in balancing exploration-exploitation.

Many algorithmic decision-making applications have been found to replicate biases, against their designers' will. Such behavior emerges from historical biases or poor representation of a certain population in a training dataset, decisions based on protected attributes, a platform's reward maximization objective among other reasons. Thus, a line of work has studied different fairness notions, which wish

to guarantee that such phenomena are eliminated. **Algorithmic Fairness** can be divided into *group* fairness, that ensures that demographic parity/ equality of odds/equality of opportunities are respected among populations with different protected attributes (be it race, gender, age); and *individual* fairness, that can be interpreted as "Similar individuals are treated similarly". Chapter 4 includes a brief presentation of works uncovering biases in algorithmic decision-making and some insightful fairness definitions.

However, fairness may not be guaranteed in settings where an action's reward is **manipulated**. Multiple applications are vulnerable to external agents that distort observed rewards, either serving an individual objective or adversarially. A motivating example is the case of Goodreads, a platform that suffers from *fake reviews*, a type of adversarial attack that aims at fooling the algorithm into recommending books with poor performance by corrupting the rewards (i.e., review score) of better performing books [13]. The work of Lykouris et al. [11] models the aforementioned case through a mixed stochastic and adversarial setting and studies algorithms robust to such corruptions. Chapter 6 thoroughly presents results on the mixed model.

2.2 Previous Work

Multi-Armed Bandits have been introduced as a framework by Thompson [2], and taken their name from the work of Bush and Mosteller [14]. The MAB problem is extensively documented in the works of Bubeck and Cesa-Bianchi [3], Lattimore and Szepesvári [4], Slivkins [5]. The work of Auer et al. [15] shows that any bandit algorithm is forced to suffer $\Omega(\sqrt{KT})$ regret, using tools from information theory. Algorithms that match the lower bound (up to a polylogarithmic factor) include the `Successive Elimination` Algorithm (8) proposed in Even-Dar et al. [6] and the `Upper Confidence Bound` Algorithm (9) from Auer et al. [7].

Different fairness notions have been introduced in [16, 17, 18, 19, 20]. This thesis focuses on the definition of δ-*fairness* proposed by Joseph et al. [1], a meritocratic notion that restricts a learner from favoring a sub-optimal arm. The authors propose a δ-fair algorithm, called `FairBandits` (11), that achieves $\widetilde{O}(\sqrt{K^3T})$ regret after T rounds.

When it comes to the mixed model of Lykouris et al. [11], the work of Gupta et al. [12] proposes algorithm `BARBAR` that achieves regret with an additive dependence on the corruption level C. The literature on agents with strategic behavior includes the work of Feng et al. [21], who study a model where arms behave as strategic agents who wish to maximize the number of times they get pulled by overvaluing their own reward up to a budget. Similarly, Braverman et al. [22] consider a setting where agents/arms present a lower reward, keeping the difference from the realized one as a utility for themselves. A combination of the above is studied in Esmaeili et al. [23].

2.3 Contribution

Our contribution is an ε**-relaxation** of the δ-fair definition from Joseph et al. [1]. In our definition, a learner *favors* an arm over another if she is playing them

with probabilities more than ε away from each other. Through this relaxation, we introduce `FairTruthful (FT)` Algorithm (12), that achieves a gracefully optimized regret of $\widetilde{O}\left(\min\left(\sqrt{1/\varepsilon}\right)\sqrt{KT}\right)$, where parameter ε can be tuned to cater to each application. The regret analysis of `FT` can be found in Chapter 5. In Chapter 6 we also provide an analysis of the performance of `UCB` in the adversarial corruption setting. The link between fair and robust algorithms is an open question.

Chapter 3

The Stochastic Multi-Armed Bandit Problem

The Multi-Armed Bandit Problem is a simple framework to model a sequential game between a **learner** and an **environment** taking place over a **time horizon**. Before formally defining the terms above, let us present some applications where this model may be of use.

- **Casino** A gambler playing with multiple slot machines aims at maximizing her revenue through learning some (maybe statistical) information on the rewards of each machine. Slot machines are also called *one-armed bandits* and that is how the name Multi-Armed Bandit problem occurred.
- **Recommendation Systems** A recommendation system (learner) wishes to learn users' preferences while observing their interaction (i.e., clicks or likes) with different recommendations.
- **Network Routing** A navigation application opts to find the shortest path between two nodes among a combinatorially large set of available paths through feedback provided each time a path is chosen.

In the general case the game is played over a horizon of T rounds. In each round t a bandit algorithm (learner) chooses an action A_t from a fixed set of actions $\mathcal{A}$ and the environment reveals the action's reward r_{A_t}. Naturally, a learner's choice in round t can only depend on the rewards she has observed and not on future rewards. The mapping of history $H_{t-1} = \big((A_1, r_{A_1}), (A_2, r_{A_2}), \ldots, (A_{t-1}, r_{A_{t-1}})\big)$ to the action A_t is called the learner's **policy**. The Multi-Armed Bandit Problem (MAB) is often referred to as the K**-armed** Bandit Problem, where K denotes the number of possible actions/arms. For the rest of the analysis we will be using the terms *action* and *arm* interchangeably.

The MAB Problem was introduced in Thompson [1933] as an attempt to efficiently conduct medical trials. Since then, a line of work (Bubeck and Cesa-Bianchi [2012], Lattimore and Szepesvári [2020], Slivkins [2022]) has thoroughly contributed in enriching and collecting most of the main results on the field. Given this general

setup, multiple variants of the MAB problem have been studied to cater to each application. Depending on the information provided by the environment after choosing an action, we divide bandit problems into three main classes: (i) *bandit feedback*, when the learner is only informed about the reward of the arm she chose; (ii) *full feedback*, when the learner is informed about the rewards of all possible actions and (iii) *partial feedback*, when the information provided lies between the two former cases.

Another major classification stems from the way rewards are generated. The two main categories are *stochastic* rewards, when rewards are i.i.d. samples drawn from a fixed distribution $\mathcal{D}$ over the arms and *adversarial* rewards, when we assume the existence of an adversary that sets an action's reward arbitrarily or subject to some constraints.

The main question that arises from this setup is how to evaluate a learner given a specific environment. To answer this question, the notion of **regret** was introduced. Regret is defined with respect to a policy π and accounts for the difference between the cumulative reward collected by policy π and the cumulative reward collected by the learner. Generalizing this notion, one can compute regret with respect to a family of policies Π as the maximum regret w.r.t. any policy π in Π. The regret of an algorithm $\mathcal{A}$ in a MAB problem can be expressed as:

$$R(T) = \max_{\pi \in \Pi} \left(\sum_{t=1}^{T} \mu_{i_t \sim \pi} \right) - \sum_{t=1}^{T} \mu_{i_t \sim \pi_{\mathcal{A}}}.$$

The above is often called *pseudo-regret* in literature because the difference is computed with respect to the mean values and not the realized rewards.

The choice of Π differs depending on the variant of MAB being studied. The present work focuses on **stochastic** MAB problem and therefore the regret computed is relative to the policy π^* that chooses the arm with the highest mean reward.

Applications

We hereby reference some indicative works on bandit applications in a plethora of different domains. Bouneffouf and Rish [24] thoroughly present bandit literature on real-life applications in the fields of healthcare [25, 26, 27], finance [28, 29], dynamic pricing [30, 31], anomaly detection [32, 33] and more. Recommendation systems applications can be found in [34, 35, 36, 37], while the respective literature has been enriched with works upon users' interaction with a bandit algorithm [38, 39, 40]. Bandit applications to Internet routing and congestion control and communications are presented in [41, 42, 43].

3.1 Model

In this section we formally define the model studied throughout this thesis. Our model falls into the category of stochastic MAB with bandit feedback.

In each round t in $[T]$ learner picks an arm A_t among K available arms and observes reward $r_{A_t,t} \sim \mathcal{D}_{A_t}$, where $\mathcal{D}_{A_t}$ is the distribution associated with arm a. We assume bounded rewards such that $r_{\cdot,t} \in [0, 1]$ for all arms and all rounds $t \in [T]$.

The mean value of $\mathcal{D}_i$ will be of utmost importance in our analysis, so we will use $\mu_i = \mathbb{E}[\mathcal{D}_i]$ for simplicity in notation. We define the *best* arm i^* to be the arm with the highest mean reward μ^*, namely:

$$i^* = \arg\max_{i \in [K]} (\mu_i).$$

Without loss of generality we assume that only one arm satisfies the above property. We further define $\Delta_i = \mu_{i^*} - \mu_i$ as the *reward gap* for arm i. It is clear from the definition of the best arm that $\Delta_i \geq 0$ for all arms, with the equality holding only for $i = i^*$.

Computing the pseudo-regret relative to the policy π^* which picks arm i^* with probability 1 we get the following expression:

$$R(T) = T \cdot \mu^* - \sum_{t=1}^{T} \mu_{i_t} = \sum_{t=1}^{T} (\mu^* - \mu_{i_t}) = \sum_{t=1}^{T} \Delta_{i_t}. \tag{3.1}$$

Let $n_{i,t}$ be the total number of pulls of arm i until round t. Then the above expression can be written as:

$$R(T) = \sum_{i=1}^{K} n_{i,T} \cdot (\mu^* - \mu_i) = \sum_{i=1}^{K} n_{i,T} \cdot \Delta_i. \tag{3.2}$$

In the following sections, we present the basic algorithms proposed for the Stochastic MAB problem and their regret analysis.

3.2 Non-Adaptive Exploration

We begin with briefly presenting algorithms that follow non-adaptive exploration. *Non-adaptive exploration* refers to algorithms that neglect the history of observed rewards in some exploration rounds. A round t is characterized as an *exploration round* if the observed tuple (A_t, r_{i_t}) is used by the algorithm in a future round. For a deterministic algorithm to be considered non-adaptive, it should define the exploration rounds as well as the choice of arms in all of them before the first round. For a randomized algorithm, the criterion is altered to satisfying the above property for any realization of its random seed.

Although not having optimal performance in terms of regret, a brief analysis of non-adaptive algorithms may be beneficiary to the reader's understanding of the model.

A simple such algorithm is uniformly exploring all available actions and then committing to the empirically best one. This is the case of `Explore-First` Algorithm in Slivkins [5] (also appearing under the name of `Explore-Then-Commit (ETC)` Algorithm in Lattimore and Szepesvári [4]), one of the first bandit algorithms introduced in Robbins [1952], Anscombe [1963]. Inserting randomness into the `Explore-First` Algorithm we get the ϵ-`Greedy` Algorithm, both achieving $\widetilde{O}(T^{2/3})$

regret[1]. The performance of the latter is thoroughly documented in Sutton and Barto [46]. We present the algorithm below.

Algorithm 7: ϵ-`Greedy`

Input: number of arms K, horizon T, ϵ.
for $t \in [T]$ **do**
    Toss a coin with heads probability ϵ_t
    if *heads* **then**
        Explore an arm uniformly at random
    else
        Exploit arm with highest empirical mean

We can see that rounds in ϵ-`Greedy` are clearly divided into exploration and exploitation rounds, depending on the outcome of the coin flip. Before proceeding with computing the regret, we define the terms clean and bad event that appear in most of the following proofs.

The **empirical mean reward** $\widehat{\mu}_{i,t}$ of arm i until round t can be expressed as:

$$\widehat{\mu}_{i,t} = \frac{\sum_{t'=1}^{t} r_{i,t'}}{n_{i,t}}.$$

The distance $|\widehat{\mu}_{i,t} - \mu_i|$ can be bound using the Hoeffding Inequality (A.1.1):

$$\mathbb{P}\left[\left|\widehat{\mu}_{i,t} - \mu_i\right| \geq \mathtt{rad}_t(i)\right] \leq 2\exp(-2\mathtt{rad}_t(i)^2 \cdot n_{i,t}),$$

where $\mathtt{rad}_t(i)$ is usually referred to as the *confidence radius* of μ_i. Using the confidence radius we can define $\mathtt{conf}_t(i) = [\widehat{\mu}_{i,t} - \mathtt{rad}_t(i), \widehat{\mu}_{i,t} + \mathtt{rad}_t(i)]$ as the *confidence interval* of μ_i and for simplicity we will denote the upper/lower confidence bounds of $\mathtt{conf}_t(i)$ either as $UCB_{i,t}/LCB_{i,t}$ or $u_{i,t}/l_{i,t}$.

We define the **clean** event as the event $\mathcal{E} := \left\{\forall i \in [K] \forall t \in [T] : |\widehat{\mu}_{i,t} - \mu_i| \leq \mathtt{rad}_t(i)\right\}$ in which all mean values μ_i belong to their respective confidence intervals. Its complementary $\mathcal{E}^C$ is called the **bad event**. The above terms are defined with respect to the value $\mathtt{rad}_t(i)$, which may differ depending on the algorithm. In the case of ϵ-`Greedy` we set $\mathtt{rad}_t(i) = \sqrt{\frac{2\log T}{n_{i,t}}}$, which yields $\mathbb{P}\left[\left|\widehat{\mu}_{i,t} - \mu_i\right| \geq \mathtt{rad}_t(i)\right] \leq 2/T^4$. Thus, taking a union bound over all arms $i \in [K]$ and all rounds $t \in [T]$, the probability of the bad event is:

$$\mathbb{P}[\mathcal{E}^C] \leq 2K/T^3 \leq 2/T^2. \tag{3.3}$$

Combining the above with the probability $\mathbb{P}\left[\widehat{\mu}_{i^*,t} \geq \max_{i \neq i^*} \widehat{\mu}_{i,t}\right]$ that is bounded by $\exp\left(-\frac{t\Delta_i^2}{4K}\right)$ through a similar argument we can prove the following theorem.

Theorem 3.2.1. *Algorithm ϵ-`Greedy` with $\epsilon_t = t^{-1/3}(\log Kt)^{2/3}$ incurs regret*

$$\mathbb{E}[R(t)] = t^{2/3}O(K\log t)^{1/3}.$$

[1]The $\widetilde{O}$ symbolism, eliminates poly-logarithmic dependencies.

3.3 Adaptive Exploration

The $T^{2/3}$ dependence we observed in the regret of non-adaptive algorithms can be improved through **adaptive** exploration. In this section, we will present two fundamental adaptive exploration algorithms: `Successive Elimination` and `UCB`, that achieve regret $\widetilde{O}(\sqrt{T})$.

3.3.1 Successive Elimination Algorithm

`Successive Elimination` Algorithm improves exploration through eliminating arms that are *proven* to be worse than others, depending on their confidence intervals. The correctness of said elimination lies upon the definition of the clean event. The version described below can be found in Slivkins [5] and it is a variant of the algorithm proposed in Even-Dar et al. [2002], which eliminated one arm at a time - the one with the minimum empirical mean reward. The active set technique is also used in a similar manner in Auer and Ortner [2010].

For the `Successive Elimination` Algorithm, we will be using the same confidence radius as in ϵ-`Greedy`, namely $\texttt{rad}_t(i) = \sqrt{2\log T / n_{i,t}}$.

Algorithm 8: `Successive Elimination`

```
1  Input: K, T
   /* Initialize active set to [K]                    */
2  S ← [K]
3  while t ≤ T do
       /* Play every active arm once                  */
4      for i ∈ S do
5          Pull arm i
6          Update u_i, l_i
       /* Deactivation policy                         */
7      for i ∈ S do
8          if ∃j ∈ S such that l_j > u_i then
9              S ← S \ i
```

Algorithm 8 operates in phases: in each phase there is exactly one pull of all active arms. The algorithm keeps track of arms' confidence intervals and as soon as a phase terminates and the intervals of a pair of arms cease to overlap, the arm with the lower $\widehat{\mu}$ gets eliminated. Assuming that the clean event holds, the elimination rule never eliminates the optimal arm.

Theorem 3.3.1. *`Successive Elimination` Algorithm incurs regret*

$$\mathbb{E}[R(t)] = O(\sqrt{Kt\log T}),$$

in each round $t \leq T$.

Proof. Consider the clean event. We begin with a simple observation. If an arm i is active until round t, then $u_{i,t} \geq l_{i^*,t}$; otherwise putting $j = i^*$ in line 7 of

Algorithm 8, arm i would have been eliminated. Thus, using:

$$\left.\begin{array}{rcccl} l_{i^*,t} & = & \widehat{\mu}_{i^*,t} - \texttt{rad}_t(i^*) & \geq & \mu^* - 2\texttt{rad}_t(i^*) \\ u_{i,t} & = & \widehat{\mu}_{i,t} + \texttt{rad}_t(i) & \leq & \mu_i + 2\texttt{rad}_t(i) \end{array}\right\}$$

we acquire the following bound on $\Delta_i \leq 2(\texttt{rad}_t(i) + \texttt{rad}_t(i^*)) \leq 4\texttt{rad}_t(i^*)$. Let t be the last round of the last phase in which arm i is active, then $n_{i,t} = n_{i^*,t}$. Hence:

$$\begin{aligned} \Delta_i &\leq 4\texttt{rad}_t(i) \leq 4\sqrt{\frac{2\log T}{n_{i,t}}} && (n_{i,T} = n_{i,t} + 1) \\ &= O\left(\sqrt{\frac{2\log T}{n_{i,T}}}\right). \end{aligned}$$

Using the regret expression of Equation (3.2) we get:

$$\begin{aligned} R(t) = \sum_{i\in[K]} n_{i,T}\Delta_i &\leq \sum_{i\in[K]} n_{i,T}\left(\sqrt{\frac{2\log T}{n_{i,T}}}\right) \\ &\leq O(\sqrt{\log T}) \sum_{i\in[K]} \sqrt{n_{i,T}}. \end{aligned} \tag{3.4}$$

Applying Jensen's Inequality (A.1.3) on $\sum_{i\in[K]} \sqrt{n_{i,T}}$, using the concavity of the function $f(x) = \sqrt{x}$ we obtain the following bound:

$$\frac{\sum_{i\in[K]} \sqrt{n_{i,T}}}{K} \leq \sqrt{\frac{\sum_{i\in[K]} n_{i,T}}{K}} = \sqrt{\frac{t}{K}}.$$

Thus, Equation (3.4) yields $R(t) \leq O\left(\sqrt{Kt\log T}\right)$. □

3.3.2 Upper Confidence Bound Algorithm (UCB)

The most common phrase that collocates with `Upper Confidence Bound` Algorithm `(UCB)` is "**Optimism under Uncertainty**". The invariant behind `UCB` is to always pick the arm with the highest upper confidence bound, trusting that the optimistic bound (as it is higher than the empirical mean) is indicative of the arm's true mean reward. Intuitively, this measure lets the exploration-exploitation trade-off balance itself out by evaluating the sum of

$$UCB_{i,t} = \underbrace{\widehat{\mu}_{i,t}}_{\text{exploitation factor}} + \underbrace{\texttt{rad}_t(i).}_{\text{exploration factor}}$$

Thus, both arms having a high *exploitation* factor (high empirical mean) and arms having a high *exploration* factor (large confidence bounds due to fewer number of pulls) contest for the learner's choice. The optimism notion was introduced by Lai and Robbins [1985]. After this, a line of work ([49], [50], [51], [7], [3]) studied multiple variants of algorithms following the optimism principle, with the main difference being the choice of the confidence radius. The algorithm presented below is very similar to `UCB1` introduced by Auer et al. [2002a] and is gracefully simple (we will be referring to `UCB1` as `UCB`). The confidence radius is $\texttt{rad}_t(i) = \sqrt{2\log\frac{1}{\delta}/n_{i,t}}$, where $\delta = f(t)$. When computing regret we will set $\delta = 1/T^2$.

Algorithm 9: `UCB`

Input: number of arms K, time horizon T, δ
for $t \in [T]$ **do**
  $A_t \leftarrow \arg\max_{i \in [K]}(u_{i,t})$ // $u_{i,t} = \widehat{\mu}_{i,t} + \texttt{rad}_t(i)$

The regret analysis of `UCB` can be expressed very similarly to that of Theorem 3.3.1, with a more careful argument to compare $n_{i,t}$ with $n_{i^*,t}$. For completeness, we provide below a different, yet widely used, analysis that bounds $\mathbb{E}[n_{i,t}]$ instead of Δ_i to achieve the same result.

Theorem 3.3.2. *`UCB` Algorithm incurs regret*

$$\mathbb{E}[R(t)] = O(\sqrt{Kt \log T}),$$

in each round $t \leq T$.

Proof. We begin with defining a slightly different *clean* event w.r.t. an arm i, so as to bound the probability of the algorithm choosing a sub-optimal arm.

$$\mathcal{E}_i = \left\{ \mu^* < \min_{t \in [T]} u_{i^*,t} \right\} \cap \left\{ \widehat{\mu}_{i,N_i} + \sqrt{\frac{2\log(1/\delta)}{N_i}} < \mu^* \right\}.$$

Under the clean event we know that the optimal arm is not underestimated in any round t and arm's i upper confidence bound is below μ^* (which is below $u_{i^*,t}$) after N_i rounds. The value of N_i will be determined after computation to achieve the desired bounds. In the same time, it is obvious that **arm i cannot be played more than N_i times**, since assuming the opposite would mean that there exists a round $t > N_i$ such that $n_{i,t} = N_i$. Then:

$$\begin{aligned} u_{i,t} &= \widehat{\mu}_{i,t} + \sqrt{\frac{2\log(1/\delta)}{n_{i,t}}} \\ &= \widehat{\mu}_{i,N_i} + \sqrt{\frac{2\log(1/\delta)}{N_i}} && (n_{i,t} = N_i) \\ &< \mu^* < u_{i^*,t}, && (\text{definition of } \mathcal{E}_i) \end{aligned}$$

which contradicts the assumption.

Using the law of total probability we can express $\mathbb{E}[n_{i,t}]$ as:

$$\begin{aligned} \mathbb{E}[n_{i,t}] &= \mathbb{E}\left[n_{i,t} \middle| \mathcal{E}_i\right] + \mathbb{E}\left[n_{i,t} \middle| \mathcal{E}_i^C\right] \\ &\leq N_i + T \cdot \mathbb{P}[\mathcal{E}_i^C]. \end{aligned} \tag{3.5}$$

Now the probability of the *bad* event can be bounded by:

$$\mathbb{P}\left[\mathcal{E}_i^C\right] = \mathbb{P}\left[\left\{\mu^* \geq \min_{t\in[T]} u_{i^*,t}\right\} \cup \left\{\widehat{\mu}_{i,N_i} + \sqrt{\frac{2\log(1/\delta)}{N_i}} \geq \mu^*\right\}\right]$$
$$\leq \underbrace{\mathbb{P}\left[\mu^* \geq \min_{t\in[T]} u_{i^*,t}\right]}_{P_1} + \underbrace{\mathbb{P}\left[\widehat{\mu}_{i,N_i} + \sqrt{\frac{2\log(1/\delta)}{N_i}} \geq \mu^*\right]}_{P_2}, \tag{3.6}$$

where $P_1 < t\delta$ through a Hoeffding inequality (A.1.1) application and a union bound. Bounding P_2 requires a choice of N_i that caters to the desired result. Assume that N_i is chosen so that:

$$\Delta_i - \sqrt{\frac{2\log(1/\delta)}{N_i}} \geq c\Delta_i, \tag{3.7}$$

where $c \in (0,1)$ is a constant to be tuned. Then P_2 can be written as:

$$\begin{aligned}
P_2 &= \mathbb{P}\left[\widehat{\mu}_{i,N_i} - \mu_i \geq \Delta_i - \sqrt{\frac{2\log(1/\delta)}{N_i}}\right] && \text{(definition of } \Delta_i\text{)} \\
&\leq \mathbb{P}[\widehat{\mu}_{i,N_i} - \mu_i \geq c\Delta_i] && \text{(Equation (3.7))} \\
&\leq \exp\left(-\frac{N_i c^2 \Delta_i^2}{2}\right). && \text{(Hoeffding inequality (A.1.1))}
\end{aligned}$$

Using these bounds, Equation (3.6) incurs:

$$\mathbb{P}\left[\mathcal{E}_i^C\right] \leq t\delta + \exp\left(-\frac{N_i c^2 \Delta_i^2}{2}\right). \tag{3.8}$$

The last thing to get our result is setting the value of N_i and c. Plugging $N_i = \left\lceil \frac{2\log(1/\delta)}{(1-c)^2\Delta_i^2} \right\rceil$, (the minimum value of N_i that satisfies 3.7) and $c = 1/2$ into Equation (3.8), and using the result in Equation (3.5), we obtain:

$$\mathbb{E}[n_{i,t}] \leq 3 + \frac{16\log T}{\Delta_i^2}.$$

Using the regret expression of 3.2 we end up with the following bound on regret:

$$\begin{aligned}
\mathbb{E}[R(t)] &\leq \sum_{i\in[K]} \mathbb{E}[n_{i,t}]\Delta_i \\
&\leq \sum_{i\in[K]:\Delta_i<\Delta} \mathbb{E}[n_{i,t}] \cdot \Delta + \sum_{i\in[K]:\Delta_i\geq\Delta} 3\Delta_i + \frac{16\log T}{\Delta_i} \\
&\leq T\cdot\Delta + \sum_{i\in[K]} \frac{16\log T}{\Delta} + \sum_{i\in[K]} 3\Delta_i \\
&\leq 8\sqrt{KT\log T} + 3\sum_{i\in[K]} \Delta_i. && (\Delta = \sqrt{16K\log T/T})
\end{aligned}$$

□

3.4 Unknown time horizon T

In all algorithms presented in sections 3.2, 3.3 we assumed that the time horizon T is a known parameter to the learner. Algorithms that do not take the time horizon as a parameter are called *anytime* algorithms. The work of Besson and Kaufmann [52] provides a theoretical guarantee to why such an assumption does not alter the results in the case of unknown T but up to a constant factor.

The authors formally define the `Doubling Trick`, a meta-algorithm that takes as in input an algorithm $\mathcal{A}$ and a time sequence $(T_i)_{i\in\mathbb{N}}$ and runs memory-less copies of it in epochs of (increasing) length $T_{\text{epoch i}} = T_i - T_{i-1}$. This technique was first introduced by Auer et al. [53]. The `Doubling Trick` is presented below as it was defined in [52].

Algorithm 10: `Doubling Trick`

Input: algorithm $\mathcal{A}$, doubling sequence $(T_i)_{i\in\mathbb{N}}$
Let $i \leftarrow 0$, and initialize algorithm $\mathcal{A}^{(0)} \leftarrow \mathcal{A}_{T_0}$
for $t \in [T-1]$ **do**
  if $t > T_i$ **then**
    `/* Full restart */`
    $i \leftarrow i+1$
    $\mathcal{A}^{(i)} \leftarrow \mathcal{A}_{T_i - T_{i-1}}$
  `/* Play algorithm` $\mathcal{A}^{(i)}$ `*/`
  $A_t \leftarrow \mathcal{A}^{(i)}(t - T_i)$

Theorem 3.4.1. *If an algorithm $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \leq cT^\gamma(\log T)^\delta + f(T)$, for $0 < \gamma < 1, \delta \geq 0$ and for $c > 0$, and an increasing function $f(t) = o\left(t^\gamma(\log t)^\delta\right)$ (at $t \to \infty$), then the anytime version $\mathcal{A}' := \texttt{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the geometric sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*, b > 1$ (i.e., $T_i = \lfloor T_0 b^i \rfloor$) with the condition $T_0(b-1) > 1$ if $\delta > 0$, satisfies,*

$$R_T(\mathcal{A}') \leq \ell(\gamma, \delta, T_0, b) c T^\gamma (\log T)^\delta + g(T),$$

with an increasing function $g(t) = o\left(t^\gamma(\log t)^\delta\right)$, and a constant loss $\ell(\gamma, \delta, T_0, b) > 1$,

$$\ell(\gamma, \delta, T_0, b) := \left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \times \frac{b^\gamma(b-1)^\gamma}{b^\gamma - 1}.$$

The reader is referred to the work [52] for the proof of the theorem above. The authors provide similar results for an upper bound on `DT` with exponential horizons and tight lower bounds for both cases. For fixed γ, δ, algorithm $\mathcal{A}'$ suffers constant regret with respect to $\mathcal{A}_T$.

3.5 Lower Bound

In this section we present the lower bound on the regret of any bandit algorithm. The proof is from Auer et al. [15] and the version below is from Slivkins [5].

Theorem 3.5.1. *Fix time horizon T and the number of arms K. For any bandit algorithm, there exists a problem instance such that $\mathbb{E}[R(T)] \geq \Omega(\sqrt{KT})$.*

KL-divergence

Before proceeding with the proof, we give a brief overview of the **Kullback-Leibler** or **KL-divergence**, a tool from information theory that is particularly useful in our proof. It is defined as

$$KL(p,q) = \sum_{x \in \Omega} p(x) \ln \frac{p(x)}{q(x)} = \mathbb{E}_p \left[\ln \frac{p(x)}{q(x)} \right],$$

where p, q are probability distributions defined on the sample space Ω. KL divergence is used to compute the statistical distance between the distributions p, q and is also referred to as **relative entropy**, denoted $D_{KL}(p||q)$.

Applying the standard properties of KL-divergence (Theorem A.1.4) we obtain the following result.

Lemma 3.5.2. *Consider sample space $\Omega = \{0,1\}^n$ and two distributions on Ω, $p = RC_\epsilon^n$ and $q = RC_0^n$, for some $\epsilon \in (0, 1/2)$. Then $|p(A) - q(A)| \leq \epsilon\sqrt{n}$ for any event $A \subset \Omega$.*

Another preliminary for the following proof is the definition of **best-arm identification**. Best-arm identification is a variant of the MAB problem where the algorithm makes a prediction on the best arm y_t after each round t. In this setting the objective is not to minimize a regret function but to maximize the probability $\mathbb{P}[y_t = i^*]$, where i^* is the arm with the actual highest mean reward.

The family of instances we will be using for the lower bound is described by Bernoulli distributions $D_i, i \in [K]$ with expected mean given by the following rule.

$$\mathcal{I}_j = \begin{cases} \mu_i = 1/2 + \epsilon/2 & \text{if } i = j \\ \mu_i = 1/2 & \text{if } i \neq j \text{ or } j = 0. \end{cases}$$

In instance $\mathcal{I}_0$ all arms i behave like fair coins RC_0^i; in the rest of the instances $\mathcal{I}_j$, arm j is chosen to be the best arm with a distribution of a biased coin RC_ϵ^j while the distribution of arm $i \neq j$ is RC_0^i. The difference of the mean values is $\Delta_i = \epsilon/2$ for all arms $i \neq j$. On a higher level, what this proof aims to show is that playing each arm i $\frac{\log T}{\Delta_i}$ times is necessary to achieve optimal regret.

Lemma 3.5.3. *Consider a best-arm identification problem with $T \leq \frac{cK}{\epsilon^2}$ for a small enough absolute constant $c > 0$. Fix any deterministic algorithm for this problem. Then there exists at least $\lceil K/3 \rceil$ arms a such that for problem instances $\mathcal{I}_a$ we have*

$$\mathbb{P}[y_T = a | \mathcal{I}_a] < 3/4.$$

Proof. We define the tuple $(r_t(a) : a \in [K], t \in [T])$ where $r_t(a)$ is the observed reward for the t-th time the algorithm chooses arm a. The tuple $(r_s(a))_{s \in [t]}$ belongs in the sample space $\Omega_a^t = \{0,1\}^t$. The complete sample space for the algorithm is the product $\Omega = \prod_{a \in [K]} \Omega_a^T$. Conditioning on an instance $\mathcal{I}_j$ we define the following distribution on Ω:

$$P_j(A) = \mathbb{P}[A \mid \mathcal{I}_j] \text{ for each} A \subset \Omega. \tag{3.9}$$

The above describes the probabilities of different realizations of rewards, given an instance $\mathcal{I}_j$.

Assuming that instance $\mathcal{I}_0$ holds and using simple contradiction arguments, we can make the following observations:

1. There are more than $2K/3$ arms j such that $\mathbb{E}_0[n_{j,T}] \leq 3T/K$,
2. There are more than $2K/3$ arms j such that $\mathbb{P}_0[y_T = j] \leq 3/K$.

The first observation is upon the total number of pulls of certain arms and the second upon their chances to be predicted as best arms. Applying a Markov Inequality (A.1.5) on the first observation we obtain $\mathbb{P}[n_{j,T} \leq 24T/K] \geq 7/8$.

As the number of arms cannot extend K, there are at least $K/3$ arms for which both observations hold. We will prove that for an arm j that satisfies the above

$$P_j[y_T = j] \leq 1/2.$$

As $\mathbb{P}[n_{j,T} \leq 24T/K] \geq 7/8$ we restrict the sample space to

$$\Omega^* = \Omega_j^m \times \prod_{a \neq j} \Omega_a^T,$$

where $m = \min(T, 24T/K)$, thus in Ω^* arm j is pulled $24T/K$ times. We define $\mathbb{P}_j^*$ on Ω^* in the same way that we defined $\mathbb{P}_j$ on Ω. Using Lemma 3.5.2 we obtain

$$2\left|P_0^*(A) - P_j^*(A)\right| \leq \epsilon\sqrt{m}.$$

Plugging $T \leq \frac{cK}{\epsilon^2}$ above and tuning constant c the bound becomes

$$|P_0^*(A) - P_j^*(A)| \leq 1/8, \text{for all events } A \in \Omega^*. \tag{3.10}$$

As event $\{y_T = j\}$ may not be a subset of Ω^* (due to the need of more pulls of arm j), we define two slightly differentiated events:

$$A = \{y_T = j \text{ and } n_{j,T} \leq m\} \text{ and } A' = \{n_{j,T} > m\}.$$

From Equation (3.10) we have

$$\begin{aligned} P_j^*(A) &\leq \frac{1}{8} + P_0^*(A) \\ &\leq \frac{1}{8} + P_0^*[y_T = j] \\ &\leq \frac{1}{4}. \qquad (P_0^*[y_T = j] \leq 3T/K \text{ and } T \leq cK/\epsilon^2.) \end{aligned}$$

Similarly we can prove that $P_j^*(A') \leq 1/4$. Combining the above we obtain that $P_j[y_T = j] \leq 1/2$ which concludes the proof. □

The following corollary is proven trivially using Lemma 3.5.3.

Corollary 3.5.4. *Assume T as in Lemma 3.5.3.Fix any algorithm for best-arm identification. Choose an arm a uniformly at random, and run the algorithm on instance $\mathcal{I}_a$. Then $\mathbb{P}[y_T \neq a] \geq 1/12$, where the probability is over the choice of arm a, the randomness in rewards and the algorithm.*

Proof. [Theorem 3.5.1] Fix the parameter $\epsilon > 0$ and consider a random instance $\mathcal{I}_a$. Assume that $T \leq \frac{cK}{\epsilon^2}$, as in Lemma 3.5.3.

Given a round t we can apply the results on best-arm identification, through considering the algorithm's prediction y_t to be the arm A_t pulled in round t. Using Corollary 3.5.4 we obtain that $\mathbb{P}[y_t \neq a] \geq 1/12$. Taking expectation on Δ_{A_t} - namely computing how much regret is accumulated in each round in expectation - we get that

$$\begin{aligned}
\mathbb{E}[\Delta_{A_t}] &= \mathbb{P}[y_t \neq a] \cdot \Delta + \mathbb{P}[y_t = a] \cdot 0 \\
&\geq \frac{1}{12} \cdot \frac{\epsilon}{2} && (\Delta = \Delta_i = \tfrac{\epsilon}{2} \text{ for all } i \in [K]) \\
&\geq \frac{\epsilon}{24}
\end{aligned}$$

The expected regret can be expressed as

$$\mathbb{E}[R(T)] = \sum_{t=1}^{T} \mathbb{E}[\Delta_{A_t}] \geq \frac{\epsilon T}{24}.$$

Tuning $\epsilon = \sqrt{\frac{cK}{T}}$; i.e. the largest value for which $T \leq \frac{cK}{\epsilon^2}$ yields the result. □

Chapter 4

Fairness notions in Stochastic MAB

The Multi-Armed Bandit framework is used in many algorithmic decision-making applications. Thus, there are natural constraints stemming from the environment of each application. One such constraint is the **fairness** constraint that is the main subject of this thesis. Applications that ask for fairness guarantees involve targeted advertising, clinical trials, admission/lending processes, decisions about bail and/or sentencing and many more.

Recent work (Ferrara [8]) has uncovered biases in decisions made by learning algorithms that are caused by the machine learning pipeline and not the designer's will. Such biases include:

- **Sampling/Representation Bias** Bias that occurs when the population to be modeled is not accurately represented in the data provided. This can lead to poor performance on underrepresented individuals or groups.
- **Algorithmic Bias** Bias that occurs when an algorithm makes a decision prioritizing protected attributes such as gender, age, economic status and more.
- **Confirmation Bias** Bias that occurs when decisions are made based on previous ones made by humans and thus reflect biases present in the decision makers' behaviour.

Examples of the above have been documented in various works. Angwin et al. [9] investigated the accuracy of the recidivism algorithm used by Correctional Offender Management Profiling for Alternative Sanctions (COMPAS), a system to predict the risk of recidivism among defendants. Among other results, they showed that black defendants were more likely to be labeled as higher risk compared to their white counterparts conditioning on defendants that did not recidivate over a two-year period, whereas white defendants were more likely to be labeled as low risk compared to their black counterparts, conditioned on defendants that re-offended within the next two years. In the targeted advertising field, Lambrecht and Tucker

[10] did an empirical-quantitative field test among 191 countries showing that STEM job advertisements were more likely to be displayed to male audience by multiple platforms, due to cost-effectiveness. Köchling and Wehner [54] provided in depth analysis of decision-making in the context of HR recruitment and Obermeyer et al. [55] study algorithmic bias appearing in health classification algorithms that result in racial bias against Black patients.

Moreover, applying fairness constraints on decision-making algorithms improves the users' trust on the platform. Results from Claure et al. [18] studying a resource distribution application with fairness constraints in human-robot teams show that even though the constraint imposed that teammates with better performance scores were chosen less often, the median score of the team was improved. The work of Jaillet et al. [56] links a fairness constraint on a revenue management application with higher user satisfaction of the platform.

Multiple definitions of fairness have been proposed and they can be classified in the following main categories:

- **Individual Fairness** Notions of this group follow the principle "Similar individuals should be treated similarly" (Dwork et al. [57]) and are defined with respect to a closeness criterion.
- **Group Fairness** Such notions ensure that different groups are treated equally or proportionally. Sub-classes of group fairness include demographic parity, equality of opportunities and equality of odds, and groups are usually defined with respect to one or more protected attributes such as age, race, gender. Group fairness does not guarantee individual fairness inside a specific group.

4.1 Definitions of Fairness

4.1.1 Fairness of Exposure

The definition of **Fairness of Exposure** from Wang et al. [16] is a very natural meritocratic notion of fairness. They borrow ideas from [58] and generalize their results to arms with arbitrary reward distributions and merit functions. Wang et al. [16] propose a *policy* π^* under which the amount of exposure given to each arm is *proportional* to its merit, quantified through an application-dependent merit function $f(\cdot) > 0$.

$$\frac{\pi^*(a)}{f(\mu_a)} = \frac{\pi^*(a')}{f(\mu_{a'})}, \forall a, a' \in [K].$$

As policy π^* can only be learned through exploration, no algorithm without prior information can follow π^* in early rounds. Thus, this definition of fairness is not in the form of a fairness constraint that should be satisfied in any round $t \in [T]$. To quantify an algorithm's fairness of exposure, the **fairness regret** $FR(T)$ is introduced, using policy π^* as a benchmark.

$$FR(T) = \sum_{t\in[T]} \sum_{a\in[K]} \left|\pi^*(a) - \pi_t(a)\right|.$$

In the same time, the reward regret is also computed relatively to π^*.

$$RR(T) = \sum_{t\in[T]} \sum_{a\in[K]} \pi^*(a)\mu_a - \sum_{t\in[T]} \sum_{a\in[K]} \pi_t(a)\mu_a.$$

In their work, Wang et al. [16] develop two algorithms: `FairX-UCB, FairX-TS` variants of the `UCB` algorithm (3.3.2) and `Thompson Sampling (TS)` algorithm ([59]) and study their fairness and reward regret, given the *FairX* setting that imposes certain conditions on the merit functions[1]. With careful tuning of the `FairX-UCB` parameters, they show that with probability at least $1-\delta$

$$FR_{FairX-UCB}(T) = \widetilde{O}\left(\frac{L\sqrt{KT}}{\gamma}\right), \text{ and}$$

$$RR_{FairX-UCB}(T) = \widetilde{O}\left(\sqrt{KT}\right).$$

4.1.2 Fairness through maximizing Nash social welfare

The model studied in Hossain et al. [60] is an extension of the standard MAB, where in each round t, N agents express their individual reward of arm I_t. The motivation behind the model is that, making an algorithmic social choice which affects groups in a different manner is prone to develop *the tyranny of the majority* dynamic [61].

Fix a round t. Let $p = (p_j)_{j\in[K]}$ be the learner's policy and $\mu^*_{i,j}$ be the expected reward of arm i for agent $i \in [N]$. The expected utility of policy p for agent i is $\sum_j p_j \cdot \mu^*_{i,j}$. Maximizing Nash social welfare asks for the maximization of the product of the utilities of all agents. In formal form:

$$\max_p \text{NSW}(p, \mu^*) = \max_p \prod_{i\in[N]} \left(\sum_{j\in[K]} p_j \cdot \mu^*_{i,j}\right).$$

As in the case of fairness of exposure, the learner is asked to compete with an optimal policy p^* and not with the best fixed arm. Thus, the form of regret is defined with respect to the NSW of the optimal policy.

$$\mathbb{E}[R^T] = \sum_{t\in[T]} \max_p \text{NSW}(p, \mu^*) - \sum_{t\in[T]} \text{NSW}(p^t, \mu^*),$$

where p^t is the learner's policy in round t.

In their work, they propose slightly different variations of `Explore-First, ϵ-Greedy, UCB` algorithms that achieve sub-linear NSW regret. We briefly present the respective bounds.

[1]The first condition needs $f(\mu_a) \geq \gamma > 0$ for all $a \in [K]$ and the second one asks for the merit function f to be L-Lipschitz continuous for some constant $L > 0$.

Algorithm	Regret	Parameters
`Explore-First`	$\widetilde{O}\left(N^{\frac{2}{3}}K^{\frac{1}{3}}T^{\frac{2}{3}}\right)$	$L=\tilde{\Theta}\left(N^{\frac{2}{3}}K^{-\frac{2}{3}}T^{\frac{2}{3}}\right)$
	$\widetilde{O}\left(N^{\frac{1}{3}}K^{\frac{2}{3}}T^{\frac{2}{3}}\right)$	$L=\tilde{\Theta}\left(N^{\frac{1}{3}}K^{-\frac{1}{3}}T^{\frac{2}{3}}\right)$
ϵ-`Greedy`	$\widetilde{O}\left(N^{\frac{2}{3}}K^{\frac{1}{3}}T^{\frac{2}{3}}\right)$	$\epsilon_t=\tilde{\Theta}\left(N^{\frac{2}{3}}K^{\frac{1}{3}}t^{-\frac{1}{3}}\right)$
	$\widetilde{O}\left(N^{\frac{1}{3}}K^{\frac{2}{3}}T^{\frac{2}{3}}\right)$	$\epsilon_t=\tilde{\Theta}\left(N^{\frac{1}{3}}K^{\frac{2}{3}}t^{-\frac{1}{3}}\right)$
`UCB`	$O\left(NKT^{\frac{1}{2}}\log(NKT)\right)$	$a_t=N$
	$O\left(N^{\frac{1}{2}}K^{\frac{3}{2}}T^{\frac{1}{2}}\log^{\frac{3}{2}}(NKT)\right)$	$a_t=\sqrt{12NK\log(NKt)}$

Table 4.1: Regret bounds for NSW fair algorithms. For more details on the parameters L, a_t the reader is referred to [60].

4.1.3 Fairness through a minimum pulling rate

The concept of achieving fairness through guaranteeing a minimum pulling rate to each arm has been studied in recent works. The work of Li et al. [17] suggests such a constraint in the Combinatorial Sleeping bandit setting[2], Claure et al. [18] and Chen et al. [19] study resource/task allocation problems through the model of classic and contextual MAB imposing similar constraints and Patil et al. [20] introduce the Fair-MAB Problem and a meta-algorithm called `Fair-Learn` as a framework of the class of Fair-MAB algorithms. Part of the work of [20] will be presented below.

A Fair-MAB instance is described with a tuple $\langle T, [K], (\mu_i)_{i\in[K]}, (r_i)_{i\in[K]}\rangle$, where T is the time horizon, $[K]$ is the set of arms, $\mu_i \in [0,1]$ is the mean reward of arm i, and $r_i \in [0, 1/K)$ is the minimum pulling rate associated with arm i. Naturally it should be that $\sum_{i\in[K]} r_i \leq 1$. The bound $1/K$ is selected because the authors consider guaranteeing a rate larger than the proportional one to be unfair[3]. The reward distributions $\mathcal{D}_i$ are $Bernoulli(\mu_i)$ with the mean value μ_i being unknown to the algorithm (learner). Let $n_{i,t}$ be the total pulls of arm i until round t. Given the model above, they define fairness as follows.

Definition 4.1.1. Given an unfairness tolerance $a \geq 0$, a Fair-MAB algorithm $\mathcal{A}$ is said to be a-fair if $\lfloor r_i t - n_{i,t} \rfloor \leq a$ for all $t \leq T$ and for all arms $i \in [K]$.

The expression $\lfloor r_i t - n_{i,t} \rfloor$ is the difference between the minimum required pulls of arm i and its realized number of pulls and we will be referring to it as *pulling difference*. Setting $a = 0$ and taking expectation on the pulling difference per-round we get the *asymptotic* fairness definition from [17].

Given the a-relaxation, any Fair-MAB optimal algorithm should satisfy the following:

$$\text{If } \lfloor r_i T \rfloor - a > 0 \text{ then } n_{i,t} = \lfloor r_i T \rfloor - a; \text{ else } n_{i,t} = 0, \text{ for all arms } i \neq i^*.$$

[2]This model is a mixture of combinatorial bandits, where multiple arms can be pulled together and form a super-arm (see [62], [63]); and sleeping bandits, where certain arms may be unavailable in a number of rounds (see [64]).

[3]This assumption is not made in other works imposing such constraints, while [18],[19] study the case where a uniform minimum pulling rate v is achieved.

Hence, a fairness-aware r-Regret with respect to the above optimal property is introduced.

$$\mathcal{R}^r_{\mathcal{A}}(T) = \sum_{i\in[K]} \Delta_i \cdot (\mathbb{E}[n_{i,t}] - \max(0, \lfloor r_i T \rfloor - a)) .$$

The main contribution of Patil et al. [20] is the Fair-MAB algorithm `Fair-Learn`. The input of `Fair-Learn` is the unfairness tolerance a and a learning algorithm `Learn` $(\cdot)$. In each round t, the algorithm maintains a set $A(t)$ of arms whose fairness constraint is not satisfied and pulls the arm with the largest pulling difference unless $A(t)$ is empty, in which case algorithm `Learn` picks the next arm.

They prove that `Fair-Learn` is a-fair irrespective of the learning algorithm `Learn` provided as input and conclude their work with some computational results on the r-Regret $\mathcal{R}^r(T)$ and the pseudo regret $\mathcal{R}(T)$ (w.r.t. the policy selecting arm i^* in each round t) of the `Fair-Learn` when `UCB` is given as a learning algorithm. The aforementioned results are the following.

$$\mathcal{R}^r_{Fair-UCB}(T) \leq \left(1 + \frac{\pi^2}{3}\right) \sum_{i\in[K]} \Delta_i + \sum_{\substack{i\in[K] \\ i\neq i^*}} \Delta_i \cdot \left(\frac{8\ln T}{\Delta_i^2} - (r_i T - a)\right),$$

$$\mathcal{R}_{Fair-UCB}(T) \leq \sum_{i\in S(T)} \Delta \cdot (r_i T - a) + \sum_{\substack{i\in[K] \\ i\neq i^*}} \frac{8\ln T}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i\in[K]} \Delta_i,$$

where $S(T) = \{i | r_i T - a < \frac{8\ln T}{\Delta_i^2}\}$. It is obvious that if $S(T) \neq \emptyset$, then the pseudo-regret of `Fair-UCB` is linear in T.

4.1.4 Related work

Another approach to the minimum pulling rate constraint is *penalized regret*, introduced by Fang et al. [65]. Let $\tau_k \geq 0, k \in [K]$ be the minimum pulling rate associated with arm k with $\sum_{k\in[K]} \tau_k < 1$. They define the following penalized reward

$$S_{pen,\pi}(T) = S_\pi(T) - \sum_{k=1}^{K} A_k \left(\tau_k T - N_{k,\pi}(T)\right)_+ ,$$

where $S_\pi(T)$ is the total reward collected following policy π until round T, A_k is a non-negative penalty rate associated with arm k and $N_{k,\pi}(T)$ is the total number of pulls of arm k until round T following policy π. The pseudo-regret accumulated with respect to the policy π^* that chooses the arm with the highest mean reward μ^* in every round t is

$$\begin{aligned} L(T) &= \mu^* T - \mathbb{E}[S_{pen,\pi}(T)] \\ &= \sum_{k=1}^{K} \left[\Delta_k \mathbb{E}[N_k(T)] + A_k \mathbb{E}[(\tau_k T - N_k(T))_+]\right] . \end{aligned}$$

Contrary to strict non-asymptotic fairness guarantees proposed by other works, in this setting the learner is able to choose whether she cares to fulfill the minimum pulling rate of an arm depending on the respective penalty she receives.

Lastly, we make a brief reference to the work of Killian et al. [66] on Equitable Restless MAB (ERMAB). The RMAB framework is used for decision making where a central entity should decide on an optimal allocation of a limited number of resources (rewards) to a fixed number of arms. It is widely used in public health, treatment scheduling and other sensitive decision making applications. Given the nature of these applications, some type of equity should be established as to ensure optimal social welfare. They study a definition of group fairness that wishes to be optimal over two objectives: maximin reward and maximum Nash welfare. Given a grouping on the K arms the maximin reward (MMR) maximizes a group's minimum expected total reward and guarantees equality of outcomes. Optimizing Nash welfare through the maximization of the product of groups' rewards ensures a balanced allocation. Their work aims to solve the offline problem, where arm models are known.

4.2 δ-Fairness

The definition of **δ-fairness** proposed by Joseph et al. [1] was thoroughly studied for this thesis. Their fairness constraint is related to the notion of individual fairness, they apply it to the classic stochastic MAB model (3.1) and generalize their results to the contextual bandit setting. For the purposes of this thesis, we present the results on the classic stochastic MAB problem.

Using the same notation as in Chapter 3. Let h be the history $H_{t-1} = ((A_1, r_1), \ldots, (A_{t-1}, r_{t-1}))$ and $\pi_t(j|h)$ be the probability that an algorithm $\mathcal{A}$ chooses arm j in round t given a history h.

Definition 4.2.1. [δ-Fairness] An algorithm $\mathcal{A}$ is **δ-fair** if, for all sequences of rewards $r_{A_1}, \ldots, r_{A_t}$ and all payoff distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$ with probability at least $1 - \delta$ over the realization of the history h, for all rounds $t \in [T]$ and all pairs of arms $j, j' \in [K]$,

$$\pi_t(j|h) > \pi_t(j'|h) \text{ only if } \mu_j > \mu_{j'}.$$

For simplicity we will be using the notation $\pi_{j,t}$ instead of $\pi_t(j|h)$. At a higher level, δ-fairness suggests that an algorithm cannot *favor* arm j over arm j' (i.e., pull arm j with higher probability than arm j') until it has gathered enough information to be sure that arm j has a higher mean reward; expect with probability at most δ.

Their work focuses on the pseudo-regret achieved by δ-fair algorithms with respect to the policy π^* that chooses the arm with the highest expected reward. The expression of said reward has been computed in Section 3.1.

$$R(T) = T \cdot \mu^* - \sum_{t=1}^{T} \mu_{A_t} = \sum_{t=1}^{T} (\mu^* - \mu_{A_t}) = \sum_{t=1}^{T} \Delta_{A_t}. \qquad \text{(Equation (3.1))}$$

Joseph et al. prove a lower bound on the rounds T in which any δ-fair algorithm experiences constant per-round regret. Their result is the following theorem.

Theorem 4.2.2. *There is a distribution P over K-arm instances of the stochastic multi-armed bandit problem such that any fair algorithm run on P experiences constant per-round regret for at least*

$$T = \Omega\left(K^3 \ln \frac{1}{\delta}\right)$$

rounds.

They propose the `FairBandits` Algorithm that matches the dependence on K computed in the lower bound. `FairBandits` is a randomized elimination algorithm, similar to the `Successive Elimination` Algorithm (8) with a different elimination rule. Throughout the analysis we will be using the term **linked** for arms whose confidence intervals overlap and the term **chained** for arms who belong in the same component of the transitive closure of the linked relation. In each round t `FairBandits` keeps track of a set S_t of *active* arms; i.e., arms that are *chained* to arm i^*, and pulls an arm from S_t uniformly at random. Any arm not contained in S_t gets eliminated, never to be pulled again. Thus, the cardinality of S_t is decreasing with respect to t. The algorithm is presented below.

Algorithm 11: `FairBandits`

Input: number of arms K, δ.
`/* Initialize active set and statistical information */`
$S_0 \leftarrow \{1, \ldots, K\}$.
for $i \in [K]$ **do**
  $\widehat{\mu}_{i,0} \leftarrow 1/2, u_{i,0} \leftarrow 1, l_{i,0} \leftarrow 0, n_{i,0} \leftarrow 0$
for *round* $t \in [T]$ **do**
  $i_t^* \leftarrow \arg\max_{i \in S_{t-1}} u_{i,t}$
  $S_t \leftarrow \{j | j \text{ chains to } i_t^*, j \in S_{t-1}\}$
  Pull arm $j \in S_t$ uniformly at random
  Observe reward $r_{j,t}$
  `/* Update statistical information for arm` j `*/`
  $n_{j,t} \leftarrow n_{j,t-1} + 1$
  $\widehat{\mu}_{j,t} \leftarrow \frac{1}{n_{j,t}}(\widehat{\mu}_{j,t-1} \cdot n_{j,t-1} + r_{j,t})$
  $\texttt{rad}_t(j) \leftarrow \sqrt{\frac{\ln((\pi \cdot (t+1))^2)/3\delta}{2n_{j,t}}}$
  $[l_{j,t}, u_{j,t}] \leftarrow [\widehat{\mu}_{j,t} - \texttt{rad}_t(j), \widehat{\mu}_{j,t} + \texttt{rad}_t(j)]$
  for $i \in S_t, i \neq j$ **do**
    $\widehat{\mu}_{i,t} \leftarrow \widehat{\mu}_{i,t-1}, u_{i,t} \leftarrow u_{i,t-1}, l_{i,t} \leftarrow l_{i,t-1}, n_{i,t} \leftarrow n_{i,t-1}$

From the choice of the confidence radius we can easily prove the following lemma.

Lemma 4.2.3. *With probability at least $1 - \delta$, for every arm i and round t $l_{i,t} \leq \mu_i \leq u_{i,t}$.*

The proof follows from a standard Hoeffding Inequality (A.1.1) application and a union bound on all arms $i \in [K]$ and all rounds $t \in [T]$. Using Lemma 4.2.3, we are able to show that Algorithm 11 is δ-fair. We refer the reader to Theorem 1 of [1], where she can find the respective proof.

Theorem 4.2.4. *If $\delta < 1/\sqrt{T}$, then* `FairBandits` *has regret*

$$R(T) = O\left(\sqrt{K^3 T \ln \frac{TK}{\delta}}\right).$$

Before proceeding with the main proof, Joseph et al. prove a lower bound on the total number of pulls of an active arm i until round t by applying an additive Chernoff bound on $n_{i,t} = \sum_{t' \leq t} X_{t'}$, where $X_{t'}$ is an indicator random variable of whether arm i was pulled in round t'. The following result is achieved using the fact that $\mathbb{P}[X_{t'}] = \frac{1}{|S_{t'}|} \geq \frac{1}{K}$ for all $t' \leq t$.

Lemma 4.2.5. *With probability at least* $1 - \frac{\delta}{2t^2}$

$$n_{i,t} \geq \frac{t}{K} - \sqrt{\frac{t}{2} \ln\left(\frac{2Kt^2}{\delta}\right)},$$

for all $i \in S_t$.

The bound on the total regret is computed below.

Proof. [Theorem 4.2.4] The regret expression we will be using is

$$R(T) = \sum_{i \in [T]} \Delta_{A_t}, \tag{4.1}$$

where A_t is the arm pulled by the algorithm in round t. It suffices to bound Δ_{I_t} to get our result.

Fix an arm i and a round t in which $i \in S_t$. Then

$$2\mathtt{rad}_t(i) = 2\sqrt{\frac{\ln\left(\pi \cdot (t+1)\right)^2 / 3\delta}{2n_{i,t}}} \leq 2\sqrt{\frac{\ln\left(\pi \cdot (t+1)\right)^2 / 3\delta}{2\left(\frac{t}{K} - \sqrt{\frac{t}{2}\ln(\frac{2Kt^2}{\delta})}\right)}} = \eta(t). \tag{4.2}$$

Arm $i \in S_t$, so it is chained to i^*. Let $C \subseteq [K]$ be the chain of arms between (i, i^*). Applying the definition of the *linked* relation in every link of the chain we end up with the following

$$\begin{aligned} l_i &\geq u_{i^*} - \sum_{j \in C} 2\mathtt{rad}_t(j) \\ &\geq u_{i^*} - \sum_{j \in C} \eta(t) && \text{(Equation (4.2))} \\ &\geq u_{i^*} - K \cdot \eta(t). && (|C| \leq K) \end{aligned}$$

Using Lemma 4.2.3 we get that $\Delta_i \leq K \cdot \eta(t)$.

We define the clean event as

$$\mathcal{E} = \{\forall i \in [K] \forall t \in [T] : \mu_i \in [l_{i,t}, u_{i,t}]\} \cap \left\{\forall i \in [K] \forall t \in [T] : n_{i,t} \geq \frac{t}{K} - \sqrt{\frac{t}{2} \ln\left(\frac{2Kt^2}{\delta}\right)}\right\}.$$

From Lemma 4.2.3, and a union bound on all rounds $t \in [T]$ on Lemma 4.2.5 we get that

$$\mathbb{P}\left[\mathcal{E}^C\right] \leq \left(1 + \frac{\pi}{2}\right)\delta. \tag{4.3}$$

Hence, Equation (4.1) yields

$$\begin{aligned}
\mathbb{E}[R(T)] &= \mathbb{E}[R(T)|\mathcal{E}] \cdot \mathbb{P}[\mathcal{E}] + \mathbb{E}[R(T)|\mathcal{E}^C] \cdot \mathbb{P}[\mathcal{E}^C] \\
&\leq \underbrace{\mathbb{E}[R(T)|\mathcal{E}]}_{R} + \left(1 + \frac{\pi}{2}\right)\delta T, && \text{(Equation (4.3))}
\end{aligned}$$

where

$$\begin{aligned}
R &= \sum_{t\in[T]} \Delta_{I_t} \\
&\leq \sum_{t\in[T]} \min(1, K \cdot \eta(t)) \leq K \sum_{t\in[T]} \min(1, \eta(t)) \\
&= K \cdot O\left(\sum_{t\in[T] \text{ s.t. } t/k > 2\sqrt{t\ln(tK/\delta)}} \sqrt{\frac{\ln(t/\delta)}{\frac{t}{K} - \sqrt{t\ln(tK/\delta)}}} + \sum_{t\in[T] \text{ s.t. } t/k \leq 2\sqrt{t\ln(tK/\delta)}} 1 \right) \\
&= K \cdot \widetilde{O}\left(\int_{t=1}^{T} \sqrt{\frac{\ln(t/\delta)}{\frac{t}{2K}}} + K^2 \ln(K/\delta) \right) \\
&= \widetilde{O}\left(K^{3/2}\sqrt{2T}\sqrt{\ln\frac{KT}{\delta}} + K^3 \ln\frac{K}{\delta} \right) \\
&= \widetilde{O}\left(K^{3/2}\sqrt{T\ln\frac{KT}{\delta}} + K^3 \right).
\end{aligned}$$

Plugging the derivation of R and $\delta \leq \sqrt{\frac{1}{T}}$ into $\mathbb{E}[R(T)]$ we get

$$\mathbb{E}[R(T)] = \widetilde{O}\left(K^{3/2}\sqrt{T\ln\frac{KT}{\delta}} + K^3 \right).$$

□

Chapter 5

(ε, δ)-Fairness

This chapter contains the main contribution of this thesis. Motivated by the δ-*fairness* definition in Joseph et al. [1], we propose the following relaxed definition:

Definition 5.0.1. [(ε, δ)-Fairness] An algorithm $\mathcal{A}$ is (ε, δ)**-fair** if, for all sequences of rewards $r_{A_1}, \ldots, r_{A_t}$ and all payoff distributions $\mathcal{D}_1, \ldots, \mathcal{D}_k$ with probability at least $1 - \delta$ over the realization of the history h, for $\varepsilon > 0$, for all rounds $t \in [T]$ and all pairs of arms $j, j' \in [k]$,

$$\pi_t(j|h) > \pi_t(j'|h) + \varepsilon \text{ only if } \mu_j > \mu'_j$$

where $\pi_t(j|h)$ is the probability that algorithm $\mathcal{A}$ chooses arm j in round t given a history h. For simplicity we will be using the notation $\pi_{j,t}$ for the aforementioned probability.

At a high level, this definition has the same intuition as that of δ-fairness; i.e., a fair algorithm cannot favor any arm j over arm j' unless arm j is has a higher mean value than j', with probability greater than $1 - \delta$. The relaxation appears in the way *favoring an arm* is defined. In the (ε, δ)-fairness setting, favoring an arm over another is defined as playing them with probabilities that are more than ε away from each other. It is obvious that when ε is set to 0, the two definitions are identical, thus any $(0, \delta)$-fair algorithm is also δ-fair. The case where $\varepsilon = 1$ allows the learner to choose her policy without any fairness constraint.

5.1 FT Algorithm

We proceed with presenting `Fair-Truthful (FT)` Algorithm, a simple (ε, δ)-fair, no-regret generalization of the `FairBandits`(δ) algorithm (11). Using the notion of two arms being *linked* or *chained* from Section 4.2, together with the ε relaxation, `FT` Algorithm incurs a gracefully optimized regret, replacing the $\sqrt{K^3}$ dependence on K with $\min\left\{\sqrt{\frac{K}{\varepsilon}}, \sqrt{K^3}\right\}$, that significantly affects the result for $\varepsilon > 1/K^2$. The role of ε is captured in Figure 5.1.

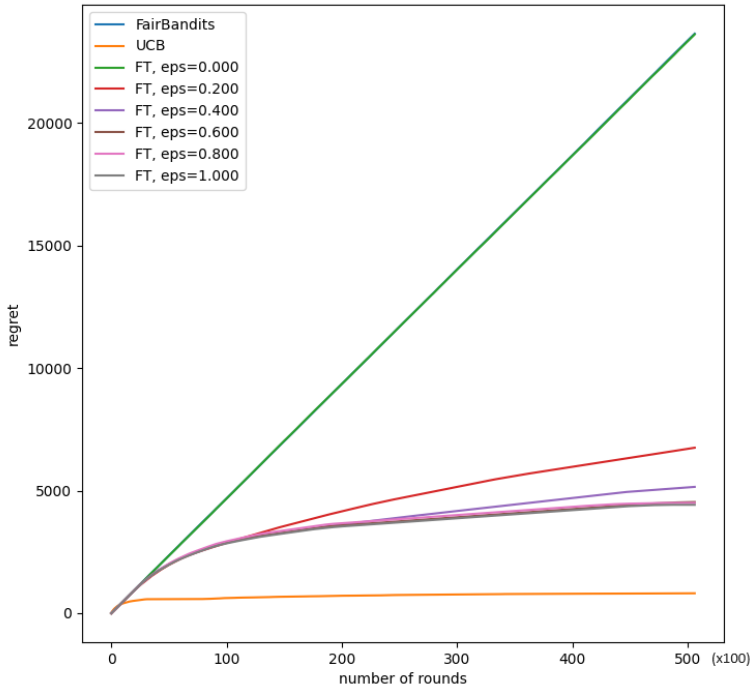

Figure 5.1: Regret plot for instance with $K = 15$, $T = K^4$.

Algorithm 12: `Fair algorithm for Truthful agents` (FT)

Input: number of arms K, ε, δ.

`/* Initialize active set and statistical information */`

$S_0 \leftarrow \{1, \ldots, K\}$.

for $i \in [K]$ **do**

  $\widehat{\mu}_{i,0} \leftarrow 1/2, u_{i,0} \leftarrow 1, l_{i,0} \leftarrow 0, n_{i,0} \leftarrow 0$

for *round* $t \in [T]$ **do**

  `/* compute the probability distribution` π_t `over active arms. */`

  $\pi_t \leftarrow$ `Grouping`(S_{t-1}, ε) (13)

  Sample an arm $j \sim \pi_t$.

  Observe the reward $r_{j,t}$.

  `/* Update statistical information for arm` j `*/`

  $n_{j,t} \leftarrow n_{j,t-1} + 1$

  $\widehat{\mu}_{j,t} \leftarrow \frac{1}{n_{j,t}}(\widehat{\mu}_{j,t-1} \cdot n_{j,t-1} + r_{j,t})$

  $\texttt{rad}_t(j) \leftarrow \sqrt{\frac{\ln((\pi \cdot (t+1))^2)/3\delta}{2n_{j,t}}}$

  $[l_{j,t}, u_{j,t}] \leftarrow [\widehat{\mu}_{j,t} - \texttt{rad}_t(j), \widehat{\mu}_{j,t} + \texttt{rad}_t(j)]$

  $i_t^\star \leftarrow \arg\max_{i \in S_{t-1}} u_{i,t}$

  $S_t \leftarrow \{i | i \text{ chains to } i_t^\star\}$

  for $i \in S_t, i \neq j$ **do**

    $\widehat{\mu}_{i,t} \leftarrow \widehat{\mu}_{i,t-1}, u_{i,t} \leftarrow u_{i,t-1}, l_{i,t} \leftarrow l_{i,t-1}, n_{i,t} \leftarrow n_{i,t-1}$

The novelty of our work relies on the `Grouping` sub-procedure described below. Groups are formed sequentially using a descending order on the arms' upper confidence bounds. Each group has a pivot arm, that is the arm with the highest UCB of the arms not yet assigned to a group. If an arm is placed in a group it must be **linked** to either the group's pivot arm or any arm in the last group formed.

Algorithm 13: Grouping

Input: Set of active arms S_t, ε.

`/* Initialize number of groups and set of non-assigned arms */`

$M \leftarrow 0$

NA $\leftarrow S_t$

`/* Divide active arms into groups. */`

while $NA \neq \emptyset$ **do**

  Update group counter M: $M \leftarrow M + 1$

  Pivot arm for group M: $j^\star \leftarrow \arg\max_{i \in \text{NA}} u_{i,t}$.

  Assign arms linked to $j^\star$ to group M:

  $G_{M,t} \leftarrow \{i \in \text{NA} : u_{i,t} \geq l_{j^\star,t} \text{ or } u_{i,t} \geq l_{k,t}, k \in G_{M-1,t}\}$.

  Update set of active arms: NA $\leftarrow$ NA $\setminus G_{M,t}$.

Solve the following LP: `// Compute the distribution`

$$\begin{aligned} \text{maximize} \quad & \tilde{\pi}_{1,t} \\ \text{subject to} \quad & \sum_{i=1}^{M} |G_{i,t}| \tilde{\pi}_{i,t} = 1, \\ & \tilde{\pi}_{i,t} \leq \tilde{\pi}_{i+1,t} + \varepsilon, \quad i = 1, \ldots, M-1 \\ & \tilde{\pi}_{i,t} \geq 0 \qquad\qquad\quad i = 1, \ldots, M \end{aligned}$$

Return: Distribution over arms π_t: $\{\pi_{i,t} \leftarrow \tilde{\pi}_{j,t} \text{ s.t. } i \in G_{j,t}, \forall j \in [M]\}$

For the rest of the analysis we will be referring to the parameter n of group $G_{n,t}$ as the **rank** of group $G_{n,t}$. Some natural observations on the `Grouping` sub-procedure are that:

- Given a round t, there exists only one group of each rank.
- The arm with the highest upper confidence bound is always placed in group of rank 1.
- An empty group cannot exist.
- Given a round t the lower the rank of group $G_{n,t}$, the greater the probability $\tilde{\pi}_{n,t}$ of its arms being pulled due to the constraints of the LP.

Since the confidence radius is defined in the same way as in Algorithm 11, Lemma 4.2.3 follows in the same way as in Section 4.2.

Theorem 5.1.1. *`FT` Algorithm is (ε, δ)-fair.*

Lemma 5.1.2. *Fix a round t and an arm i in group $G_{I,t}$. Arm i is un-linked from all arms in groups with rank $r < I - 1$.*

Proof. Assume there exists an arm j in group $G_{J,t}$ with $J < I - 1$ that is linked to arm i. This yields $u_{i,t} \geq l_{j,t}$. So, in the $(J+1)$-th iteration of Algorithm 13 arm i

is still in NA and it would be assigned to group of rank $J+1 \neq I$ which contradicts the assumption. □

Proof. [Theorem 5.1.1] Fix a pair of arms i, j such that $\tilde{\pi}_{i,t} > \tilde{\pi}_{j,t} + \varepsilon$. Then for the groups $G_{I,t}, G_{J,t}$ containing arms i, j respectively, it must be $I < J - 1$; if not the corresponding LP constraint would be violated. Then, from Lemma 5.1.2 arms i, j are **un-linked** from each other.

Lemma 4.2.3 states that with probability greater that $1 - \delta$, for every arm i and every round t: $l_{i,t} \leq \mu_i \leq u_{i,t}$. Thus for arms i, j: $\mu_i > l_{i,t} > u_{j,t} > \mu_j$ with probability at least $1 - \delta$. □

5.1.1 Regret Analysis

In this section we compute the regret achieved by Algorithm 12.

Theorem 5.1.3. *Algorithm 12 incurs regret:*

$$R(T) = \mathcal{O}\left(\min\left(\sqrt{\frac{1}{\varepsilon}}, K\right)\sqrt{KT\log(KT/\delta)}\right) \tag{5.1}$$

Before we proceed with the main proof we prove four auxiliary lemmas. Lemma 5.1.4 upper bounds the maximum number of groups with non-zero probability, Lemma 5.1.7 upper bounds the number of pulls of a sub-optimal arm and Lemmas 5.1.5, 5.1.6 support the proof of the latter.

Lemma 5.1.4. *For any round $t \in [T]$, the number of groups with non-zero probability $\tilde{\pi}_{\cdot,t}$ is upper bounded by $m = O(\min\{\sqrt{1/\varepsilon}, K\})$.*

Proof. Fix a round $t \in [T]$ and assume that the active arms are divided into $M > 0$ groups. Our goal is to find the maximum number of groups with $\tilde{\pi}_{i,t} > 0$, where $i \in [M]$. Let $G_{m,t}$ be the last group with non-zero probability (i.e., $\forall j \in [M]$ s.t., $j \geq m+1 : \tilde{\pi}_{j,t} = 0$). Note that any optimal solution of the LP in Grouping can be expressed as following distribution:

$$\forall i \in [m] : \tilde{\pi}_{i,t} = \tilde{\pi}_{m,t} + (m - i)\varepsilon \tag{5.2}$$

Assume that there exists $\varepsilon' < \varepsilon$ such that:

$$\tilde{\pi}'_{1,t} = \tilde{\pi}_{m',t} + (m' - 1)\varepsilon' > \tilde{\pi}_{m,t} + (m - 1)\varepsilon > \tilde{\pi}_{1,t}. \tag{5.3}$$

In other words, we are assuming that maximizing $\tilde{\pi}_{1,t}$ can be achieved with a number $m' > m$ of groups having non-zero probability, that have less "distance" between them. Then, summing up all arms' probabilities we get that:

$$\sum_{i\in[m]} |G_{i,t}|\tilde{\pi}_{i,t} = \sum_{i\in[m']} |G_{i,t}|\tilde{\pi}_{i,t} = 1.$$

Since $m' > m$ we have:

$$\sum_{i\in[m]} |G_{i,t}|\left(\tilde{\pi}_{i,t} - \tilde{\pi}'_{i,t}\right) = \sum_{i=m+1}^{i=m'} |G_{i,t}|\tilde{\pi}'_{i,t} > 0,$$

where the last inequality is due to our assumption that groups in $[m, m']$ get non-zero probability. So, there exists $j \in [m]$ such that $\tilde{\pi}'_{j,t} < \tilde{\pi}_{j,t}$. Hence:

$$\begin{aligned}\tilde{\pi}'_{1,t} &= \tilde{\pi}'_{m,t} + (m'-1)\varepsilon' = \tilde{\pi}'_{m,t} + (m'-1)\varepsilon' + (m'-j)\varepsilon' - (m'-j)\varepsilon' \\ &= \tilde{\pi}'_{j,t} + (j-1)\varepsilon' \\ &< \tilde{\pi}_{j,t} + (j-1)\varepsilon' < \tilde{\pi}_{j,t} + (j-1)\varepsilon = \tilde{\pi}_{1,t} \qquad (\varepsilon' < \varepsilon)\end{aligned}$$

which contradicts Equation (5.3). Hence, there is no ε' to achieve higher $\tilde{\pi}_{1,t}$ with $m' > m$ groups. Now, using that $\tilde{\pi}_{1,t}$ gets maximized with Equation (5.2) we can continue with the analysis on m. From the LP constraint we get:

$$1 = \sum_{i\in[m]} |G_{i,t}|\tilde{\pi}_{i,t} \geq \sum_{i\in[m]} \tilde{\pi}_{i,t} = \sum_{i\in[m]} \tilde{\pi}_{m,t} + (m-i)\varepsilon \geq \sum_{i\in[m]} (m-i)\varepsilon \geq \varepsilon\frac{m(m-1)}{2} \geq \frac{\varepsilon m^2}{2}.$$

As a result, $m \leq \sqrt{2/\varepsilon}$. Finally, since the number of groups m cannot exceed the number of arms, we get the result that $m = O(\min\{\sqrt{1/\varepsilon}, K\})$. □

Lemma 5.1.5. *Fix a pair of arms i, j in groups $G_{I,t}, G_{J,t}$ respectively. If arm i is placed in a group of a lower rank, then it has a higher upper confidence bound $u_{i,t} > u_{j,t}$.*

Proof. Since arm i got placed in group I, from the 4$^{\text{th}}$ line of Algorithm 13, it must be

$$u_{i,t} > L \text{ where } L = \min(l_{I*,t}, \min_{k\in G_{I-1,t}} (l_{k,t})).$$

Assuming $u_{j,t} \geq u_{i,t}$, we would get that $u_{j,t} > L$ which results in arm j being in a group of rank $J \leq I$. Thus, our assumption is contradicted. □

Lemma 5.1.6. *Fix a pair of arms i, j such that $\mu_i > \mu_j$ and a round $t \in [T]$. Let arms i, j be in groups $G_{I,t}, G_{J,t}$ respectively. If $I > J$ then $\mathbb{E}[n_{i,t}] > \mathbb{E}[n_{j,t}]$.*

Proof. Since $I > J$ Lemma 5.1.5 yields $u_{i,t} < u_{j,t}$. Thus, in expectation:

$$\mu_i + \mathbb{E}[\texttt{rad}_t(i)] < \mu_j + \mathbb{E}[\texttt{rad}_t(j)]. \qquad (\mathbb{E}[\widehat{\mu}_{i,t}] = \mu_i)$$

Given $\mu_i > \mu_j$ we get $\mathbb{E}[\texttt{rad}_t(i)] < \mathbb{E}[\texttt{rad}_t(j)]$, which provides our result. □

Lemma 5.1.7. *Fix a pair of arms i, j such that $\mu_i > \mu_j$ and a round $t \in [T]$. Then $\mathbb{E}[n_{i,t}] \geq \mathbb{E}[n_{j,t}]$.*

Proof. Let I_t, J_t be ranks of arms i, j respective groups in round t. If $I_t > J_t$ then the result follows trivially from Lemma 5.1.6. Else if $I_t \leq J_t$, then there can be two cases either (i) $I_{t'} \leq J_{t'}$ for all rounds $t' \in [t]$, which trivially results to $\mathbb{E}[n_{i,t}] \geq \mathbb{E}[n_{j,t}]$, or (ii) there exists a set of rounds $\mathcal{T} \subseteq [t]$ such that $I_{t'} > J_{t'}, \forall t' \in \mathcal{T}$.

Assuming the second case, let $t_0 = \max_{t\in\mathcal{T}}(t)$. Then, from Lemma 5.1.6:

$$\mathbb{E}[n_{i,t_0}] > \mathbb{E}[n_{j,t_0}]. \qquad (1)$$

From the definition of t_0, we get that $I_{t'} \leq J_{t'}$ for all rounds $t' \in (t_0, t]$. Thus, using the same argument as in case (i) we trivially get:

$$\mathbb{E}[n_{i,t_0:t}] > \mathbb{E}[n_{j,t_0:t}], \tag{2}$$

where $n_{i,t_1:t_2}$ is the number of pulls of arm i between rounds t_1 and t_2. Summing up (1), (2) we get our result. □

Proof. [Theorem 5.1.3] The regret in any stochastic-MAB problem can be written as:

$$R(T) = \sum_{i \in [K]} n_{i,T} \Delta_i,$$

where $n_{i,T}$ is the number of rounds arm i is played until time horizon T, and $\Delta_i = |\mu_{i^\star} - \mu_i|$. Let m be the bound from Lemma 5.1.4. We will argue that:

$$\mathbb{E}[R(T)] = O(m\sqrt{KT \log(KT/\delta)}).$$

First, we bound Δ_i using the number of groups with non-zero probability in the active set as well as the size of the confidence intervals of the arms in said groups.

Let $p_{i,t}$ be the pivot arm of group $G_{i,t}$. Arm $p_{i,t}$ must be linked to another arm a in group $G_{j,t}$ with $j < i$ (in order to be in the active set), but line 4 of algorithm 13 states that it cannot be linked to any arm in groups where $j < i - 1$. Thus, arm a must be in group $G_{i-1,t}$. Using a similar argument, arm a must be linked to either arm $p_{i-1,t}$ or to some arm in group $G_{i-2,t}$. Without loss of generality we assume the former case [1]. For simplicity in notation we will be using $u_{p_n,t}, l_{p_n,t}$ instead of $u_{p_{n,t},t}, l_{p_{n,t},t}$, in the equations below. So:

$$\begin{aligned}
u_{p_i,t} \geq l_{a,t} &= u_{a,t} - 2\texttt{rad}_t(a) && \text{(Arms } p_{i,t}, a \text{ are linked.)} \\
&\geq l_{p_{i-1},t} - 2\texttt{rad}_t(a) && \text{(Arms } a, p_{i-1,t} \text{ are linked.)} \\
&= u_{p_{i+1},t} - 2\texttt{rad}_t(p_{i-1,t}) - 2\texttt{rad}_t(a) \\
&= u_{p_{i+1},t} - 2\left(\texttt{rad}_t(p_{i-1,t}) + \texttt{rad}_t(a)\right).
\end{aligned}$$

Applying the above consecutively for the pivot arm $p_{n,t}$ of group $G_{n,t}$, we have the following bound:

$$u_{p_n,t} \geq u_{i^*} - 2 \sum_{j \in [n-1]} \{\texttt{rad}_t(p_{i-1,t}) + \texttt{rad}_t(a_j)\} \tag{5.4}$$

where arm a_j is the arm chaining $p_{j+1,t}$ to $p_{j,t}$ (see arm a above).

Fix arm i in group $G_{n,t}$ and let t be the last round arm i is pulled before it gets eliminated from the active set. In order for arm i to be placed in group $G_{n,t}$, it must be linked to either $p_{n,t}$ or some arm a in group $G_{n-1,t}$. Again, without loss of generality we assume the former case. After careful analysis (found in Appendix A.2), we obtain the following bound on Δ_i:

$$\Delta_i \leq 2\sqrt{2}m\sqrt{\frac{\log(KT/\delta)}{\gamma_i \cdot \mathbb{E}[n_{i,t}]}}.$$

[1] Assuming the latter case would only decrease the "length" of the chain to u_{i^*} in Equation 5.4 which makes no difference to our conclusion.

Now, we can compute the expected regret of Algorithm 12:

$$\begin{aligned}
\mathbb{E}[R(T)] &= \sum_{i\in[K]} \mathbb{E}[n_{i,T}]\Delta_i \\
&\leq 4m \sum_{i\in[K]} \mathbb{E}[n_{i,t}] \sqrt{\frac{\log(KT/\delta)}{\gamma_i \cdot \mathbb{E}[n_{i,T}]}} \\
&= O\left(\min\left(\sqrt{\frac{1}{\varepsilon}}, K\right)\sqrt{KT\log(KT/\delta)}\right). \qquad \text{(Appendix A.2)}
\end{aligned}$$

□

Chapter 6

Adversarial Attacks on Stochastic Bandits

What happens when an arm's reward is not representative of its quality? In the models described below an arm (or an adversary) may manipulate its reward (or the whole reward vector) to fool the learner. The goal of the learner is to be **robust** to such manipulation; i.e. maintain its regret bounds up to a factor dependent on the disruption observed. Real world behaviours that can be modeled as adversarial attacks on stochastic bandits include click fraud, fake reviews, spam emails and have been studied both from an algorithmic and from an economic aspect. In these settings the corruption of the rewards is considered to be adversarial without serving any strategic objective. This model is thoroughly presented in Section 6.2. Strategic manipulation is more suitable to describe behavior where arms/agents wish to fulfil a specific objective. The objective studied below is the maximization of the total number of pulls they get, which is a natural goal in the context of recommendation systems, where an arm wishes to maximize the times it gets recommended. In order for this objective to be fulfilled, an arm should appear to have a higher mean reward value. Feng et al. [21] use the example of a restaurant in a recommendation platform, that may lower its prices through user specific discounts so as to increase its click-through rate/rating score and thus its observed overall mean value. These actions are usually subject to a budget, since it is costly for an agent to provide unlimited discounts. The formal description of the latter model follows in Section 6.1.

6.1 Strategic Manipulation Model

The work of Braverman et al. [22] is the first to consider strategic reward manipulation. In their model, agents/arms present a lower reward, keeping the difference from the realized one as a utility for themselves. The opposite behavior, i.e., offering a higher reward (w.r.t. a budget) is studied in Feng et al. [21]. A combination of the above, where reward may be manipulated to appear both higher and lower is the case in Esmaeili et al. [23].

The model to describe strategic manipulation introduced by Feng et al. [21] consists of

the classic stochastic MAB model (see Section 3.1) enriched with $(B_i)_{i\in[K]}$ denoting an arm's manipulation budget. The sum of all budgets $(B_i)_{i\in[K]}$ is denoted with B. In this setting, arm i has a budget $B_i \geq 0$ that can be spent throughout all rounds $t \in [T]$ in order for the arm to appear to have a greater mean reward in the eyes of the learner. We will use the notation $r_{i,t}$ for the *true*/stochastic reward of arm i in round t and $\widehat{r}_{i,t}$ for the *manipulated* reward. The manipulated reward can be expressed as $\widehat{r}_{i,t} = r_{i,t} + a_{i,t}$, where $a_{i,t}$ is the budget spent by arm i in round t. The budget constraint states that for all arms $i \in [K]$: $0 \leq \sum_{t\in[T]} a_{i,t} = \sum_{t\in[T]} \widehat{r}_{i,t} - r_{i,t} \leq B_i$. The history $h_{i,t} = \{A_{t'}, r_{t'}, a_{i,t}\}_{t'\in[t]} \in \mathcal{H}_{i,t}$ observed by arm i until round t contains information of the algorithm's choices until round t and the manipulation added by arm i so far. The *adaptive* manipulation **strategy** $S^{(i)}$ of arm i is defined as a function $S^{(i)} : \mathcal{H}_{i,t} \times [K] \to \mathbb{R}$ that maps the arm's observed history until round t and the algorithm's pick A_t to a manipulation $a_{i,t}$. It is called an adaptive strategy because the arm is informed of the algorithm's choice A_t before setting the value of $a_{i,t}$.

Strategizing arises because arms are also equipped with an objective to maximize the expected number of pulls they get. For the rest of the analysis we will be using the terms arms and **agents** interchangeably to refer to arms with such an objective. Given this objective, arm i has no incentive to spend budget in rounds when $A_t \neq i$, thus $S^{(i)}(h_{i,t-1}, A_t) = 0$, if $A_t \neq i$. In this model, we are interested in the **robustness** of an algorithm; i.e. the ability of the algorithm to maintain its regret bound irrespective of the manipulation received.

Feng et al. [21] focus on studying the robustness of known algorithms used in the stochastic MAB setting. They prove that `UCB` (3.3.2), ε-`Greedy` (3.2) and `Thompson Sampling` ([59]) are intrinsically robust to strategic manipulation. Their result on the case of `UCB`[1] is the following.

Theorem 6.1.1. *For any manipulation strategy S of the strategic arms, the regret of the UCB principal is bounded by*

$$\mathbb{E}[R(T)] \leq \sum_{\substack{i\in[K]\\ i\neq i^*}} \left[\max\left\{3B_i, \frac{81\sigma^2 \ln T}{\Delta_i}\right\} + (1 + 3\Delta_i)\right]$$

The proof of the theorem can be found in Feng et al. [21]. It is important to observe that the above theorem holds for all manipulation strategies, even those not satisfying the arms' objective.

Feng et al. [21] show that their result is tight through equilibrium arguments on LIS (Lump Sum Investing) manipulation strategy, in which arm i spends all of its remaining budget on its first pull.

6.2 Adversarial Corruptions Model

We present a mixed adversarial and stochastic MAB model, where arms' stochastic rewards may be corrupted by an adversary. Before proceeding with the mixed

[1]The version of `UCB` they study is (α, ψ)-`UCB` from Bubeck and Cesa-Bianchi [3]. The distribution $\mathcal{D}$ of the arms' stochastic rewards is assumed to be σ-sub-Gaussian.

model we briefly describe the adversarial MAB setting to give a better intuition to the reader. The difference between the stochastic and the adversarial model is that rewards are not generated by a distribution $\mathcal{D}$ but they are arbitrarily chosen by an adversary. There are works in the adversarial bandit literature that study different types of adversaries (oblivious, adaptive) depending on the information they have about the learner's policy. The reader is referred to Auer et al. [53] for a deeper analysis of adversarial bandit problems. In the mixed model introduced by Lykouris et al. [11], arms' rewards are divided into a stochastic part (generated by a distribution $\mathcal{D}$) and an adversarial part (chosen by an adaptive adversary). We will use the notation $r_{i,t}$ for the *true*/stochastic reward of arm i in round t and $\widehat{r}_{i,t}$ for the *corrupted* reward. The corrupted reward can be expressed as $\widehat{r}_{i,t} = r_{i,t} + c_{i,t}$, where $c_{i,t}$ is the corruption added by the adversary. Let $R_t = (r_{i,t})_{i\in[K]} \in [0,1]^K$, $\widehat{R}_t = (\widehat{r}_{i,t})_{i\in[K]} \in [0,1]^K$ be the true and corrupted reward vectors, respectively and $h_t = \Big((A_1, \widehat{R}_1), \ldots, (A_t, \widehat{R}_t)\Big)$ the realized history. Fix a round t, the protocol between the learner and the adversary is described below.

1. The learner chooses a distribution π_t over arms $[K]$.
2. The environment sets the stochastic reward for each arm a: $r_{a,t} \sim \mathcal{D}_a$.
3. The adversary observes the realization of R_t as well as the history h_{t-1} and returns a corrupted reward vector $\widehat{R}_t$.
4. The learner pulls arm A_t according to her policy and observes reward $\widehat{r}_{A_t,t}$.

The mixed model can be viewed as a generalization of the strategic manipulation model where arm's i budget B_i does not need to be positive and arms are free from the pull maximization objective. Moreover, arms are informed about all the true rewards but are only aware of the distribution π_t on arms and not the algorithm's choice A_t, before manipulating their reward[2].

We present the two metrics proposed to measure the total corruption. Lykouris et al. [11] call an instance C-corrupted if for all realizations of the random variables:

$$\sum_t \max_a |\widehat{r}_{a,t} - r_{a,t}| \leq C.$$

Gupta et al. [12] define the level of corruption C as:

$$C = \sum_{i\in[T]} ||\widehat{R}_t - R_t||.$$

The changes between these metrics are negligible.

6.2.1 Adversarial attacks on `UCB`

The adversarial setting negates some of the previous results on standard stochastic MAB algorithms, like `UCB`. `UCB` is a deterministic algorithm, thus the distribution p_t published to the adversary trivially yields the algorithm's choice A_t.

[2]The distinction between π_t and A_t is only substantial if the learner is not following a deterministic policy.

Theorem 6.2.1. *Given a time horizon T, `UCB` ceases to be robust when run against a C-corrupted adversary with $C = \Omega(\log T)$ with probability greater than $1 - \frac{1}{\sqrt{T}}$.*

Proof. Assume an instance of two arms where $\mu_1 = 1$, $\mu_2 = 1/2$ and an adversary with the following policy:

$$c_{1,t} = -r_{1,t} \text{ if } A_t = 1; \text{ else } 0,$$

$$c_{2,t} = 0.$$

Using the above policy, the adversary achieves $\widehat{\mu}_{1,t} = 0, \forall t \in [T]$. Assume that there exists a round $t < T$ where $A_t = 1$ and $n_{1,t} > 8 \log T$. Then,

$$u_{1,t} = \widehat{\mu}_{1,t} + \texttt{conf}_t(i) < 1/2 = \mu_2.$$

From a standard application of the Hoeffding Inequality Theorem A.1.1 we get that $l_{2,t} \leq \mu_2 \leq u_{2,t}$ with probability greater that $1 - 1/\sqrt{T}$. Putting the second inequality together with the bound on $u_{1,2}$ we obtain

$$u_{1,t} < u_{2,t},$$

which contradicts our assumption that $A_t = 1$. Hence, arm 1 will be pulled in at most $8 \log T$ rounds. Computing the corruption level needed for this we get

$$C = \sum_{\substack{t \in [T] \\ A_t = 1}} |-r_{1,t}| \leq \sum_{\substack{t \in [T] \\ A_t = 1}} 1 \leq 8 \log T.$$

The regret accumulated is:

$$R(T) = \sum_{\substack{t \in [T] \\ A_t = 2}} \frac{1}{2} \geq (T - 8 \log T)\frac{1}{2} = \Omega(T).$$

We showed that corruptions of level $C = \Omega(\log T)$ make `UCB` suffer linear regret. □

However, restricting $c_{i,t} \geq 0$ for all arms $i \in [K]$ and all rounds $t \in [T]$ yields the following result.

Theorem 6.2.2. *In a mixed model with C-level corruption, `UCB` incurs regret*

$$\mathbb{E}[R(t)] = O\left(\sum_{i \in [K], i \neq i^*} \left[3\Delta_i + \frac{16 \log T}{\Delta_i}\right] + 4KC\right),$$

in each round t.

Proof. In the mixed model the empirical mean of arm i until round t can be expressed

as

$$\begin{aligned}\widehat{\mu}_{i,t} &= \frac{\sum_{t\in[T] \text{ and } A_t=i} \widehat{r}_{i_t}}{n_{i,t}} \\ &= \frac{\sum_{t\in[T] \text{ and } A_t=i} r_{i_t} + c_{i,t}}{n_{i,t}} \\ &\leq \frac{\sum_{t\in[T] \text{ and } A_t=i} r_{i_t}}{n_{i,t}} + \frac{C}{n_{i,t}} \\ &= \widetilde{\mu}_{i,t} + \frac{C}{n_{i,t}}, \end{aligned} \tag{6.1}$$

where $\widetilde{\mu}_{i,t}$ is the empirical mean of the stochastic part of the arm's reward. The proof is similar to the one from the classic stochastic model, found in Theorem 3.3.2, until Equation (3.6). We now choose N_i so that

$$\Delta_i - \sqrt{\frac{2\log(1/\delta)}{N_i}} - \frac{C}{N_i} \geq c\Delta_i, \tag{6.2}$$

where $c \in (0,1)$. Then P_2 can be written as

$$\begin{aligned} P_2 &= \mathbb{P}\left[\widehat{\mu}_{i,N_i} - \mu_i \geq \Delta_i - \sqrt{\frac{2\log(1/\delta)}{N_i}}\right] \\ &\leq \mathbb{P}\left[\widetilde{\mu}_{i,N_i} + \frac{C}{N_i} - \mu_i \geq \Delta_i - \sqrt{\frac{2\log(1/\delta)}{N_i}}\right] && \text{(Equation (6.1))} \\ &\leq \mathbb{P}[\widetilde{\mu}_{i,N_i} - \mu_i \geq c\Delta_i] && \text{(Equation (6.2))} \\ &\leq \exp\left(-\frac{N_i c^2 \Delta_i^2}{2}\right). \end{aligned}$$

The above yields Equation (3.8). Following the same steps as in Theorem 3.3.2, with $N_i = \left\lceil \frac{2\log(1/\delta)}{(1-c)^2\Delta_i^2} + \frac{4C}{(1-c)\Delta_i} \right\rceil$ and $c = 1/2$, we obtain

$$\mathbb{E}[n_{i,t}] \leq 3 + \frac{16\log T}{\Delta_i^2} + \frac{8C}{\Delta_i}.$$

The regret accumulated is

$$\begin{aligned} \mathbb{E}[R(t)] &= \sum_{i\in[K], i\neq i^*} \mathbb{E}[n_{i,t}]\Delta_i \\ &\leq \sum_{i\in[K], i\neq i^*} \left[3\Delta_i + \frac{16\log T}{\Delta_i}\right] + 4KC. \end{aligned}$$

□

The restriction $c_{i,t} \geq 0$ is crucial in order to bound

$$P_1 = \mathbb{P}\left[\mu^* \geq \min_{t\in[T]} u_{i^*,t}\right] \leq t\delta.$$

We know from Theorem 3.3.2 that $\sum_{i \neq i^*} \left[3\Delta_i + \frac{16 \log T}{\Delta_i} \right]$ yields the $\widetilde{O}(\sqrt{KT})$ bound, thus for $C = \widetilde{O}(\sqrt{T/K})$ `UCB` is robust in the mixed setting.

6.2.2 Algorithms robust to adversarial corruptions

Lykouris et al. [11] proposed the `Multi-layer Active Arm Elimination Race` algorithm, which achieves regret

$$O\left(\sum_{i \neq i^*} \frac{C \cdot K \log(KT/\delta) + \log T}{\Delta_i} \cdot \log(KT/\delta) \right),$$

with high probability. `Multi-layer Active Arm Elimination Race` has degrading regret performance, linearly to the corruption injected and it suffers linear regret for $C = \Omega(\sqrt{T/K})$. Gupta et al. [12] introduced the `BARBAR` (Bandit Algorithm with Robustness: Bad Arms get Recourse) algorithm, whose regret is bound by

$$O\left(KC + \sum_{i \neq i^*} \frac{\log T}{\Delta_i} \cdot \log\left(\frac{K}{\delta} \log T \right) \right),$$

with high probability. This algorithm retrieves the standard $\sqrt{KT}$ regret up to a logarithmic factor, when $C = 0$, and maintains it for $C = O(\sqrt{T/K})$. Both algorithms achieve the regret bounds while being **agnostic** to the amount of corruption injected. We present them below and provide a deeper analysis of the intuition behind `BARBAR`.

Multi-layer Active Arm Elimination Race

The idea behind `Multi-layer Active Arm Elimination Race` is to simultaneously run multiple instances of the `Successive Elimination` Algorithm, which we will be calling *layers*. In each round, the learner samples a layer $\ell \sim \mathcal{L}$ to update and the key notion to make a layer tolerate corruption is **sub-sampling**, through the distribution $\mathcal{L}$. For simplicity, assume that we keep 2 layers of `SE` and that C is known to the learner. Then, if the learner updates the first layer with probability $1 - 1/C$ and the second one with probability $1/C$, the second instance observes $O(1)$ corruptions in expectation. Using a concentration inequality, the second layer observes $O(\log T)$ corruptions, with high probability. Thus, enlarging the confidence radius of the second layer by $\log T / n_{i,t}^{\ell}$ makes it robust to corruption C. Since the second layer is corruption-tolerant, we know that it will not eliminate the best arm (with high probability). Thus, the scheme of **global eliminations** is proposed, which "broadcasts" eliminations made from a more trustworthy layer to less tolerant ones.

In order to extend the above to the case where C is unknown, Lykouris et al. [11] propose keeping $[\log T]$ layers of SE and using a distribution $\mathcal{L} = (2^{-l})_{l \in [\log T]}$[3]. Sub-sampling with $\mathcal{L}$, implies that layers $\ell \geq \log C$ observe $O(\log T)$ corruption

[3] In order for the sum of probabilities to be equal to 1, $\mathbb{P}[\ell = 1] = 1/2 + \left(1 - \sum_{l=1}^{\lceil \log T \rceil} 2^{-l}\right)$.

with high probability. Through the adequate enlargement in the confidence radius, which is set to $\texttt{wd}^\ell = O\left(\sqrt{\frac{\log T}{n^\ell_{i,t}}} + \frac{\log T}{n^\ell_{i,t}}\right)$, these layers become robust to corruption, while staying agnostic to its actual amount. In this case, an elimination happening in layer l is broadcast to all layers $l' < l$, that are deemed to be less tolerant.

The term *race* in the algorithm's name should now be more clear to the reader, as multiple layers race to find the optimal arm, while being corrected by eliminations made by layers more robust to corruption. The algorithm is presented below.

Algorithm 14: `Multi-layer Active Arm Elimination Race`

Initialize $n^\ell(a) = 0, \widetilde{\mu}^\ell(a) = 0, \mathcal{I}^\ell = \emptyset$ for all $a \in [K]$ and $\ell \in [\log T]$
for *Rounds* $t = 1..T$ **do**
  Sample layer $\ell \in [\log T]$ with probability $2^{-\ell}$. With remaining probability sample $\ell = 1$
  if $[K] \setminus \mathcal{I}^\ell \neq \emptyset$ **then**
    Play arm $a^t \leftarrow \arg\min_{a \in [K] \setminus \mathcal{I}^\ell} n^\ell(a)$
    Update $\widetilde{\mu}^\ell(a^t) \leftarrow \left[n^\ell(a)\widetilde{\mu}^\ell(a^t) + r^t(a^t)\right] / \left[n^\ell(a) + 1\right]$ and $n^\ell(a) \leftarrow n^\ell(a) + 1$
    while *exists arms* $a, a' \in [K] \setminus \mathcal{I}^\ell$ *with* $\widetilde{\mu}^\ell(a) - \widetilde{\mu}^\ell(a') > \texttt{wd}^\ell(a) + \texttt{wd}^\ell(a')$ **do**
      Eliminate a' by adding it to $\mathcal{I}^{\ell'}$ for all $\ell' \leq \ell$
  else
    Find minimum ℓ' such that $[K] \setminus \mathcal{I}^{\ell'} \neq \emptyset$ and play an arbitrary arm in that set

Their main result is the following.

Theorem 6.2.3. *Algorithm 14 which is agnostic to the corruption level C, when run with widths $\texttt{wd}^\ell = \sqrt{\frac{4K\log(T/\delta)}{n^\ell(a)}} + \frac{4K\log(T/\delta)}{n^\ell(a)}$ has regret:*

$$O\left(\sum_{i \neq i^*} \frac{C \cdot K\log(KT/\delta) + \log T}{\Delta_i} \cdot \log(KT/\delta)\right).$$

The regret expression is the sum of the regret accumulated by corruption-tolerant layers; i.e. layers $\ell \geq \log C$, and the regret accumulated by lower rank layers. The former incur the standard $O\left(\frac{\log T}{\Delta_i}\right)$. In order to bound the regret of corrupted layers, Lykouris et al. [11] bound the amount of rounds it takes for arm $i \neq i^*$ to be eliminated by a layer $\ell^\star \geq \log C$ and thus get eliminated in all lower levels. The reader is referred to [11] for a more detailed analysis.

Their work also provides a lower bound on the regret of any MAB algorithm with standard pseudo-regret.

Theorem 6.2.4. *Consider a multi-armed bandits algorithm that has the property that for any stochastic input in the two arm setting, it has pseudo-regret bounded by $\frac{c\log T}{\Delta}$ where $\Delta = |\mu_1 - \mu_2|$. For any $\varepsilon, \varepsilon' \in (0,1)$, there is a corruption level C with $T^\varepsilon < C < T^{\varepsilon'}$ and a C-corrupted instance such that with constant probability the regret is $\Omega(C)$.*

BARBAR Algorithm

BARBAR algorithm runs in epochs, with exponentially increasing number of rounds; i.e. the m^{th} epoch lasts for roughly $N_m = 2^{2m}$ rounds. The intuition behind the algorithm is that corruption happening in epoch m can only affect the algorithm's choices in epoch $m+1$, thus an exponentially increasing amount of corruption C_m is needed to keep manipulating the learner's observations. This is implemented through constructing a distribution over arms taking into consideration an estimation over Δ_i, computed as $\Delta_i^m = \max\{2^{-m}, r_\star^m - r_i^m\}$, where r_i^m is the observed mean reward of arm i in epoch m and $r_\star^m = \max_i\{r_i^m - \frac{1}{16}\Delta_i^{m-1}\}$ is a lower confidence bound for the observed mean reward of the best arm in epoch m using information of the arm's rewards from epoch $m-1$.

Algorithm 15: BARBAR

Parameters: confidence $\delta \in (0,1)$, time horizon T.
Initialize $T_0 = 0$ and $\Delta_i^0 = 1$ for all $i \in [K]$.
Set $\lambda = 1024 \ln(\frac{8K}{\delta} \log_2 T)$.
for *epochs* $m = 1, 2, \dots$ **do**
  Set $n_i^m = \lambda(\Delta_i^{m-1})^{-2}$ for all $i \in [K]$.
  Set $N_m = \sum_{i=1}^K n_i^m$ and $T_m = T_{m-1} + N_m$.
  for $t = T_{m-1} + 1$ *to* T_m **do**
    Choose an arm i with probability n_i^m / N_m and pull it.
  Set $r_i^m = S_i / n_i^m$ where S_i is the total reward from the pulls of arm i in this epoch.
  Set $r_\star^m = \max_i\{r_i^m - \frac{1}{16}\Delta_i^{m-1}\}$
  Set $\Delta_i^m = \max\{2^{-m}, r_\star^m - r_i^m\}$

Theorem 6.2.5. *With probability at least $1-\delta$, the regret of Algorithm 15 is bounded by*

$$O\left(KC + \sum_{i \neq i^*} \frac{\log T}{\Delta_i} \log\left(\frac{K}{\delta} \log T\right)\right).$$

Before proceeding with the main part of the proof, Gupta et al. [12] prove the following crucial inequalities.

- An upper and lower bound on the length N_m of epoch m is given by

$$\lambda 2^{2(m-1)} \leq N_m \leq K\lambda 2^{2(m-1)}. \tag{6.3}$$

  (Lemma 2 from [12])

- Let C_m be the random variable denoting the sum of the corruptions in epoch m. Let $\tilde{n}_i^m$ be the random variable denoting the actual number of pulls of arm i in epoch m. Then, event $\mathcal{E}$ is defined as:

$$\mathcal{E} := \left\{\forall m, i : |r_i^m - \mu_i| \leq \frac{2C_m}{N_m} + \frac{\Delta_i^{m-1}}{16} \text{ and } \tilde{n}_i^m \leq 2n_i^m\right\}.$$

  Using a multiplicative Chernoff-Hoeffding bound (A.1.2) on the second condition of $\mathcal{E}$ and a Freedman-type concentration inequality for martingales for

the first one, they prove that

$$\mathbb{P}[\mathcal{E}] \geq 1 - \delta. \tag{6.4}$$

(Lemma 3 from [12])

- The sum $\rho_m := \sum_{s=1}^{m} \frac{2C_s}{8^{m-s}N_s}$ is defined as the discounted corruption rate and appears to be very useful to the analysis of the last inequality. Conditioning on $\mathcal{E}$, then for all epochs m and arms i it holds that

$$\Delta_i^m \geq \frac{1}{2}\Delta_i - 3\rho_m - \frac{3}{4}2^{-m}. \tag{6.5}$$

(Lemma 7 from [12])

Proof. Under event $\mathcal{E}$, the regret of Algorithm 15 can be expressed as

$$\mathcal{R} = \sum_{m=1}^{M}\sum_{i=1}^{K} \Delta_i \tilde{n}_i^m \leq 2\sum_{m=1}^{M}\sum_{i=1}^{K} \Delta_i n_i^m. \tag{6.6}$$

The proof focuses on bounding $\mathcal{R}_i^m = \Delta_i n_i^m$ in the following three cases: (i) $0 < \Delta_i \leq \frac{4}{2^m}$; (ii) $\Delta_i > \frac{4}{2^m}$ and $\rho_{m-1} < \frac{\Delta_i}{32}$; and (iii) $\Delta_i > \frac{4}{2^m}$ and $\rho_{m-1} \geq \frac{\Delta_i}{32}$.

Case (i) Using $n_i^m \leq \lambda 2^{2(m-1)}$ from lines 5,11 of the Algorithm 15 we obtain

$$\mathcal{R}_i^m \leq \frac{4\lambda}{\Delta_i}.$$

Case (ii) Given the bounds on ρ_{m-1}, Δ_i and 6.5 we have the following bound on Δ_i^{m-1}

$$\Delta_i^{m-1} \geq \frac{1}{2}\Delta_i - \frac{3}{32}\Delta_i - \frac{3}{4}2^{-(m-1)} \geq \Delta_i\left(\frac{1}{2} - \frac{3}{32} - \frac{3}{8}\right) \geq \frac{1}{32}\Delta_i$$

The definition of n_i^m incurs $n_i^m = \frac{\lambda}{\Delta_i^{m-1^2}} \leq \frac{32^2\lambda}{\Delta_i^2}$. Thus, we obtain

$$\mathcal{R}_i^m \leq \frac{32^2\lambda}{\Delta_i}.$$

Case (iii) Using $n_i^m \leq \lambda 2^{2(m-1)}$ as in case (i) together with $\Delta_i \leq 32\rho_{m-1}$ we get

$$\mathcal{R}_i^m \leq 8\lambda\rho_{m-1}2^{2m}$$

Summing up the bounds of all three cases, Equation (6.6) can be written as

$$\mathcal{R} \leq 32^2\lambda\sum_{i\neq i^*}\frac{\log T}{\Delta_i} + 8\lambda\sum_{i\neq i^*}\sum_{m=1}^{M}\rho_{m-1}2^{2m}.$$

Using the definition of ρ_{m-1} the second term is bounded by

$$\begin{aligned}
\sum_{m=1}^{M} \rho_{m-1} 2^{2m} &\leq \sum_{m=1}^{M} 2^{2m} \sum_{s=1}^{m-1} \frac{2C_s}{8^{m-1-s} N_s} \\
&= 2 \sum_{s=1}^{M} C_s \sum_{m=s}^{M} \frac{2^{2m}}{8^{m-1-s} N_s} \\
&\leq 2 \sum_{s=1}^{M} C_s \frac{16}{\lambda} \cdot \sum_{m=s}^{M} \frac{4^{m-1-s}}{8^{m-1-s}} && (N_s \geq \lambda 2^{2(s-1)} \text{ from 6.3}) \\
&\leq \frac{32}{\lambda} \sum_{s=1}^{M} C_s \sum_{j=1}^{\infty} 2^{-j} \\
&\leq \frac{32}{\lambda} \sum_{s=1}^{M} C_s = \frac{32C}{\lambda}.
\end{aligned}$$

The desired bound on $\mathcal{R}$ follows trivially. □

The authors also provide better regret bounds under some special assumptions. The cases of known corruption level, corruption on an unknown prefix C (the adversary only corrupts the rewards of the firsy C rounds) and known maximal mean reward μ^* remove the K from the KC dependence in the first regret term. The case of fixed (unknown) corruption rate $\eta \in (0, 1)$ switches the dependence on KC to ηT and knowning the minimal gap $\Delta = \min_{i \neq i^*} \Delta_i$ incurs regret $\widetilde{O}\left(C + \frac{K\lambda}{\Delta}\right)$.

Bibliography

[1] M. Joseph, M. Kearns, J. Morgenstern, and A. Roth, "Fairness in learning: Classic and contextual bandits," 2016.

[2] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, pp. 285–294, 1933. [Online]. Available: https://api.semanticscholar.org/CorpusID:120462794

[3] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *CoRR*, vol. abs/1204.5721, 2012. [Online]. Available: http://arxiv.org/abs/1204.5721

[4] T. Lattimore and C. Szepesvári, *Bandit Algorithms*, 07 2020.

[5] A. Slivkins, "Introduction to multi-armed bandits," 2022.

[6] E. Even-Dar, S. Mannor, and Y. Mansour, "Pac bounds for multi-armed bandit and markov decision processes," in *Annual Conference Computational Learning Theory*, 2002. [Online]. Available: https://api.semanticscholar.org/CorpusID:12757817

[7] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multi-armed bandit problem," *Machine Learning*, vol. 47, pp. 235–256, 05 2002a.

[8] E. Ferrara, "Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies," *Sci*, vol. 6, no. 1, p. 3, Dec. 2023. [Online]. Available: http://dx.doi.org/10.3390/sci6010003

[9] J. Angwin, J. Larson, S. Mattu, and K. Lauren, "Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks." 05 2016. [Online]. Available: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[10] A. Lambrecht and C. Tucker, "Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads," *Management Science*, vol. 65, 04 2019.

[11] T. Lykouris, V. Mirrokni, and R. Paes Leme, "Stochastic bandits robust to adversarial corruptions," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, 2018, pp. 114–122.

[12] A. Gupta, T. Koren, and K. Talwar, "Better algorithms for stochastic bandits with adversarial corruptions," in *Conference on Learning Theory*. PMLR, 2019, pp. 1562–1578.

[13] A. A. Alter and E. A. Harris, "https://www.nytimes.com/2023/06/26/books/goodreads-review-bombing.html," 2023. [Online]. Available: https://www.nytimes.com/2023/06/26/books/goodreads-review-bombing.html

[14] R. R. Bush and F. Mosteller, "A stochastic model with applications to learning," *The Annals of Mathematical Statistics*, pp. 559–585, 1953.

[15] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM J. Comput.*, vol. 32, pp. 48–77, 2002b. [Online]. Available: https://api.semanticscholar.org/CorpusID:13209702

[16] L. Wang, Y. Bai, W. Sun, and T. Joachims, "Fairness of exposure in stochastic bandits," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 686–10 696.

[17] F. Li, J. Liu, and B. Ji, "Combinatorial sleeping bandits with fairness constraints," *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1702–1710, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:58006636

[18] H. Claure, Y. Chen, J. Modi, M. Jung, and S. Nikolaidis, "Multi-armed bandits with fairness constraints for distributing resources to human teammates," 2020.

[19] Y. Chen, A. Cuellar, H. Luo, J. Modi, H. Nemlekar, and S. Nikolaidis, "Fair contextual multi-armed bandits: Theory and experiments," 2019.

[20] V. Patil, G. Ghalme, V. Nair, and Y. Narahari, "Achieving fairness in the stochastic multi-armed bandit problem," 2020.

[21] Z. Feng, D. Parkes, and H. Xu, "The intrinsic robustness of stochastic bandits to strategic manipulation," in *International Conference on Machine Learning*. PMLR, 2020, pp. 3092–3101.

[22] M. Braverman, J. Mao, J. Schneider, and S. M. Weinberg, "Multi-armed bandit problems with strategic arms," in *Conference on Learning Theory*. PMLR, 2019, pp. 383–416.

[23] S. A. Esmaeili, S. Shin, and A. Slivkins, "Robust and performance incentivizing algorithms for multi-armed bandits with strategic agents," *arXiv preprint arXiv:2312.07929*, 2023.

[24] D. Bouneffouf and I. Rish, "A survey on practical applications of multi-armed and contextual bandits," 2019.

[25] D. Bouneffouf, I. Rish, and G. A. Cecchi, "Bandit models of human behavior: Reward processing in mental disorders," in *Artificial General Intelligence: 10th International Conference, AGI 2017, Melbourne, VIC, Australia, August 15-18, 2017, Proceedings 10*. Springer, 2017, pp. 237–248.

[26] A. Durand, C. Achilleos, D. Iacovides, K. Strati, G. D. Mitsis, and J. Pineau, "Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis," in *Machine learning for healthcare conference*. PMLR, 2018, pp. 67–82.

[27] H. Lei, Y. Lu, A. Tewari, and S. A. Murphy, "An actor-critic contextual bandit algorithm for personalized mobile health interventions," *arXiv preprint arXiv:1706.09090*, 2017.

[28] W. Shen, J. Wang, Y.-G. Jiang, and H. Zha, "Portfolio choices with orthogonal bandit learning," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[29] X. Huo and F. Fu, "Risk-aware multi-armed bandit problem with application to portfolio selection," *Royal Society open science*, vol. 4, no. 11, p. 171377, 2017.

[30] K. Misra, E. M. Schwartz, and J. Abernethy, "Dynamic online pricing with incomplete information using multiarmed bandit experiments," *Marketing Science*, vol. 38, no. 2, pp. 226–252, 2019.

[31] J. W. Mueller, V. Syrgkanis, and M. Taddy, "Low-rank bandit methods for high-dimensional dynamic pricing," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[32] D. Soemers, T. Brys, K. Driessens, M. Winands, and A. Nowé, "Adapting to concept drift in credit card transaction data streams using contextual bandits and decision trees," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[33] K. Ding, J. Li, and H. Liu, "Interactive anomaly detection on attributed networks," in *Proceedings of the twelfth ACM international conference on web search and data mining*, 2019, pp. 357–365.

[34] Q. Zhou, X. Zhang, J. Xu, and B. Liang, "Large-scale bandit approaches for recommender systems," in *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part I 24*. Springer, 2017, pp. 811–821.

[35] D. Bouneffouf, R. Laroche, T. Urvoy, R. Féraud, and R. Allesiardo, "Contextual bandit for active learning: Active thompson sampling," in *Neural Information Processing: 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part I 21*. Springer, 2014, pp. 405–412.

[36] G. Bresler, D. Shah, and L. F. Voloch, "Collaborative filtering with low regret," in *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*, 2016, pp. 207–220.

[37] S. Li, A. Karatzoglou, and C. Gentile, "Collaborative filtering bandits," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2016, pp. 539–548.

[38] J. Dai, B. Flanigan, N. Haghtalab, M. Jagadeesan, and C. Podimata, "Can probabilistic feedback drive user impacts in online platforms?" *arXiv preprint arXiv:2401.05304*, 2024.

[39] K. Roose, "The making of a youtube radical," *The New York Times*, vol. 8, 2019.

[40] A. Y. Agan, D. Davenport, J. Ludwig, and S. Mullainathan, "Automating automaticity: How the context of human choice affects the extent of algorithmic bias," National Bureau of Economic Research, Tech. Rep., 2023.

[41] J. Jiang, S. Sun, V. Sekar, and H. Zhang, "Pytheas: Enabling {Data-Driven} quality of experience optimization using {Group-Based}{Exploration-Exploitation}," in *14th USENIX symposium on networked systems design and implementation (NSDI 17)*, 2017, pp. 393–406.

[42] M. Dong, T. Meng, D. Zarchy, E. Arslan, Y. Gilad, B. Godfrey, and M. Schapira, "{PCC} vivace:{Online-Learning} congestion control," in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, 2018, pp. 343–356.

[43] L. Sun, J. Hou, and T. Shu, "Spatial and temporal contextual multi-armed bandit handovers in ultra-dense mmwave cellular networks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 12, pp. 3423–3438, 2020.

[44] H. E. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, pp. 527–535, 1952. [Online]. Available: https://api.semanticscholar.org/CorpusID:15556973

[45] F. J. Anscombe, "Sequential medical trials," *Journal of the American Statistical Association*, vol. 58, pp. 365–383, 1963. [Online]. Available: https://api.semanticscholar.org/CorpusID:120532497

[46] R. Sutton and A. Barto, "Reinforcement learning: An introduction," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 1054–1054, 1998.

[47] P. Auer and R. Ortner, "Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem," *Periodica Mathematica Hungarica*, vol. 61, pp. 55–65, 09 2010.

[48] T. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0196885885900028

[49] T. L. Lai, "Adaptive treatment allocation and the multi-armed bandit problem," *Annals of Statistics*, vol. 15, pp. 1091–1114, 1987. [Online]. Available: https://api.semanticscholar.org/CorpusID:120034176

[50] R. Agrawal, "Sample mean based index policies by o(log n) regret for the multi-armed bandit problem," *Advances in Applied Probability*, vol. 27, pp. 1054 – 1078, 1995. [Online]. Available: https://api.semanticscholar.org/CorpusID:120313529

[51] L. P. Kaelbling, *Learning in embedded systems*. MIT Press, 1993.

[52] L. Besson and E. Kaufmann, "What doubling tricks can and can't do for multi-armed bandits," 2018.

[53] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pp. 322–331, 1995. [Online]. Available: https://api.semanticscholar.org/CorpusID:8963242

[54] A. Köchling and M. C. Wehner, "Discriminated by an algorithm: a systematic review of discrimination and fairness by algorithmic decision-making in the context of hr recruitment and hr development," *Business Research*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:228855538

[55] Z. Obermeyer, B. Powers, C. Vogeli, and S. Mullainathan, "Dissecting racial bias in an algorithm used to manage the health of populations," *Science*, vol. 366, no. 6464, pp. 447–453, 2019. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.aax2342

[56] P. Jaillet, C. Podimata, and Z. Zhou, "Grace period is all you need: Individual fairness without revenue loss in revenue management," *ArXiv*, vol. abs/2402.08533, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:267637245

[57] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," 2011.

[58] Y. Liu, G. Radanovic, C. Dimitrakakis, D. Mandal, and D. C. Parkes, "Calibrated fairness in bandits," 2017.

[59] D. Russo, B. V. Roy, A. Kazerouni, I. Osband, and Z. Wen, "A tutorial on thompson sampling," 2020.

[60] S. Hossain, E. Micha, and N. Shah, "Fair algorithms for multi-agent multi-armed bandits," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 24 005–24 017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/c96ebeee051996333b6d70b2da6191b0-Paper.pdf

[61] H. Moulin, *Fair division and collective welfare*. MIT press, 2004.

[62] N. Cesa-Bianchi and G. Lugosi, "Combinatorial bandits," *J. Comput. Syst. Sci.*, vol. 78, pp. 1404–1422, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:6854100

[63] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 151–159. [Online]. Available: https://proceedings.mlr.press/v28/chen13a.html

[64] R. D. Kleinberg, A. Niculescu-Mizil, and Y. Sharma, "Regret bounds for sleeping experts and bandits," *Machine Learning*, vol. 80, pp. 245–272, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:15442140

[65] G. Fang, P. Li, and G. Samorodnitsky, "On penalization in stochastic multi-armed bandits," *ArXiv*, vol. abs/2211.08311, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253523030

[66] J. A. Killian, M. Jain, Y. Jia, J. Amar, E. Huang, and M. Tambe, "Equitable restless multi-armed bandits: A general framework inspired by digital health," 2023.

Appendix A

Supplementary Material

A.1 Concentration Inequalities

Theorem A.1.1 (Hoeffding's Inequality)**.** *Let $X_1, \ldots, X_t$ be independent $[0,1]$-valued random variables and let $X = \sum_{i\in[t]} X_i$. Then for all $\epsilon \geq 0$*

$$\mathbb{P}[X - \mathbb{E}[X] \geq \epsilon] \leq \exp(-2\varepsilon^2 t)$$

and

$$\mathbb{P}[X - \mathbb{E}[X] \leq -\epsilon] \leq \exp(-2\varepsilon^2 t).$$

Theorem A.1.2 (Multiplicative Chernoff Bound)**.** *Let $X_1, \ldots, X_t$ be independent $[0,1]$-valued random variables and let $X = \sum_{i\in[t]} X_i$. Then for any $\epsilon > 0$*

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

Theorem A.1.3 (Jensen's Inequality)**.** *Let g be a concave function and let X be a random variable. Then*

$$g(\mathbb{E}[X]) \geq \mathbb{E}[g(X)].$$

Theorem A.1.4. *KL-divergence satisfies the following properties:*

1. *$KL(p,q) \geq 0$ for any two distributions p, q with the equality holding if and only if $p = q$.*
   (Gibbs' Inequality)
2. *Let the sample space be a product $\Omega = \Omega_1 \times \cdots \times \Omega_n$. Let p, q be two distributions on Ω such that $p = p_1 \times \cdots \times p_n$ and $q = q_1 \times \cdots \times q_n$, where p_j, q_j are distributions on Ω_j, for each $j \in [n]$. Then $KL(p,q) = \sum_{j=1}^{n} KL(p_j, q_j)$.*
   (Chain rule)
3. *For any event $A \subset \Omega$ it holds that $2\left(p(A) - q(A)\right)^2 \leq KL(p,q)$.*
   (Pinsker's Inequality)

4. *Let p, q be the probability distributions of two random coins with expected mean $1/2, (1+\epsilon)/2$ respectively. We use the notation $p = RC_0, q = RC_\epsilon$. Then $KL(RC_\epsilon, RC_0) \leq 2\epsilon^2$ and $KL(RC_0, RC_\epsilon) \leq \epsilon^2$ for all $\epsilon \in (0, 1/2)$.*

Theorem A.1.5 (Markov's Inequality)**.** *If $\mathcal{X}$ is a non-negative random variable and $a > 0$, then*

$$\mathbb{P}[\mathcal{X} \geq a] \leq \frac{\mathbb{E}[\mathcal{X}]}{a}.$$

A.2 Omitted proofs

Proof. (Missing derivation of bound on Δ_i in Theorem 5.1.3)

$$\begin{aligned}
l_{i,t} &\geq u_{p_n,t} - 2(\texttt{rad}_t(p_n) + \texttt{rad}_t(i)) \\
&\geq u_{i^*} - 2\sum_{j\in[n-1]} (\texttt{rad}_t(p_j) + \texttt{rad}_t(a_j)) - 2\left(\texttt{rad}_t(p_n) + \texttt{rad}_t(i)\right) \\
&\qquad\qquad \text{(Equation (5.4).)} \\
&\geq u_{i^*} - 2\sum_{j\in[n]} (\texttt{rad}_t(p_j) + \texttt{rad}_t(a_j)). \qquad (a_n = i.)
\end{aligned}$$

Replacing $\texttt{rad}_t(i)$ with $\sqrt{\frac{\ln \pi(T+1)^2/3\delta}{2n_{i,t}}} = \sqrt{\frac{L}{2n_{i,t}}}$ we get:

$$\begin{aligned}
u_{i^*} - l_{i,t} &\leq 2\sum_{j\in[n]} \left(\sqrt{\frac{L}{2n_{p_{j,t},t}}} + \sqrt{\frac{L}{2n_{a_j,t}}}\right) \\
&\leq 2\sum_{j\in[n-1]} \left(\sqrt{\frac{L}{2\gamma_{p_{j,t}} \cdot E[n_{p_{j,t},t}]}} + \sqrt{\frac{L}{2\gamma_{a_j} \cdot E[n_{a_j,t}]}}\right) \\
&\qquad\qquad \text{(Setting } \gamma_i = 1 - \sqrt{\tfrac{3\log(2KT^2)}{\mathbb{E}[n_{i,t}]}} \text{ in A.1.2.)} \\
&\leq 2\sum_{j\in[n]} 2\sqrt{\frac{L}{2\gamma_i \cdot E[n_{i,t}]}} \qquad \text{(Lemma 5.1.7.)} \\
&\leq 2\sqrt{2}n\sqrt{\frac{L}{\gamma_i \cdot \mathbb{E}[n_{i,t}]}}.
\end{aligned}$$

Lemma 5.1.7 applies to arms that are proven to be "better" than (i.e., got un-linked from) arm i, but the sum in the last inequality contains arms $p_{n,t}, a_{n-1}$ which are not yet un-linked from arm i. So the above is missing an argument about arm i being un-linked from group n after round t. The only reason why this may not be true is that arm i might be eliminated because the chain broke in a point closer to arm i^* and not because arm i itself got unchained from all other arms (and thus got deactivated). However, it is obvious that regret gets maximized if arms get eliminated one at a time, thus the above does not alter our result.

We upper bound the last term above using the fact that round t is the last round in which arm i gets pulled, which means that probability $\pi_{i,t}$ is non-zero. Thus, using

Lemma 5.1.4, n cannot exceed the maximum number m of groups with non-zero probability.

Finally, since arm i is active in round t, it is still chained to i^*. So with probability at least $1-\delta$:

$$\begin{aligned}\Delta_i = \mu_{i^*} - \mu_i &\le u_{i^*,t} - l_{i,t} \\ &\le 2\sqrt{2}m\sqrt{\frac{\ln(\pi(T+1)^2/3\delta)}{\gamma_i \cdot \mathbb{E}[n_{i,t}]}}. && (n \le m)\end{aligned}$$

□

Proof. (Missing derivation of $\mathbb{E}[R(T)]$ in Theorem 5.1.3)

$$\begin{aligned}E[R(T)] = \sum_{i\in[K]} E[n_{i,T}]\Delta_i &\le 2\sqrt{2}m \sum_{i\in[K]} E[n_{i,T}]\sqrt{\frac{\ln(\pi(T+1)^2/3\delta)}{\gamma_i \cdot E[n_{i,T}]}} + O\left(\delta + \frac{1}{T}\right)T \\ &\le 2\sqrt{2}m\left(\sum_{i\in[K]s.t.\gamma_i>1/2} E[n_{i,T}]\sqrt{\frac{2\ln(\pi(T+1)^2/3\delta)}{E[n_{i,T}]}} + \sum_{i\in[K]s.t.\gamma_i\le 1/2} E[n_{i,T}]\cdot 1\right) + O\left(\sqrt{T}\right) \\ &&& (\delta \le \tfrac{1}{\sqrt{T}}) \\ &\le 4m\left(\sum_{i\in[K]}\sqrt{E[n_{i,T}]\ln(\pi(T+1)^2/3\delta)} + 12K\log(2KT^2)\right) + O\left(\sqrt{T}\right) \\ &&& (\text{If } \gamma_i \le 1/2 \text{ then } E[n_{i,T}] \le 12\log(2KT^2).) \\ &\le 4m\left(\sqrt{KT\ln(\pi(T+1)^2/3\delta)} + 12K\log(2KT^2)\right) + O\left(\sqrt{T}\right) \\ &&& (\text{Jensen's Inequality.}) \\ &= O\left(m\sqrt{KT\ln(\pi(T+1)^2/3\delta)} + 12K\log(2KT^2)\right) + O\left(\sqrt{T}\right) \\ &= O\left(\min\left(\sqrt{\frac{1}{\varepsilon}}, K\right)\sqrt{KT\ln(KT/\delta)}\right). && (\text{Lemma } 5.1.4.)\end{aligned}$$

□