On Bandit Learning and Pricing in Markets

Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften

vorgelegt beim Fachbereich Informatik
der Johann Wolfgang Goethe-Universität
in Frankfurt am Main

von

Paresh Nakhe
aus Kolhapur, Indien

Frankfurt 2018
(D 30)

vom Fachbereich 12 Informatik und Mathematik der

Johann Wolfgang Goethe - Universität als Dissertation angenommen.

Dekan : Prof. Dr. Andreas Bernig

Gutachter : Prof. Dr. Martin Hoefer

Prof. Dr. Georg Schnitger

Datum der Disputation : 2018

Acknowledgements

First and foremost, I would like to thank my parents for their unwavering belief in me and for the countless sacrifices they have had to make to get me where I am. I would like to especially thank my father, who encouraged me to dream big and stood by me whenever I doubted myself. I would like to thank my brother for his patience and support during my weakest moments. Also, my niece whose innocent smile always helped me lighten up. I am also grateful to my friend Andrea for her patience with me during this time.

I would like to thank my advisor for his support throughout the course of my Ph.D. I am grateful to him for granting me the freedom to pursue my interests even when they were not aligned with his. I would like to thank my other collaborators Rebecca Reiffenhäuser and Yun Kuen Cheung for their contributions. I learnt a lot from both of them. Last but not the least, I am grateful for the support of Max Planck Institut, Saarbrücken, where I did a part of my work, as also Goethe University.

Abstract

A lot of software systems today need to make real-time decisions to optimize an objective of interest. This could be maximizing the click-through rate of an ad displayed on a web page or profit for an online trading software. The performance of these systems is crucial for the parties involved. Although great progress has been made over the years in understanding such online systems and devising efficient algorithms, a fine-grained analysis and problem specific solutions are often missing. This dissertation focuses on two such specific problems: bandit learning and pricing in gross-substitutes markets.

Bandit learning problems are a prominent class of sequential learning problems with several real-world applications. The classical algorithms proposed for these problems, although optimal in a theoretical sense often tend to overlook model-specific properties. With this as our motivation, we explore several sequential learning models and give efficient algorithms for them. Our approaches, inspired by several classical works, incorporate the model-specific properties to derive better performance bounds.

The second part of the thesis investigates an important class of price update strategies in static markets. Specifically, we investigate the effectiveness of these strategies in terms of the total revenue generated by the sellers and the convergence of the resulting dynamics to market equilibrium. We further extend this study to a class of dynamic markets. Interestingly, in contrast to most prior works on this topic, we demonstrate that these price update dynamics may be interpreted as resulting from revenue optimizing actions of the sellers. No such interpretation was known previously. As a part of this investigation, we also study some specialized forms of no-regret dynamics and prediction techniques for supply estimation. These approaches based on learning algorithms are shown to be particularly effective in dynamic markets.

Zusammenfassung

Viele Softwaresysteme müssen heute Echtzeitentscheidungen treffen, um eine bestimmte Zielfunktion zu optimieren. Das Ziel könnte die Maximierung der Klickrate einer Internetanzeige sein oder die Optimierung des Gewinns einer Online-Handelssoftware. Die Leistung dieser Systeme ist für die beteiligten Parteien von entscheidender Bedeutung. Obwohl im Laufe der Jahre große Fortschritte beim Verständnis solcher Online-Systeme und der Entwicklung effizienter Algorithmen gemacht wurden, fehlen oft eine feingranular Analyse und problemspezifische Lösungen. Diese Dissertation konzentriert sich auf zwei spezifische Probleme dieser Art: Bandit-Lernen und Preisgestaltung in Märkten.

Im ersten Teil untersuchen wir Verlustmodelle, die eine Mischung aus den gut untersuchten gegnerischen und stochastischen Modellen sind. Dabei ergeben sich Algorithmen, die die zusätzliche Struktur ausnutzen. Wir zeigen auch, wie einige existierende Techniken und Ideen genutzt werden können, um effiziente Algorithmen für unsere problemspezifischen Modelle zu entwickeln. Im zweiten Teil untersuchen wir das etablierte Konzept des Marktgleichgewichts und zeigen dabei, dass dieses Lösungskonzept auch als Ergebnis der strategischen Interaktionen zwischen den beteiligten Akteuren, nämlich den Käufern und Verkäufern, entsteht. Darüber hinaus analysieren wir diese Interaktionen für dynamische Märkte und demonstrieren ihre Wirksamkeit bei der Aufrechterhaltung einer ungefähren Markträumung.

Bandit-Lernen

Betrachte das folgende klassische Problem das aus einem lernenden Agenten und einer Menge von Aktionen besteht. In jedem Zeitschritt wählt der Agent eine dieser Aktionen und bekommt einen zugehörigen Nutzen. Er lernt jedoch nichts über den Nutzen, den er bekommen hätte, wenn er eine der anderen Aktionen gewählt hätte. Die Nutzenwerte, die mit den Aktionen assoziiert sind, können entweder beliebig sein oder aus einer Wahrscheinlichkeitsverteilung gezogen werden. Das hängt von dem Modell ab. Dieses Problem heißt in Literatur *Multi-armed Bandit Problem.*

Dieses Problem ist in den letzten zwei Jahrzehnten in vielen Varianten erforscht worden. Das Interesse besteht nicht nur für die theoretischen Aspekte, sondern auch für die Anwendbarkeit aus reale Probleme. Betrachte das folgende Beispiel als Motivation:

Eine Hotelbuchungsfirma bietet Buchungsmöglichkeiten rund um die Welt. Für jedes Zimmer bzw. Apartment, das durch die Firma gebucht wird, bekommt sie einen Anteil der Miete als Kommission. Deshalb beeinflusst die Wahrscheinlichkeit, dass ein Zimmer gebucht wird, direkt den Umsatz der Firma. Die Firma hat schon festgestellt, dass diese Wahrscheinlichkeit direkt proportional zur Qualität der Fotos ist, die den Kunden gezeigt werden. Dieser Qualitätsparameter ist aber subjektiv und hängt von den Kunden ab. Angenommen, dass viele Fotos zu einem Zimmer verfügbar sind, aber nur eines gezeigt werden kann, wie soll die Firma die Fotos auswählen, um zu lernen, welches Foto für jede ZImmer optimal ist, und dadurch den Umsatz zu optimieren?

Um die Leistung der Algorithmen zu messen, benutzen wir in den meisten Fällen die bekannte Standarddefinition von Regret als die Differenz zwischen dem gesamten Verlust des Algorithmus und dem Verlust einer Strategie die immer die beste Aktion in Nachhinein wählt. Für das Modell, wenn die Nutzenwerte von einer unbekannten aber bestimmten Wahrscheinlichkeitsverteilung gezogen werden, existieren Algorithmen, die eine obere Regretschranke von $O(\log T)$ liefern, wobei T die Anzahl der Zeitschritte ist. Wenn die Werte dagegen beliebig sind, d.h. ohne stochastische Annahme, dann gibt es Algorithmen, die eine bestmögliche obere Regretschranke von $O(\sqrt{T})$ liefern. Es gibt aber Fälle, in denen die Nutzenwerte nicht beliebig, sondern semi-strukturiert sind. Intuitiv sollte mehr Struktur in den Nutzenwerten bessere Regretschranken ermöglichen. In Anlehnung an einige der neueren Arbeiten definieren wir auch Modelle, die einen gewissen Grad an Struktur aufweisen. Das ist das Hauptthema in Kapitel 2.

Um die Struktur auszunutzen, benutzen wir *Trenderkennung* als Haupttechnik. Grob gesagt, wenn sich die Nutzenwerte der Aktionen deutlich verbessern, bzw. verschlechtern, wird eine Verschiebung des Wahrscheinlichkeitsgewichts auf die neue beste Aktion ausgelöst. Ein wichtiger Vorteil dieses Vorgehens ist, dass man damit eine obere Regretschranke in Bezug auf eine Strategie zeigen kann, die die beste Aktion in jedem Trend auswählt. Dies ist eine viel stärkere Garantie im Vergleich zur Standarddefinition. Genauer gesagt zeigen wir, dass auch in Bezug auf diesen stärkeren Benchmark eine Regretschranke von $O(\sqrt{T})$ erreichen werden können.

Kapitel 3 handelt von einer anderen Variante von Bandit Problemen, sogennanten kombinatorische Multi-Armed Bandit Probleme mit Rechen- und Wechselkosten. Das kombinatorische Multi-Armed Bandit Problem ist ähnlich zur klassischen Variante, außer dass der Agent jetzt nicht nur eine, sondern eine Menge von Aktionen wählen muss. Eine gültige Menge von Aktionen wird von dem Modell bestimmt. Dieser Ansatz war genau das Thema der Arbeit von Kveton et al. [1]. Wir betrachten eine Variante dieses Problems, in der dem Agenten Kosten entstehen, um eine Aktion zu berechnen und um eine gewählte Aktion zu ändern.

Eine Motivation sind Sensornetzwerke, in denen ein zentraler Agent einen minimalen Spannbaum für effiziente Kommunikation lernen will. Für dieses Beispiel sind die möglichen Spannbäume die Aktionen. Zur Berechnung der Aktion für den Fall des Spannbaumes, muss ein verteilter Algorithmus im Netzwerk ausgeführt werden. Diese Berechnung verbraucht jedoch Energie, die knapp ist, und deswegen entstehen Kosten dafür.

Preisgestaltung in wettbewerbsorientierten Märkten

Das Internet hat die Art und Weise, wie Waren gekauft und verkauft werden, revolutioniert. Dies hat eine Reihe von neuen Möglichkeiten eröffnet, den Preis der Waren strategisch und dynamisch zu bestimmen. Dies gilt insbesondere für Einzelhandels- und Bekleidungsgeschäfte im Internet, für die die Kosten und der Aufwand für die Preisaktualisierung vernachlässigbar geworden sind. Diese Flexibilität hat die Forschung in den letzten zehn Jahren zu einer dynamischen Preisgestaltung angetrieben. Meist geht es dabei um die Bestimmung optimaler Verkaufspreise in einer unbekannten Umgebung, um ein Ziel, normalerweise die Einnahmen, zu optimieren. In Verbindung mit dem Vorhandensein von digital verfügbaren und häufig aktualisierten Verkaufsdaten, kann dies auch als (Online-) Lernproblem angesehen werden.

Ausgehend von dieser Motivation konzentrieren wir uns im zweiten Teil der Arbeit auf die folgenden zwei Fragen: 1. Wie soll ein Verkäufer in einem wettbewerbsorientierten Markt den Preis für sein Gut bestimmen, um den Ertrag zu optimieren? 2. Wie wirken sich dynamische Marktparameter auf die Markträumung aus? Um unsere Antworten auf diese Fragen zu beschreiben, müssen wir zunächst ein generisches Marktmodell einführen. Der Markt besteht aus einer bestimmten Anzahl von Käufern und Verkäufern. Jeder Verkäufer bringt ein einzigartiges Gut zum Markt. Jedes Käufer hat sein eigenes Budget. Assoziiert mit jedem Käufer ist eine Nutzenfunktion, die den Wert einer Menge von Gütern bestimmt. In jedem Zeitschritt wählt jeder Verkäufer einen Preis für sein Gut. Basierend auf diesen Preisen und der Nutzenfunktion verlangt jeder Käufer eine Menge der Güter. Der Verkäufer beachtet die Nachfrage und aktualisiert den Preis seines Gutes im nächsten Zeitschritt, um seinen Ertrag zu verbessern. Die optimale Preis-Update-Strategie hängt von den Nutzenfunktionen ab

In Kapitel 6 untersuchen wir eine allgemeine Preisstrategie, die auf Algorithmen zur Regret Minimierung basiert. Dies ist eine Verallgemeinerung der Multi-Armed-Bandit-Algorithmen, die wir im ersten Abschnitt betrachtet haben, zu konvexen, bzw. konkaven Funktionen. Der Kern unserer Idee stammt aus einer Arbeit von Syrgkanis et al. [2]. Die Autoren beweisen, dass für ein Spiel mit mehreren Agenten, wenn jeder Agent

einen Regret-minimierenden Algorithmus mit einem geeigneten Schrittgrößenparameter verwendet, der eine bestimmte technische Eigenschaft erfüllt, dann ist der individuelle Regret jedes Agenten durch $O(T^{1/4})$ begrenzt, wobei T die Gesamtzahl der Runden ist. Das ist eine deutliche Verbesserung im Vergleich zu den Standardalgorithmen wie Online Gradient Descent [3], die eine Regretschranke von $O(T^{1/2})$ liefern.

Wir modellieren das dynamische Preisgestaltungproblem als ein Spiel, wobei die Verkäufer die Spieler sind, der Preis ihre Aktion ist, und der Ertrag ihr Nutzen ist. Es wird angenommen, dass die Nutzenfunktionen die IGS Eigenschaft erfüllen. Mit diesem Eigenschaft kann man den Umsatz als eine konkave Funktion darstellen. Es ermöglicht uns, die oben gennante Technik von Syrgkanis et al. zu verwenden. Man braucht jedoch noch mehrere andere Ideen, um eine scharfe Schranke auf den Umsatzverlust zu beweisen.

In Kapitel 5 konzentrieren wir uns auf einen anderen Preisaktualisierungsprozess, der "Tatonnement" genannt wird. In früheren Arbeiten wurde gezeigt, dass dieser Prozess zu einem Gleichgewicht konvergiert, d.h. zu Preisen, bei denen die Nachfrage für jedes Gut gleich dem Angebot ist. In diesem Kapitel zeigen wir, dass dieser Prozess für eine prominente Klasse von Nutzenfunktionen auch individuell für die Verkäufer rational ist und den Umsatz der Verkäufer optimiert.

Der Ansatz in diesem Kapitel unterscheidet sich Kapitel 6 auf zwei Arten: Hier konzentrieren wir uns auf bestimmte Klassen von Nutzenfunktionen, nämlich konstante Substitutionselastizität (auf Englisch, CES). Diese Klasse von Nutzenfunktionen wird oft in der Wirtschaftsforschung verwendet. Daneben zeigen wir in diesem Kapitel eine Obergrenze für den Ertragsverlust eines Verkäufers in Bezug auf eine Strategie, die in jedem Zeitschritt den Beste-Antwort-Preis wählt und dadurch ein stärkerer Benchmark ist.

Für das untersuchte Nutzenmodell zeigen wir, dass in einem statischen Markt, wenn jeder Verkäufer den Tatonnement-Prozess anwendet, der Ertragsverlust eines Verkäufers durch eine Konstante begrenzt ist. Wir untersuchen diesen Preisaktualisierungsprozess auch, wenn das Angebot der Verkäufer dynamisch ist. In diesem Fall zeigen wir, dass der gesamt Ertragsverlust eines Verkäufers mit zunehmender Instabilität Verfügbarkeit des Gutes steigt.

Tatonnement ist ein Preisaktualisierungsprozess, der für eine große Klasse von Märkten eine Konvergenz zum Gleichgewicht garantiert. Darüber hinaus rechtfertigt und erklärt er die in der Realität auftretenden Preisanpassungsprozesse. Die meisten der bisherigen Analysen der Tatonnement-basierten Dynamik gehen jedoch bisher davon aus, dass der Markt und seine Eigenschaften (z.B. Agenten, Budgets, Nutzenfunktionen, Angebot) im Laufe der Zeit unverändert bleiben. In dem letezten Kapitel untersuchen wir

die Wirksamkeit des Tatonnementprozesses hinsichtlich seiner Fähigkeit, ein annähernd des Marktgleichgewicht in dynamischen Märkten aufrechtzuerhalten. Zu diesem Zweck konzentrieren wir uns auf einen Fisher-Markt mit CES-Nutzenfunktion. Dieser Markt besteht aus der gleichen Gruppe von Verkäufern und Käufern, die sich jedoch einem ständigen Wandel in einem der Marktparameter unterziehen. Der Marktparameter kann z.B. Angebot, Käuferbudgets oder deren Nutzenfunktionen sein. Wir erweitern unsere Technik auf eine allgemeine Klasse von Lyapunov-dynamischen Systemen mit einem Update-Prozess, der die Lyapunov-Funktion in einer einzigen Runde multiplikativ verringert.

Zusammenfassend konzentrieren wir uns auf zwei breite und wichtige Klassen von Online-Lern- /Optimierungsproblemen. Im ersten Teil der Arbeit haben wir zwei Varianten von Multi-Armed-Bandit-Problemen untersucht, nämlich die klassische Variante mit Trenderkennung und die kombinatorische Variante mit Rechenkosten. Für das erstgenannte Problem zeigt sich, dass sogar einige schwache strukturelle Eigenschaften der Verluste genutzt werden können, um starke Regretschranken abzuleiten. Für das zweite Problem wird gezeigt, dass man trotz des Overheads der Rechenkosten fast genauso gute Regretschranken zeigen kann wie im klassischen Problem.

Der zweite Teil der Arbeit besteht aus zwei wichtigen Klassen von Preisstrategien für die Ertragsoptimierung, nämlich Algorithmen zur Regret-Minimierung und Tatonnement. Bei beiden Ansätzen zeigen sich starke Regretschranken auf den Ertragsverlust bei den Verkäufern in Abhängigkeit von den Nutzenfunktionen. Es wird weiterhin gezeigt, dass der Tatonnement-Prozess gegenüber sich ändernden Marktparametern robust ist und somit ein ungefähres Marktgleichgewicht aufrecht erhält.

Contents

List of Figures

Abbreviations

CES	**C**onstant **E**lasticity of **S**ubstitution
IGS	**I**so-elastic and **G**ross **S**ubstitutes
RVU	**R**egret bounded by **V**ariation in **U**tilities
OGD	**O**nline **G**radient **D**escent
OMD	**O**ptimistic **M**irror **D**escent
FTRL	**F**ollow **T**he-**R**egularized-**L**eader
DSR	**D**ynamic **S**tochastic **R**egime
ARG	**A**dversarial **R**egime with **G**ap
CMAB	**C**ombinatorial **M**ulti **A**rmed **B**andit

Chapter 1

Introduction

Online optimization consists of a class of problems where an agent interacts with the system in discrete time steps. In these time steps, it makes irrevocable decisions with the intention of optimizing an objective of interest. In contrast to a widely studied class of optimization problems, where the entire input is known beforehand, the agent in this model receives the inputs one-at-a-time. These class of problems, having been a subject of intense research for several decades now, have resulted in two analysis approaches: First is the *competitive analysis* where the performance of an algorithm is compared to that of an offline version, i.e., assuming all the information is available from the beginning. The second approach, which also is the subject matter of this thesis, abstracts the problem as a learning problem. Depending on the actual problem being studied, a variety of benchmarks are used to measure the algorithm performance.

While sequential learning has been an area of intense research in the last decades, several new problem domains have sprung up in recent years. For example, the problem of dynamic allocation of jobs to servers in data centres to balance performance and energy efficiency, or the problem of displaying relevant items to a user on an e-commerce website. In this thesis, we focus on two such problem classes. The first class of problems also referred to as *bandit problems*, refers to the class of problems, where the learner chooses an action (or set of actions) from a pool and observes the performance of these actions. The goal of the learner is to maximize the cumulative reward (respectively, minimize the cumulative loss). This basic model has been extended in several directions since its introduction capturing a variety of learning problems encountered in reality. In the first part, we also introduce two interesting variants and give algorithms for them.

The second class of problems we study focuses broadly on the different pricing strategies in the market. Under suitable market assumptions, we propose plausible explanations for the pricing behaviour observed in the real world. We study the impact of the pricing

methods on the revenue of sellers in the market. We find the results given to be particularly interesting since they incorporate elements of competition, a most commonly observed phenomenon in real markets, in the end-results. In the subsequent sections, we introduce each of these classes of problems in more detail.

1.1 Bandit Learning

Consider the following problem faced by a web-based hotel booking company called ABC.com. ABC hosts hotel listings from all over the world. For every room booked on their website, ABC receives a certain percentage of the rent as commission. Therefore, increasing the likelihood that a website visitor actually books a room has a direct impact on their own revenue. A recent survey conducted by ABC internally revealed that the likelihood that a visitor books a room is directly proportional to the *quality* of the property picture presented to the website visitor i.e. irrespective of the actual quality of the hotel room, the first few pictures that a visitor sees has a major impact on the likelihood of the sale. Note that this quality as mentioned above is a subjective measure and may differ across the website visitors. Even if that was not the case, since ABC hosts millions of listings on its website, manually choosing the "best" picture for every property listed is not possible. How could ABC go about optimizing its sale probability and thereby its revenue?

This problem is a representative example of a class of sequential learning problems, also referred to as "bandit learning problems". These differ from classical, or machine learning-style optimization methods, in that the learner does not have access to"batch" or historical data and needs to optimize in real-time with the limited feedback that the learner has access to. Problems of this flavour have been a subject of intense study under a variety of models, and continue to be so even today. The classical version of this problem is modelled as a sequence of "bandit machines" where pulling the handle (or arm) of such a machine results in a certain reward. The goal of the learner is to maximize the cumulative reward. The reward observed on pulling the handle of a given machine can either be stochastic or adversarial depending on the model under study.

Although the problem description is straightforward in this abstract model, it allows one to construct theoretically clean models for more complex problems, for example, the one introduced above. For example, the problem of ABC may be modelled as follows: The potential customers approach the booking platform sequentially, and for each of these customers, depending on the hotel s/he searches, ABC chooses a certain set of pictures to display. Here we are implicitly assuming that for every listing, there exists a pool of pictures from which the server chooses a small set. The user perception based

on which the customer makes a decision is a stochastic variable. Therefore, for a given set of pictures displayed, there exists a certain (unknown) probability of the property being booked. The problem of optimizing the sale probability can thus be posed as the problem of finding the set of display pictures that maximizes the sale probability with respect to the unknown stochastic process.

A characteristic feature of most algorithms for online optimization is the exploration-exploitation trade-off. For example, if the rewards of the chosen action are drawn from a probability distribution, then choosing the action with maximum expected reward is clearly the optimal strategy. But without any prior knowledge of the underlying reward distribution, any algorithm is forced to *try-out* several different actions at random, until the optimal action can be clearly identified. Such random trials are often referred to in the literature as *exploration* phases. By definition, there exist no guarantees on the rewards achieved in this period. To be able to prove guarantees, the algorithm has to, at some point, use the feedback gathered in these exploration phases, to choose an action that is optimal with respect to this feedback. These are often referred to as *exploitation* phases. The optimal algorithms for such problems usually involve a clever interleaving of exploration and exploitation phases.

1.2 Online Pricing in Markets

Consider now a different scenario, one of a vegetable market in a small town. This market consists of sellers, each bringing some quantity of a vegetable every day to the market. For simplicity, suppose that each seller brings a single variety to the market. The people of the town, also the buyers, have their own individual preferences over the vegetables. For example, one particular individual may like or dislike one or more varieties of vegetables over the others. For purposes of modelling, assume that these preferences can be completely captured by some closed-form expressions. We refer to them as the *utility function* of the buyer. Furthermore, depending on the needs of the buyer, or perhaps depending on her financial capability, the buyer decides to spend a certain amount of money in a certain time interval. One may simply model this as the buyer's private budget. Based on the utility function and the private budget, which may differ across buyers, the buying decisions are made. The goal of the sellers is to price their vegetable such that their revenue is maximized.

This problem is an example of online optimization in the presence of strategic agents, or simply, competition. Models of this nature have only recently started to receive attention, particularly in the computer science community. Note that one of the primary differences from the model described in the previous section is that the reward observed

by any given agent for any given action depends on the choices of all other agents. For example, one would expect the demand observed by a certain seller for his vegetable to increase if the price of some *other* vegetable increases, causing people to shift to a more affordable option. The degree of such a shift in demand naturally depends on the inherent preferences of the buyers.

This dependency of the revenue obtained by a given seller on the prices chosen by other sellers makes this problem particularly challenging. Furthermore, changing the dependency relationship changes the entire problem and hence necessitates a new approach. In this thesis, we study the pricing problem in markets where the buyer utilities belong to a general class, namely the CES utilities. Fixing such a utility function on part of the buyers essentially defines the dependency relationship mentioned above and lends additional structure to the pricing problem. It is this structure that allows one to design more efficient algorithms than for the general online optimization problem described in the previous section. For example, as mentioned before, any learning algorithm in the general case has to adopt the exploration-exploitation strategy. With the additional structure, the learner is able to forego the purely random try-outs and adopt a fixed iterative strategy.

Our approach to design these algorithms rely on previous work in the theory of market equilibrium. It is postulated in economic theory that large repeated markets often operate close to equilibrium. In the second part of this thesis, we establish connections between the problem of revenue maximization of sellers in a market and that of distributed computation of market equilibrium for a prominent class of markets. For this class, we provide an alternative justification of the existence of market equilibrium as a result of the sellers optimizing their own revenue.

1.3 Thesis Overview

As mentioned before, in this thesis we focus on two broad classes of online optimization problems, namely *Bandit Learning* and *Pricing in Markets with Gross Substitutes Utilities*, which also form its two main parts. In the first part, we focus on two specific bandit learning problems. In Chapter 2, based on joint work with Rebecca Reiffenhäuser, we investigate two structured loss models and give algorithms that take advantage of this additional structure to obtain better learning guarantees. In Chapter 3, we explore a model commonly encountered in decentralized learning systems where computation of the new action to be taken is too expensive to be done every round. For this model, we show that in spite of the additional computation costs, one can achieve learning guarantees which are almost as good as the classical model.

The second part of the thesis focuses exclusively on the problem of pricing in markets. In Chapter 5, based on joint work with Martin Hoefer, we study an existing price adaptation strategy called tatonnement, and show that for a certain class of markets this strategy also optimizes seller revenue in a competitive market. In addition, we provide concrete bounds on the loss in revenue incurred by any seller in static and dynamic markets. In chapter 6, we continue this study but for markets with different classes of buyer utility. For this more general class of strategies studied here, we give bounds on losses in revenue of the seller with respect to suitable benchmarks. In Chapter 7, based on joint work with Yun Kuen Cheung and Martin Hoefer, we shift our focus to questions concerning the convergence properties of the tatonnement process in markets, when parameters like supply and buyer budget are subject to perturbation. The resulting analysis and conclusion is then extended to a broad class of Lyapunov dynamical systems.

1.4 List of Papers

Most of the results in this thesis are taken from a series of manuscripts, listed below for completeness.

List of Papers:

[1] Martin Hoefer and Paresh Nakhe. Revenue optimization via tatonnement in fisher markets. Manuscript under submission, 2018.

[2] Yun Kuen Cheung, Martin Hoefer, and Paresh Nakhe. Tracing equilibrium in dynamic markets via distributed adaptation. Manuscript under submission, 2018.

[3] Paresh Nakhe. Dynamic pricing in competitive markets. In *International Conference on Web and Internet Economics (WINE)*, pages 354–367. Springer, 2017.

[4] Paresh Nakhe and Rebecca Reiffenhäuser. Trend detection based regret minimization for bandit problems. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 263–271. IEEE, 2016.

Part I

Bandit Learning

Chapter 2

Learning via Trend Detection

2.1 Introduction

Consider the following problem: Suppose you own an apparel store and have purchased a fixed number of ad slots on some website, say, Facebook. Every time someone visits the website, you can choose a set of ad impressions to display. Let's assume that an ad here consists of an image of a clothing item and that each image is associated with a click-through-rate unknown to you. Your goal is to choose images to display such that the cumulative click-through-rate is maximized. How would you choose these images? This problem falls into the domain of reinforcement learning and, more specifically, multi-armed bandit learning. Contrary to supervised learning (and most of current research in statistical pattern recognition and artificial neural networks), multi-armed bandit learning is characterized by its *interactive nature* between an agent and an uncertain environment. Such a learning algorithm makes its next move based on the history of its past decisions and their outcomes.

More specifically, a multi-armed bandit problem is a sequential learning problem where the learner chooses an action from a set of actions in every round. Associated with each action is a loss unknown to the learner[1]. The goal of the learner is to minimize the loss incurred. Performance of the learning algorithm is measured by regret, compared to a certain benchmark strategy. Conventionally, in multi-armed bandit problems the benchmark strategy is to always choose the single best action in hindsight, i.e., an action with minimum cumulative loss. This problem has been thoroughly studied in a variety of settings [4–7]. A distinguishing feature of such problems is the inherent exploration-exploitation trade-off. When the losses are generated from a fixed but unknown distribution, there exist algorithms [5, 7, 8] that can achieve a regret guarantee

[1]The case with rewards is symmetric.

of $O(\log T)$, where T denotes the number of rounds. On the other hand, when losses for the actions are generated under no statistical assumption, or alternately when losses are generated by an adversary, the best possible regret guarantee that can be achieved is $O(\sqrt{T})$ [6]. Recently, interest has been developing [9, 10] in the question of achieving non-trivial regret guarantees when the loss model is semi-structured. Intuitively, more structure in the losses should enable more exploitation and hence allow for better regret guarantees. Along the lines of some of the recent work [9], we also define models exhibiting a certain degree of structure.

Often the real-world problems do not exhibit adversarial behaviour, and in many cases the losses of different actions follow a trend structure, i.e. when one action is consistently better than others in a certain interval. For such more specialized models, the standard techniques prove insufficient since they do not take advantage of these properties. In this paper, we address this deficiency using the paradigm of trend detection. Broadly, we propose a strategy that keeps track of the current trend and restarts the regret minimization algorithm whenever a trend change is detected. This allows us to give regret guarantees with respect to a strategy that chooses the best action in each trend. This is a significantly stronger benchmark than the one conventionally considered. The regret guarantee with respect to this benchmark is also called switching regret.

More importantly, our proposed strategy is not specific to a particular regret minimization algorithm unlike the approaches in some recent works [11]. In this paper, we use Exp3 as the underlying regret minimizing algorithm for its simplicity and almost optimal regret guarantee [4]. However, one can use any other algorithm and analyze it in a similar way. Because of this modular structure of the algorithm, we can extend the arguments and proofs for the conventional multi-armed bandits problem to a more general setting where instead of a single action, the learner chooses multiple actions in each round [12]. This problem has been studied in stochastic [13] and adversarial [14] settings, but to the best of our knowledge, there are no prior works giving a switching regret guarantee for it.

One of the primary motivations for studying these bandit problems comes from the domain of recommender systems. Many web tasks such as ad serving and recommendations in e-commerce systems can be modeled as bandit problems. In these problems, the system only gets feedback for the actions chosen, for example whether the user selects the recommended items or not. Notice that these systems may recommend one or more items in each round. The trend detection paradigm used in this chapter is motivated by the observation that in many cases, the performance of actions follow a trend structure. In the above mentioned case of an apparel store, for example, swimsuits may be the

better choice during the warm parts of the year, or perhaps what is currently in vogue in popular fashion.

Related Work

The problem of giving regret guarantees with respect to a switching strategy has been considered previously in several works (albeit in more restricted settings), all of which consider the case when the learner chooses exactly one action in each round. Auer et al. proposed Exp3.S [4] along the same lines as Exp3 by choosing an appropriate regularization factor for the forecaster. This enables the algorithm to quickly shift focus on to better performing actions. For an abruptly changing stochastic model, Discounted-UCB[15] and SW-UCB [16] have been proposed along the lines of UCB. In the former algorithm, a switching regret bound is achieved by progressively giving less importance to old losses while in SW-UCB, the authors achieve the same by considering a fixed size sliding window. Both these algorithms achieve a regret bound of $O(\sqrt{MT \log T})$, where M is the number of times the distribution changes and T denotes the number of rounds.

Our work is closest to the algorithm Exp3.R proposed by Feraud et al. [11] who also follow a paradigm very similar to trend detection. However, their algorithm is specific to Exp3 and applies only to the version of the bandit problem where one chooses a single action in each round. Furthermore, their algorithm assumes a certain gap in the performance of actions that depends on the knowledge of run time of the algorithm. This makes it inapplicable to a number of real-world scenarios.

The trend detection idea used in our algorithm is similar to the change detection problem studied in statistical analysis. Similar ideas have also been used for detection of concept drift in online classification [17, 18]. Common applications include fraud detection, weather prediction, and advertising. In this context, the statistical properties of a target variable change over time, and the system tries to detect this change and learn the new parameters.

Overview: We start by introducing the basic model in Section 2.2 and the two main loss structures we investigate in subsequent sections. For the standard K-armed bandit problem, we propose a new algorithm called Exp3.T in Section 2.3. This algorithm guarantees switching regret of $\tilde{O}\left(\frac{N\sqrt{TK}}{\Delta_{sp}}\right)$ where N is the number of trend changes and not known to the learner. Δ_{sp} indicates the degree of structure in loss model. This regret bound is proved in Section 2.4. This guarantee extends to the anytime setting i.e. when the duration of the run, T, is not known in advance. In Section 2.5, the analysis is further extended to the case when instead of a single action the learner chooses K actions in each round. The underlying regret minimization algorithm used in this case is

OSMD [14]. The resulting algorithm achieves switching regret of $\tilde{O}\left(\frac{Nm\sqrt{TK}}{\Delta_{sp}}\right)$. Finally, in Section 2.6, we provide empirical evidence for this algorithm's performance in the classical setting. To sum up, in comparison to the state-of-the-art algorithms, we show that our algorithms are particularly effective when the structure of the losses encountered satisfy some weak assumptions.

2.2 Model and Preliminaries

We consider a multi-armed bandit problem with losses for K distinct actions. The learner chooses one of the K actions sequentially for T rounds. Let the set of these K actions be denoted by $[K]$. The losses of these K actions can be represented by a sequence of loss vectors $\{\mathbf{x}\}_{t=1}^{T}$ where $x = \{(x_1, x_2 \cdots x_K)\}_t$. The loss sequence is divided into N *trends* of variable lengths. Their starting rounds are given by $\{T_n\}_{n=1}^{N}$ and are unknown to the learner. A trend is defined as a sequence of rounds where a set S of m actions is *significantly* better than others for the duration of this trend. We say that the trend has changed when this set of actions changes. Within each trend the losses of actions in the set S are "separated" from all others by a certain gap. Particularly, we consider a finer characterization of loss models than just stochastic or adversarial within a trend. Similar to the loss model introduced by Seldin et al [9], we focus on models exhibiting a "gap" in losses. Although this model is weaker than the adversarial model it still covers a large class of possible loss models. We express the gap in our loss models by an abstract term Δ_{sp}, the separation parameter. Although the exact definition of this parameter changes depending on the actual model, in each case it conveys the same idea that a larger value of this parameter implies a larger gap between losses of actions in S and every other action.

1. **Dynamic Stochastic Regime (DSR)**: For the stochastic loss model, the loss of each action a at round t is drawn from an unknown distribution with mean μ_t^a. Let a^* and a be any actions in sets S and $[K]-S$ respectively. Then for all rounds t in trend τ, $\mu_t^{a^*} < \mu_t^a$ and the separation parameter is defined as:

$$\Delta_{sp}(\tau) = \min_{t\in\tau}\{\mu_t^a - \mu_t^{a^*}\}.$$

   The loss model is stochastic with separation parameter $\Delta_{sp} = \min_{\tau} \Delta_{sp}(\tau) > 0$. The identity of best action a^* changes N times.

2. **Adversarial Regime with Gap (ARG)**: We use a modified version of the loss model introduced in [9]. Within each trend τ, there exists a set S of m actions

which is the best set for any interval of (sufficiently large) constant size, C. More precisely, let $\lambda_z(a) = \sum_{t \in z} \ell_{a,t}$ be the cumulative loss of an action a in interval z consisting of C rounds. Then for any action $a^* \in S$ and $a \in [K] - S$ we define the separation parameter for trend τ as:

$$\Delta_{sp}(\tau) = \min_{z \in \tau} \left\{ \frac{\min_{a' \neq a^*} \lambda_z(a') - \lambda_z(a^*)}{|z|} \right\}$$

It is the smallest average gap between any sub-optimal action and any action in set S for any interval z of size C. As in the above model, we say that a model satisfies the ARG property with separation parameter Δ_{sp} when $\Delta_{sp} = \min_{\tau} \Delta_{sp}(\tau) > 0$.

Assumption: For the algorithm considered in this chapter, we assume that the loss model, either stochastic or adversarial regime with gap, has separation parameter lower bounded by 4Δ, a constant known to us i.e. $\Delta_{sp} \geq 4\Delta$.

Notice that in the first trend, spanning from the first round till some round n, each action satisfies the gap conditions defined above for all the constituent rounds (DSR) or intervals of size C (ARG), for the respective setting. We define n to be the last such round, i.e., these conditions are violated at round $n + 1$, indicating the start of a new trend.

We study two variants of this problem. In the first variant, the algorithm chooses exactly one action every round while in the other, the algorithm can choose any set of m actions. For both the variants, the algorithm observes losses only of the actions chosen (or the single action chosen for the former variant). We assume the presence of an oblivious adversary which decides on the exact loss sequences before the start of the game. The sequence is of course not known to the algorithm. We also make the standard assumption that losses come from the $[0, 1]$ interval.

For the problem setting as described, our goal is to design an algorithm $\mathcal{A}$ to minimize the cumulative loss incurred in the T rounds that the game is played. For the case when the algorithm chooses exactly one action every round, its performance is measured with respect to a strategy that chooses the best action in each trend. Specifically, let I_t denote the action chosen by the algorithm in round t and let $X^t_{I_t}$ denote the corresponding loss incurred by this action. Then the cumulative loss incurred by the algorithm is:

$$L_{\mathcal{A}} = \sum_{t=1}^{T} X^t_{I_t}.$$

For ease of notation, we denote the rounds in trend n, i.e., $[T_n, T_{n+1} - 1]$ by $[n]$. Let $I^*_{[n]}$ be the best action in trend n, then the loss incurred by the switching strategy described above is:

$$L^* = \sum_{n=1}^{N} \sum_{t=T_n}^{T_{n+1}-1} X^t_{I^*_{[n]}},$$

where trend n occurs in the interval $[T_n, T_{n+1} - 1]$. We define the regret incurred by algorithm $\mathcal{A}$ as follows:

$$R^*_T = L_{\mathcal{A}} - L^*.$$

Exactly analogous definitions apply to the case when the algorithm chooses multiple actions in each round.

2.3 The Exp3.T Algorithm

Algorithm 1 Exp3 [19]

▷ Parameter: a non-increasing sequence of real numbers η_t
Let p_1 be the uniform distribution over $1, ..., K$.
for each interval round $t = 1, 2, \ldots T$ **do**
    Choose an arm I_t from distribution p_t
    for each arm $i = 1, \cdots K$ **do**
        $\tilde{\ell}_{i,t} = \frac{\ell_{i,t}}{p_{i,t}} \mathbb{1}_{I_t = i}$
        $\tilde{L}_{i,t} = \tilde{L}_{i,t-1} + \tilde{\ell}_{i,t}$
    end for
    $p_{i,t+1} = \frac{\exp(-\eta_t \tilde{L}_{i,t})}{\sum_{k=1}^{K} \exp(-\eta_t \tilde{L}_{k,t})}$
end for

The algorithm Exp3.T is composed of two governing ideas: The Exp3 algorithm and a trend detection routine. Exp3 (see algorithm 1) gives almost optimal regret bounds with respect to the single best action in hindsight when the loss model is adversarial. However, when the losses exhibit a certain structure or when regret with respect to a stronger benchmark is desired, Exp3 proves to be insufficient. In our algorithm, we overcome this problem by identifying *trends* in losses and resetting the Exp3 algorithm whenever a change in trend is detected. One advantage of using Exp3 in settings exhibiting structured losses is that it is robust to changes in the losses of actions as long as the best action remains same. We exploit this property in our algorithm so that it is applicable to a large class of loss models. In the analysis we use the following regret bound given by [19].

Lemma 2.1. *For any non-increasing sequence $\{\eta\}_{t\in\mathbb{N}}$, the regret of the Exp3 algorithm with K actions satisfies*

$$R_T \;\leq\; \frac{K}{2}\sum_{t=1}^{T}\eta_t \;+\; \frac{\ln K}{\eta_T}.$$

Algorithm 2 shows the skeleton of the procedure to achieve the desired bound on the switching regret. At a high level, the algorithm divides the total run into runs on smaller intervals. Within each interval the algorithm runs Exp3 (parameter η) with loss monitoring(LM) plays randomly interspersed among all rounds. The length of this interval is controlled by parameter γ. These loss monitoring plays choose different actions for a fixed number of rounds without regards to regret. The loss values collected from this process are used to give an estimation of the mean loss of each action in a given interval. The number of such plays required to give a good estimation of loss depends on the actual model under consideration and is captured by the parameter t^*. Based on this estimation, the trend detection module outputs with probability at least $1-\delta$ whether the best action has changed or not, alternatively whether the trend has changed or not.

The *Make_Schedule*$(\cdot)$ procedure assigns Exp3 plays and fixed action plays to monitor loss (exactly t^* many per action) randomly to rounds at the start of an interval and returns the randomly generated schedule. The random generation of schedule protects the algorithm from making biased estimates of actual losses.

Algorithm 2 Exp3.T

```
▷ Parameters: δ, γ and η
Set interval length |I| = Kt*/γ
for each interval I do
    Schedule ← Make_Schedule(I)
    for t = 1, 2 ··· |I| do
        if Schedule(t) = Exp3 Play then
            Call Exp3_play()
        else
            Call LM_play(Schedule(t))
        end if
    end for
    if trendDetection() == True then
        Restart Exp3
    end if
end for
```

Trend Detection

In any interval, the loss monitoring component of Algorithm 2 chooses each action a sufficient number of times, and these choices are randomly distributed over the interval. The samples obtained from these plays are used to give a bound on the deviation of the empirical mean of losses from the true mean. Particularly, we use the following lemma by Hoeffding [20] for sampling without replacement from a finite population.

Lemma 2.2. *Let $\mathcal{X} = (x_1, x_2, \cdots x_N)$ be a finite population of N real points from $[0, 1]$, $X_1, X_2 \cdots X_n$ denote random sample without replacement from $\mathcal{X}$. Then, for all $\epsilon > 0$,*

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

where $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$ is the mean of $\mathcal{X}$.

For each interval we maintain information about the empirical mean of losses for each action, i.e., the mean over loss values actually seen by the algorithm. By Lemma 2.2, all of these estimates are close to the actual mean with probability at least $1 - \delta$, where δ is a parameter of the algorithm. In the case of change in trend within an interval I, these guarantees are of course void as the losses do not maintain a uniform pattern. Therefore, a change in trend can be detected by comparing the empirical estimates obtained at the end of the next interval to those obtained prior to the trend change. This idea is represented in Algorithm 3.

Algorithm 3 trendDetection()

```
Let p be the index of the current interval
I*_p ← action with minimum empirical mean loss, μ̂, in interval p.
if p = 1 or p = 2 then
    return False
end if
if I*_p ≠ I*_{p−2} then
    return True
end if
return False
```

2.4 Regret Analysis

For ease of notation in the analysis, we define the *detector complexity* t^* as the number of loss monitoring samples required for each action so that the trend detection procedure works with probability at least $1 - \delta$, provided there is no trend change in the actual

interval. In what follows, we give detector complexity bounds for different models and use it as an abstract parameter in regret analysis.

Lemma 2.3. *The detector complexity in dynamic stochastic regime satisfies*

$$t^*_{DSR} \;=\; \frac{1}{2\Delta^2}\ln\left(\frac{4K}{\delta}\right).$$

Proof. Fix an action a and an interval I. Let the expected reward of action a on interval I be given by the sequence $\{\mu^a_t\}_{t\in I}$ and the actual realization of rewards be given by $\{X^a_t\}_{t\in I}$. First we observe that the expected reward of a over the interval I is given by

$$\mu_{a,I} \;=\; \frac{\sum_{t\in I}\mu^a_t}{|I|}.$$

Let the set of loss monitoring samples collected by our algorithm for action a be denoted by $\mathcal{Z}_a$. The algorithm uses these samples to calculate the empirical mean of rewards for the action a. We denote it by $\hat{\mu}_{\mathcal{Z}_a}$.

Step 1: First we show that the empirical mean of losses over the entire interval is close to the expected mean, $\mu_{a,I}$. Let $\{X^a_t\}_{t\in I}$ be the sequence of actual reward realizations for arm a in interval I. Denote by $\bar{\mu}_{a,I}$ the mean of these actual realizations. Applying Hoeffding's inequality,

$$\begin{aligned} P(|\mu_{a,I}-\bar{\mu}_{a,I}|>\Delta) \;&\le\; 2\exp(-2|I|\cdot\Delta^2) \\ &\le\; 2\exp(-2t^*_{DSR}\cdot\Delta^2)=\frac{\delta}{2K}, \end{aligned}$$

i.e., the empirical mean of losses for action a over the interval I is close to the actual mean with probability at least $1-\frac{\delta}{2K}$.

Step 2: Now we show that the empirical mean of loss-monitoring samples collected for action a is close to the mean of the actual realizations, $\bar{\mu}_{a,I}$. This follows from Lemma 2.2:

$$P(|\bar{\mu}_{a,I}-\hat{\mu}_{\mathcal{Z}_a}|>\Delta) \;\le\; 2\exp(-2t^*_{DSR}\Delta^2) \;=\; \frac{\delta}{2K}.$$

Therefore, with probability at least $1-\frac{\delta}{K}$, the mean of loss monitoring samples for any action is within 2Δ of the actual mean. By applying a union bound over all actions, with probability at least $1-\delta$ the same guarantee holds over all actions, which in turn implies that the trend detection module can detect whether the best action has changed with the same probability. □

Lemma 2.4. *The detector complexity in the adversarial regime with gap satisfies*

$$t^*_{ARG} \;\geq\; \frac{(b-a)^2}{8\Delta^2} \ln\left(\frac{2K}{\delta}\right)$$

when the losses in the given trend are drawn from interval $[a, b]$.

Proof. The proof for this lemma goes along the same lines as for Lemma 2.3 except that in this case we do not need step 1. Further, in this case, we can allow the empirical mean of collected samples to be within 2Δ of the actual mean of all losses in the interval instead of just Δ. For this particular loss model, if additional information about the range of losses within a trend is available, then using the generalized version of Hoeffding's inequality we achieve a tighter detector complexity bound. Unless defined otherwise, our losses are always drawn from the range $[0, 1]$.

□

In the rest of the analysis, we use the model-oblivious parameter t^* to represent t^*_{DSR} or t^*_{ARG}.

Theorem 2.5. *The expected regret of Exp3.T is*

$$R_T \;=\; O\left(\frac{N\sqrt{(TK\ln K)\ln(TK\ln K)}}{\Delta_{sp}}\right).$$

Proof. We divide the regret incurred by Exp3.T in three distinct components; the first is the regret incurred just by running and restarting of Exp3. To bound this component of total regret we use the regret bound as in Lemma 2.1. Let $F(T)$ denote the number of *false trend detections,* i.e., the number of times when there was no change in detection but the detection algorithm still indicated a change. Then the regret incurred due to Exp3 is

$$R_{Exp3} \;\leq\; \frac{K}{2}\sum_{t=1}^{T}\eta_t + \frac{(N-1+F(T))\ln K}{\eta_T}.$$

As trend detection fails with probability at most δ, the expected number of false detections is at most

$$F(T) \;\leq\; \delta\left(\frac{T}{|I|}+1\right).$$

The second component of the total regret is on account of intervals wasted due to delay in detection of trend change. Specifically, if the trend changes in a given interval I, the regret guarantee obtained as part of Exp3 is not with respect to the best action before and after trend change. As we cannot give the required guarantee for this interval, we

count this interval as *wasted* and account it towards regret. Secondly, since the trend detection algorithm detects the change with probability at least $1-\delta$, the expected number of trend detection calls required (or alternatively the expected number of intervals) is at most $\frac{1}{1-\delta}$. Therefore, the total number of wasted rounds is at most

$$R_{wasted} \;\leq\; N\left(1+\frac{1}{1-\delta}\right)|I|.$$

The third and final component of regret incurred is due to the *loss monitoring plays* in each interval. No guarantee can be given about the regret incurred in these rounds and hence all such rounds are also accounted in regret. Since in each interval there are exactly Kt^* number of such plays, the total number of such rounds is at most

$$R_{loss\_monitor} \;\leq\; Kt^*\left(\frac{T}{|I|}+1\right) \;=\; \gamma T + Kt^*.$$

Putting all together, the total regret is

$$\begin{aligned} R_T \;\leq\; & K\sum_{t=1}^{T}\eta_t + \frac{(N-1+\frac{\gamma\delta T}{Kt^*})\ln K}{\eta_T} \\ & + N\left(1+\frac{1}{1-\delta}\right)\frac{Kt^*}{\gamma} + \gamma T + Kt^*. \end{aligned}$$

Setting $\eta = \sqrt{\frac{\ln K}{TK}}$, $\gamma = \sqrt{\frac{Kt^*\ln K}{T}}$ and $\delta = \sqrt{\frac{K}{T\ln K}}$, the regret incurred by Exp3.T is

$$\begin{aligned} R_T \;\leq\; & \sqrt{TK\ln K} \;+\; N\sqrt{TK\ln K} \;+\; \sqrt{\frac{TK\ln K}{t^*}} \;+\; 2N\sqrt{\frac{TKt^*}{\ln K}} \\ & +\quad 2N\frac{K\sqrt{t^*}}{\ln K} \quad+\quad \sqrt{t^*TK\ln K} \quad+\quad Kt^*, \end{aligned}$$

where $t^* = O\left(\frac{\ln(TK\ln K)}{\Delta_{sp}^2}\right)$. Alternatively, $R_T = O\left(\frac{N\sqrt{(TK\ln K)\ln(TK\ln K)}}{\Delta_{sp}}\right)$. □

Extension to the Anytime Version

The parameters derived to achieve the desired regret bound in Theorem 2.5 depend on the knowledge of T, the length of the total run of the algorithm. This dependency can be circumvented by using a standard doubling trick. Particularly, we can divide the total time into periods of increasing size and run the original algorithm in each period. Since the guarantee of this algorithm rests crucially on the probability of correct trend detection, in our case we need to modify the δ parameter as well.

Algorithm 4 Anytime Exp3.T

▷ Choose an initial estimate T' of length of run
for $i = 0, 1, 2 \cdots$ **do**
    Let $T_i = 2^i T'$
    Set $\gamma_i = \sqrt{\frac{K t_i^* \ln K}{T_i}}$, $\delta_i = \frac{1}{T_i^{3/2}} \sqrt{\frac{K}{\ln K}}$
    Run Exp3.T with parameters γ_i, δ_i in period T_i
end for

Theorem 2.6. *The expected regret of Anytime Exp3.T with* $\eta_i = \sqrt{\frac{\ln K}{T_i K}}$, $\gamma_i = \sqrt{\frac{K t_i^* \ln K}{T_i}}$ *and* $\delta_i = \frac{1}{T_i^{3/2}} \sqrt{\frac{K}{\ln K}}$ *is* $O\left(\frac{N\sqrt{(TK \ln K) \ln(TK \ln K)}}{\Delta_{sp}}\right)$.

Proof. We follow the same steps as in the proof of Theorem 2.5. We divide the regret incurred into three different components: regret due to Exp3 algorithm, due to the wasted intervals during detection and due to the loss monitoring plays. Compared to the proof of Theorem 2.5 the only difference is that here we have to sum the regret of Exp3.T over multiple runs. If T is the actual length of play, then the number of times we run Exp3.T is at most $\log T$. The regret due to the Exp3 algorithm (running and restarting) is:

$$R_{Exp3} \leq \sum_{i=0}^{\lceil \log T \rceil} \left(\frac{K}{2} T_i \eta_i + \frac{(N_i - 1 + F(T_i)) \ln K}{\eta_i} \right),$$

where N_i and $F(T_i)$ are the number of changes in trend and number of false detections in ith run of Exp3.T respectively. As before,

$$\begin{aligned} F(T_i) &\leq \delta_i \left(\frac{T_i}{|I|_i} + 1 \right) \\ &= \frac{1}{T_i^{3/2}} \sqrt{\frac{K}{\ln K}} \cdot \left(\frac{T_i}{K t_i^*} \sqrt{\frac{K t_i^* \ln K}{T_i}} + 1 \right) \leq \frac{2}{T_i}. \end{aligned}$$

Using this bound in the above inequality

$$\begin{aligned} R_{Exp3} &\leq \sum_{i=0}^{\lceil \log T \rceil} \left[\frac{K T_i \eta_i}{2} + \frac{N \ln K}{\eta_i} + \frac{2 \ln K}{T_i \eta_i} \right] \\ &\leq \sqrt{K \ln K} \cdot \sum_{i}^{\lceil \log T \rceil} \left(\frac{\sqrt{T_i}}{2} + N\sqrt{T_i} + \frac{2}{\sqrt{T_i}} \right) \\ &\leq C_1 \left(\sqrt{TK \ln K} + N\sqrt{TK \ln K} \right). \end{aligned}$$

The inequalities follow by using parameters η_i and δ_i as defined in the algorithm. For ease of representation, we capture all constants with a single constant C_1. The regret incurred due to wasted intervals is:

$$\begin{aligned} R_{wasted} &\leq \sum_{i=0}^{\lceil \log T \rceil} N_i \left(1 + \frac{1}{1-\delta_i}\right) |I_i| \\ &\leq \sum_{i=0}^{\lceil \log T \rceil} 2N\left(1+\delta_i\right) \frac{Kt_i^*}{\gamma_i} \\ &\leq \sum_{i=0}^{\lceil \log T \rceil} \frac{4NKt_i^*}{\gamma_i} \\ &\leq \sum_{i=0}^{\lceil \log T \rceil} N\sqrt{\frac{t_i^* T_i K}{\ln K}} \leq C_2 \cdot \left(N\sqrt{\frac{TKt^*}{\ln K}}\right) \end{aligned}$$

Here we use the fact that $t_i^* = O(t^*)$, the detector complexity had we known T beforehand. All the constants involved in the above inequality are captured by C_2. Similarly, the regret due to loss monitoring plays is:

$$\begin{aligned} R_{loss\_monitor} &\leq K \sum_{i=0}^{\lceil \log T \rceil} t_i^* \frac{T_i}{|I_i|} \\ &\leq \sum_{i=0}^{\lceil \log T \rceil} \gamma_i T_i \\ &\leq C_3 \cdot \left(\sqrt{KTt^* \ln K}\right), \end{aligned}$$

where the constant C_3 captures the constants involved. Combining the above mentioned bounds we get the desired claim. This bound is only a constant factor worse than the bound proved in Theorem 2.5.

It is easy to verify that the above analysis holds if δ_i is of the order of δ. This condition is met when T' is of order at least $T^{\frac{1}{3}}$. If, however, T' is not a good estimate of T in the above sense, the output of the trend detection procedure in initial runs will not be correct with sufficiently high probability and hence the aforementioned guarantees do not hold. We account for the regret incurred in the first few runs (till $T_i \geq T^{\frac{1}{3}}$) by simply disregarding all of them and consider them as *wasted* rounds. □

The principle of trend detection and restarting of a base algorithm (Exp3 in our context) according to changes in the trend can be extended to any multi-armed bandit algorithm for the adversarial setting. The final regret guarantee obtained naturally depends on

the performance of the base algorithm. We notice, however, that due to the necessary number of exploration rounds, no base algorithm can allow us to achieve regret $o(\sqrt{T})$. In particular, by choosing an appropriate base algorithm, our framework can be adjusted to a number of different loss structures and problem settings. In the following section, we use exactly this principle to design an algorithm to minimize regret with respect to the m best actions.

2.5 Extension to Top-m Actions

In this section, we extend the ideas introduced above to a setting where in each round we choose $m > 1$ actions out of the K available. For this variant of the problem, the Exp3 algorithm cannot be used. Instead we use the more general approach proposed by Audibert et al. [14]. This approach called Online Stochastic Mirror Descent (OSMD) is based on a powerful generalization of gradient descent for sequential decision problems. Similar to Exp3, the regret bound obtained by this technique is with respect to the best combination of actions in hindsight and holds even for adversarial losses. We refer the reader to [19] for a thorough treatment of the technique. In our proposed algorithm, OSMD.T, we apply this technique as a black box and only use the final regret bound.

Lemma 2.7. *The regret of the OSMD algorithm in the m-set setting with $F(x) = \sum_{i=1}^{K} x_i \log x_i - \sum_{i=1}^{K} x_i$ and learning rate η satisfies*

$$R_T \leq \frac{\eta T K}{2} + \frac{m \log \frac{K}{m}}{\eta}.$$

Here $F(x)$ is a Legendre function and is a parameter used within the OSMD algorithm. The trend detection algorithm in this model uses the same idea as in Algorithm 3 except that instead of a single action we now check if the set of m best actions have changed with probability at least $1 - \delta$. Even in this case, we denote by t^* the number of samples needed for each action to ensure that trend detection works with the above mentioned probability. The bounds derived in Lemma 2.3 and Lemma 2.4 apply in this case too.

There are only a few differences in Algorithm 5 as compared to Algorithm 2. Firstly, instead of using Exp3 for regret minimization we use the more sophisticated OSMD algorithm. This algorithm gives tight regret guarantees and is polynomial-time computable[2]. Secondly, the trend detection algorithm changes slightly as mentioned above.

[2]The OSMD technique can also be used when there are more generic combinatorial constraints on the set of actions chosen each round. For these generic cases, the algorithm need not be polynomial-time computable. However, for the uniform matroid case (under consideration here) it is in fact polynomial-time computable.

Finally, since we choose m actions in every round, we need a factor of m lesser number of loss monitoring plays. Alternatively, the size of an interval I is chosen to be $\frac{Kt^*}{m\gamma}$.

Algorithm 5 OSMD.T

```
▷ Parameters: δ, γ and η
Set interval length |I| = Kt*/(mγ)
for each interval I do
    Schedule ← Make_Schedule(I)
    for t = 1, 2 ··· |I| do
        if Schedule(t) = OSMD Play then
            Call OSMD_play()
        else
            Call LM_play(Schedule(t) )
        end if
    end for
    if trendDetection() == True then
        Restart OSMD
    end if
end for
```

Theorem 2.8. *The expected regret of OSMD.T is*

$$R_T = O\left(\frac{Nm\sqrt{TK\ln\left(\frac{TK}{m}\right)}}{\Delta_{sp}}\right).$$

Proof. The main analysis steps in this setting are exactly the same as in Theorem 2.5. The component of regret due to the OSMD algorithm is

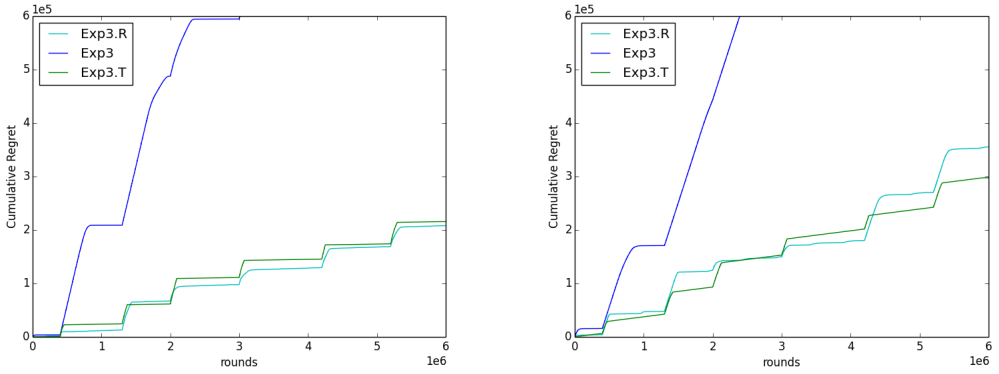
$$R_{osmd} \;\leq\; \frac{\eta TK}{2} + (N - 1 + F(T))\frac{m\log\frac{K}{m}}{\eta},$$

where $F(T)$ is the number of false detections as before and given by $F(T) \leq \delta\left(\frac{T}{|I|}+1\right)$. This inequality follows by Lemma 2.7 and considering the fact that the algorithm is restarted at most $N-1+F(T)$ times. Following the same arguments as in Theorem 2.5, the regret incurred on account of wasted intervals is at most:

$$R_{wasted} \;\leq\; Nm\left(1+\frac{1}{1-\delta}\right)|I|.$$

Unlike Theorem 2.5, each wasted round incurs a regret of m instead of 1 since we cannot guarantee regret for any of the chosen actions. Finally, since both the number of loss monitoring plays and the length of an interval are reduced by a factor of m, the regret incurred on account of the loss monitoring plays is:

$$R_{loss\_monitoring} \;\leq\; \left\lceil\frac{Kt^*}{m}\right\rceil\cdot\left\lceil\frac{T}{|I|}\right\rceil \;=\; O\left(\gamma T\right).$$



(a) $K = 2$, $\Delta_{sp} = 0.55$ (b) $K = 10$, $\Delta_{sp} = 0.40$

FIGURE 2.1: DSR Model

Putting these bounds together,

$$\begin{aligned} R_T &= R_{osmd} + R_{wasted} + R_{loss\_monitoring} \\ &\leq \frac{\eta TK}{2} + (N - 1 + \frac{\delta\gamma mT}{Kt^*})\frac{m \log \frac{K}{m}}{\eta} + Nm\left(1 + \frac{1}{1-\delta}\right)\frac{Kt^*}{\gamma m} + \gamma T. \end{aligned}$$

By setting $\eta = m\sqrt{\frac{\ln(K/m)}{TK}}$, $\delta = \sqrt{\frac{mK}{T}}$ and $\gamma = \frac{1}{m}\sqrt{\frac{Kt^*}{T}}$ we get:

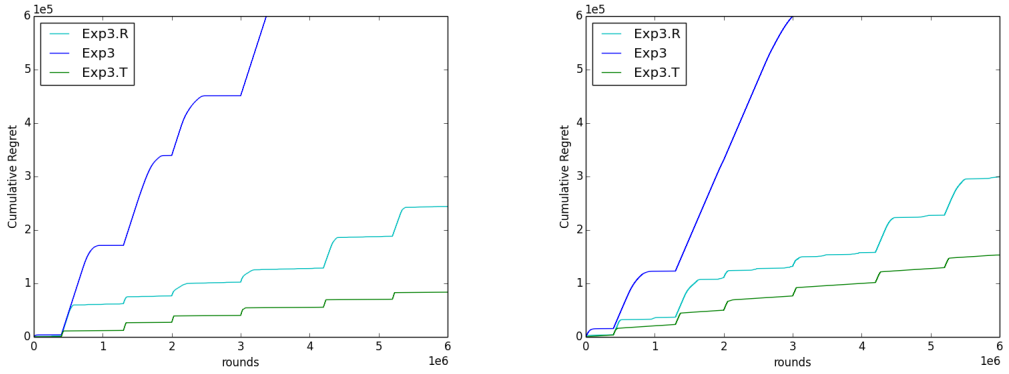
$$\begin{aligned} R_T \leq\ & m\sqrt{TK \ln \frac{K}{m}} + Nm\sqrt{TK \ln \frac{K}{m}} + \sqrt{mTK \ln \frac{K}{m} t^*} \\ & + 2Nm\sqrt{TKt^*} + 2NK\sqrt{mt^*} + \frac{1}{m}\sqrt{t^* TK}. \end{aligned}$$

Alternatively, $R_T = O\left(\frac{Nm\sqrt{TK \ln\left(\frac{TK}{m}\right)}}{\Delta_{sp}}\right)$. □

2.6 Simulations

Since our proposed algorithm falls into the domain of active learning, it is not possible to reliably use any fixed data set. Instead, to assess the performance of our algorithm we use artificially constructed loss generation models; a standard approach for problems of this nature.

For each of the two models introduced, we compare the performance of the Exp3.T algorithm with Exp3.R [11], an algorithm closest in spirit to our work. To emphasize that



(a) $K = 2$, $\Delta_{sp} = 0.40$ (b) $K = 10$, $\Delta_{sp} = 0.30$

FIGURE 2.2: ARG Model

we obtain *switching regret* guarantee, a stronger benchmark than what is conventionally used, we also compare our algorithm with Exp3 i.e., the performance, measured in terms of the cumulative loss, is with respect to a switching strategy that chooses the *best* action in each trend. Each experiment is run independently 10 times, and the mean of the results is shown in the figures.

Experiment 1: DSR model Within each trend, the expected loss of the best action is set to 0.10 and for other actions it is set to 0.5. This is the setup where $\Delta_{sp} = 0.4$. For the setup with $\Delta_{sp} = 0.55$, the expected loss of other actions is set to 0.65. For each of the loss models, we run the experiment with $K = 2$ and $K = 10$ actions respectively. The dynamic stochastic loss model used here is a representative of a worst case scenario i.e., we do not assume any information about the loss structure except for the separation parameter Δ_{sp} (refer Fig. 2.1). The performance of Exp3.T is almost identical to Exp3.R, an algorithm specifically designed for stochastic model. For a smaller gap, however, our algorithm still manages to do marginally better than Exp3.R. We note here that the parameters of the Exp3.R algorithm are set such that the assumptions required for the algorithm hold.

Experiment 2: ARG model We design the semi-structured property of the ARG model as follows: For $\Delta_{sp} = 0.3$ case, within each trend the loss of best action is a sequence of 100 consecutive 0s followed by 100 consecutive 1s. In the same rounds, losses of sub-optimal actions are 1 and 0.6 respectively. For $\Delta_{sp} = 0.4$ case, losses of the best action are same as before but losses of sub-optimal actions are kept constant at 0.9. These loss structures are chosen as representatives of the possible instances of the ARG model. The advantage of our algorithm is clearly highlighted in this more general model. The worse performance of Exp3.R is expected since it assumes more structure

than provided by the model; Exp3.T in contrast is able to exploit the little structure available and detect changes much faster.

There exists a subtle case when the guarantees presented in this chapter do not hold. This happens when the length of the interval is comparable to the total run time of the algorithm i.e. $O(T)$. For example, if the length of each interval is $T/2$, then Exp3.T does not provide any switching regret guarantee since for the first two intervals Exp3.T behaves exactly like Exp3. Therefore in the worst case, the regret bounds presented here are void but the bounds of Exp3 still apply.

Chapter 3

Learning with Computation Costs

In the previous chapter, we considered a bandit learning problem, where the learner chose only one action every round. Furthermore, it was implicitly assumed that the computation costs incurred to choose an action every round were negligible. In this chapter, we extend the classical model of the multi-armed bandits problem to account for these changes. We point out to the reader that unlike in the previous chapter, in this chapter we study algorithms that are designed for the stochastic reward model.

3.1 Introduction

Consider the following motivating example: There is a wireless sensor network with sensors spread across a geographical area. Any given sensor can communicate with other sensors in its neighborhood on fixed pre-defined channels. The throughput of these channels is, however, apriori unknown. Specifically, for any given channel, the observed throughput in any given round is drawn from a fixed but unknown distribution. Furthermore, the sensors are power constrained and incur a constant cost, in terms of the power spent, on every unit of communication. Our goal in such a network is to find a spanning tree in this network with maximum throughput to ensure efficient broadcasting of data.

This is a representative of the learning problems encountered in decentralized multi-agent systems. Another prominent one being that of multi-user channel allocations in cognitive radio networks. In this, the goal is to learn an optimal allocation of available channels to players so that the cumulative throughput is maximized. This problem has been addressed under various model assumptions, see [21–25].

Similar to the models studied previously, we assume that the performance of channels (or in general actions) is stochastic in nature. This may be viewed as being stochastic noise. The goal is to compute / learn the efficiency maximizing configuration. One may as well abstract the problem a bit to pose it as a general combinatorial multi-armed bandit problem. In this generalized model, the learner chooses a *feasible* set of actions every round. A feasible set is determined by the problem under consideration; for example, in our motivating example, actions correspond to the set of all spanning trees in the network. The learner receives as feedback the reward / loss associated with (and only) the chosen set. The CMAB problem is therefore just a generalization of the classical multi-armed bandit problem to any combinatorial constraint on the set of actions.

In spite of the generalization mentioned above, there are several factors that differentiates our motivating problem from the canonical CMAB problem and hence the same algorithms do not work out of the box. For example, there is no concept of communication cost in the canonical problem. Similarly, since it is a decentralized system, there is additional overhead involved in our problem to even compute a solution (an action), and if needed, to change it. This necessitates the need for algorithms that are frugal in updating the actions and at the same time strive to minimize the regret incurred. We would like to point out to the reader that although our motivating example consists of several decentralized agents, they are not strategic and simply follow a central protocol. In this sense, this is a centralized learning problem.

Related Work

In the context of learning in decentralized systems, most prior research has focused on problems in concurrent and reinforcement learning. These learning models assume that agents are strategic and do not model the communication explicitly. Since several players learn simultaneously from their interaction with one another, there is often a strong game-theoretic component associated with it. In contrast, in this chapter we focus on system-wide and not device-level learning. We refer the interested reader to a nice survey on this topic [26].

The approach and the analysis are inspired broadly from the classical multi-armed bandits algorithms, for example [27–31]. More recently, there has been increased interest in *combinatorial* multi-armed bandit problems (CMAB). Some recent examples include [1, 13, 32]. However, this body of work assumes that the computation required to choose a set of actions can be performed every round without any overhead.

The problem of *decentralized* multi-armed bandits has been considered in some papers previously, although for very specific problems.Avner et al. [22] study the problem of

matching users to channels in cognitive radio networks. They design an algorithm to learn an orthogonal mapping over a period of time that is stable and works with minimal assumptions on communication between agents. However, their solution uses a complicated signaling protocol, and the mapping constructed does not have any performance guarantees except that it is stable. Gai et al. [21] also study a very similar problem but relax the constraint that agents may not directly communicate. They consequently achieve much stronger performance guarantees.

To find the middle ground between the two extremes on the assumption of communication, Kalathil et al.[23, 24] proposed a new model which allows the agents to communicate for purposes of co-ordination. Such communication incurs cost and is accounted for in overall regret calculations. Along these lines they proposed two algorithms studying the problem of matching agents to actions in settings when the reward characteristics of different actions differ for different agents.

Overview: In what follows, in Section 3.2, we describe the exact model under study and some related preliminaries like the benchmark used to measure the performance of our algorithm. In Section 3.3, we describe the main algorithm of this chapter and give concrete regret bounds for it.

3.2 Model and Preliminaries

Following the terminology used for the CMAB problem [13], we define a learning problem instance by the tuple $B = (\mathcal{E}, \Theta, P)$, where $\mathcal{E} = \{1, \cdots, L\}$ is the ground set of actions (also elements) that the learner may choose from, $\Theta \subset 2^{\mathcal{E}}$ is the subset of feasible combinations of actions, and P is a fixed but unknown probability distribution over a unit cube $[0, 1]^L$. The time is discrete and proceeds in rounds. In any given round t, the leaner may choose a set of actions (also solution) A^t and observes the rewards of each action in A^t. The rewards of other actions are not observed by the learner. The reward vector of the actions at any round t, denoted by $\mathbf{w}^t$, is drawn i.i.d from the distribution P. The total reward of the learner is the sum of the rewards of each action chosen, it is denoted by

$$f(A^t, \mathbf{w}^t) \;=\; \sum_{i \in A} w^t(i).$$

We denote the expected reward of actions as $\bar{\mathbf{w}} = \mathbb{E}_{\mathbf{w} \sim P}[w]$. The model described till now is exactly the one studied by Kveton et al [13]. In addition, associated with each *decision* round, that is, the round in which the learner / algorithm recomputes the solution, is a constant *computation cost* C. Furthermore, if the recomputed solution,

Algorithm 6 CombUCB$_1$, Kveton et al. [13]

1: **Input:** Feasible set Θ
2: **for all** $t = t_0, \ldots, n$ **do**
3: // Compute upper confidence bounds
4: $U_t(e) \leftarrow \hat{w}_{T_{t-1}(e)}(e) + c_{t-1,T_{t-1}(e)} \quad \forall e \in E$
5:
6: // Solve the optimization problem
7: $A_t \leftarrow \text{argmax}_{A \in \Theta} f(A, U_t)$
8:
9: // Observe the weights of chosen items
10: Observe $\{(e, w_t(e)) : e \in A_t\}$, where $w_t \sim P$
11:
12: // Update statistics
13: $T_t(e) \leftarrow T_{t-1}(e) \quad \forall e \in E$
14: $T_t(e) \leftarrow T_t(e) + 1 \quad \forall e \in A_t$
15: $\hat{w}_{T_t(e)}(e) \leftarrow \dfrac{T_{t-1}(e)\hat{w}_{T_{t-1}(e)}(e) + w_t(e)}{T_t(e)} \quad \forall e \in A_t$
16: **end for**

differs from the previous one, then switching to the new solution also incurs a constant *switching cost* S.

The goal of the learner is to maximize the expected cumulative reward over T rounds. Let A^* denote the expected optimal solution with respect to the distribution P i.e. $A^* = \text{argmax}_{A \in \Theta} f(A, \bar{\mathbf{w}})$. The performance of the algorithm used by the learner is measured against a strategy that chooses A^* in every round. In other words, if $\pi(i)$ denotes the solution chosen by the learner in round $i \in [1, T]$, then the performance is measured in terms of the expected cumulative regret, defined as:

$$R(T) \;=\; \mathbb{E}\left[\sum_{t=1}^{T} f(A^*, \mathbf{w}^{\pi(t)}) - f(A^t, \mathbf{w}^{\pi(t)}) \;-\; C \cdot \mathbb{1}\{A^{\pi(t)} \neq A^{\pi(t-1)}\} \;-\; S \cdot \chi(t)\right],$$

where $\chi(t)$ is an indicator variable that is 1 for rounds when the algorithm computes a solution and zero otherwise.

CombUCB$_1$ Algorithm

Since our algorithm is inspired from the algorithm, CombUCB$_1$, by Kveton et al. we briefly introduce it here. This algorithm, see Algorithm 6, designed for stochastic combinatorial semi-bandits problem, was itself motivated by the classical stochastic multi-armed bandit algorithm, UCB [29]. It proceeds by computing an upper confidence bound on the expected weight for each item e as in line 4 of Algorithm 6. $\hat{w}_s(e)$ is the average of s observed weights of item e, $T_t(e)$ denotes the number of times item e was chosen in

t rounds and $c_{t,s}$ is the confidence radius around the computes average and is given as:

$$c_{t,s} = \sqrt{\frac{1.5 \log t}{s}}.$$

By a basic application of Hoeffding's inequality it can be shown that the true expected weight of an item is within this confidence radius with high probability. Next, CombUCB$_1$ calls the optimization oracle to solve the combinatorial problem with UCBs as weights (line 7) and observes the weight of all chosen items. It is important to note here that the algorithm does not incur any additional cost to solve the combinatorial problem (whereas in our algorithm we account for it). Since the weights of other items remain unknown, this feedback is said to be semi-bandit.

3.3 The CombUCB$_4$ Algorithm

The approach in this chapter is inspired from two existing algorithms in the stochastic reward model. The first, given by Kveton et al [13], ensures a logarithmic regret guarantee for the vanilla version of the problem, i.e., without switching or computation costs. Their algorithm is based on upper confidence bounds on action rewards to compute a feasible solution in each round. The second algorithm by Kalathil et al [23] uses a similar approach to give an $O(\log^2 T)$ regret bound in the case when the learner chooses a single action but also incurs computation cost. The CombUCB$_4$ algorithm presented here draws upon these techniques and ensures a $O(\log^2 T)$ regret bound for the CMAB problem with switching costs. In what follows, we use the term *action* and *element* interchangeably.

We denote the upper confidence bound of element e at time t as:

$$U_t(e) = \hat{w}_{T_{t-1}(e)}(e) + c_{t-1,T_{t-1}(e)},$$

where $\hat{w}_s(e)$ denotes the empirical mean of s observed weights, drawn i.i.d from an unknown distribution, of element e, $T_{t-1}(e)$ denotes the number of times element e was observed in $t-1$ rounds and $c_{t,s}$ is the confidence interval around the expected reward, $\bar{w}_e$, of element e, and is computed as:

$$c_{t,s} = \sqrt{\frac{2.5 \log t}{s}}.$$

We denote by A^* the optimal solution, i.e., $A^* = \underset{A \in \Theta}{\operatorname{argmax}} \sum_{e \in A} \bar{\mathbf{w}}(e)$. The *gap* of a solution A is $\Delta_A = f(A^*, \bar{\mathbf{w}}) - f(A, \bar{\mathbf{w}})$, where $f(S, \mathbf{w})$ denotes the reward of solution S under

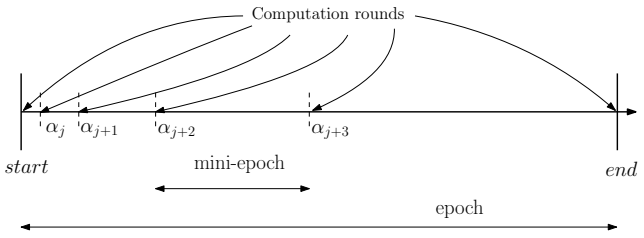


FIGURE 3.1: Epoch structure

weight $\mathbf{w}$. Let $\Delta_{e,min}$ be the minimum gap of any sub-optimal solution containing element $e \in \tilde{\mathcal{N}}$, i.e.,

$$\Delta_{e,min} = \min_{A \in \Theta : e \in A} \Delta_A,$$

where $\tilde{\mathcal{N}} = \mathcal{N} \setminus A^*$. The rounds when a solution is computed are denoted by the sequence of random variables $\{\alpha_j\}_{j=1}^{\chi(T)}$, where $\chi(T)$ is a random variable denoting the total number of computations. The actual values of these random variables depend on the particular run of the stochastic process. We refer to the run of the algorithm between two successive computations as a *mini-epoch* and the time interval during which the algorithm chooses the same solution in consecutive rounds as an epoch. Naturally, an epoch contains one or more mini-epochs, see Figure 3.1.

Algorithm 7 CombUCB$_4$

Initialization: Choose each action in $\mathcal{E}$ at least once. Update $U_t(e)$ for $e \in [1, L]$.
$\eta \leftarrow 1$
while $t \leq T$ **do**
  if $\eta = 2^p$ for some $p = 0, 1 \cdots$ **then**
    // Update UCBs
    $U_t(e) = \hat{w}_{T_{t-1}(e)}(e) + c_{t-1,T_{t-1}(e)}$

    // Compute new solution
    $A^t \leftarrow \operatorname{argmax}_{A \in \Theta} f(A, U_t)$
    if $A^t \neq A^{t-1}$ **then**
      Reset $\eta \leftarrow 1$
    end if
  else
    $A^t \leftarrow A^{t-1}$
  end if
  $\eta \leftarrow \eta + 1$

  // Update statistics
  $T_t(e) \leftarrow T_{t-1}(e)$ // $\forall e \in A^t$
  $T_t(e) \leftarrow T_{t-1}(e) + 1$ // $\forall e \in A^t$
  $\hat{w}_{T_t(e)}(e) \leftarrow \frac{\hat{w}_{T_{t-1}(e)}(e) T_{t-1}(e) + w_t(e)}{T_t(e)}$ // $\forall e \in A^t$
end while

Theorem 3.1. *The regret of algorithm CombUCB$_4$ is bounded as follows:*

$$R(T) \;\leq\; 96K^{\frac{4}{3}}\log T\left[\sum_{e\in\tilde{\mathcal{N}}}\left(\frac{2}{\Delta_{e,min}} \;+\; \frac{(1+\log T)(C+2S)}{\Delta^2_{e,min}}\right)\right] \;+\; (C+2S)\frac{\pi^2 N}{3} \;+\; 1.$$

Proof. Let $\xi_t = \{\exists e \in \mathcal{N} : |\bar{w}(e) - \hat{w}_{T_{t-1}(e)}(e)| \geq c_{t-1,T_{t-1}(e)}\}$ be the event that the empirical estimate of element e is outside the confidence interval around $\bar{w}(e)$ for some item e at round t. Let $\bar{\xi}_t$ be the complement of ξ_t, i.e. for all elements e the empirical estimate is within the confidence interval of the actual mean. For ease of exposition we shall refer to ξ as a *bad* event and $\bar{\xi}$ as *good* event. In each computation step, the algorithm incurs a constant cost, C, and for each epoch change a loss of at most S. Based on this notation, the regret incurred by CombUCB$_4$ can be expressed as:

$$\begin{aligned} R(T) \;\leq\; & \sum_{t=1}^{T}\mathbb{1}\{\Delta_{A_{\pi(t)}}\geq 0,\bar{\xi}_t\}R_{A_{\pi(t)}} \;+\; \sum_{j=1}^{\chi(T)}\mathbb{1}\{\xi_{\alpha_j}\}R_{\alpha_{j+1}-\alpha_j} \\ & + \sum_{j=1}^{\chi(T)}\Big(C + S\mathbb{1}\{A_{\pi(\alpha_j+1)}\neq A_{\pi(\alpha_j)}\}\Big), \end{aligned} \tag{3.1}$$

where $R_{\alpha_{j+1}-\alpha_j}$ is the regret incurred in the jth mini-epoch, A_i is the solution selected in epoch i, and R_{A_i} denotes the regret incurred in epoch i.

Part 1. ***Regret due to bad events:*** *Because the length, position, and number of epochs are determined by a stochastic process it is cumbersome to directly bound the regret using expression in Equation 3.1. Hence, we take an indirect approach. Instead of bounding the number of bad events as we do below, let's focus on directly bounding the regret incurred due to the bad events. Note that the expected regret incurred due to bad events is upper bounded by the following:*

$$\begin{aligned} \sum_{j=1}^{\chi(T)}\mathbb{1}_{\{\xi_{\alpha_j}\}}R_{\alpha_{j+1}-\alpha_j} \;&=\; \sum_{m=1}^{T}\sum_{k=0}^{\infty}2^k\cdot Pr\left(\textit{bad event occurs at round } m+2^k\right) \\ &=\; \sum_{m=1}^{T}\sum_{k=0}^{\infty}2^k\cdot Pr\left(|\bar{w}(e)-\hat{w}_{T_{m+2^k}(e)}(e)|\geq c_{m+2^k,T_{m+2^k}(e)}\right) \end{aligned}$$

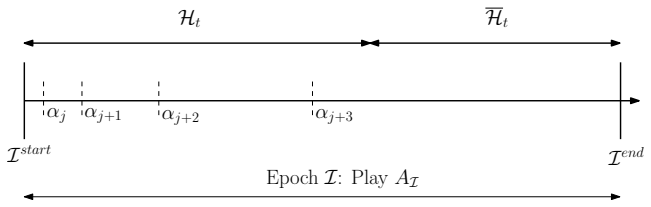


FIGURE 3.2: Regret conditioned on good events

Taking union bound on all possible values of $T_{m+2^k}(e)$*, the above expression can be upper bounded as:*

$$\begin{aligned} R_{bad}(T) &\leq \sum_{m=1}^{T}\sum_{k=0}^{\infty} 2^k \sum_{s=1}^{m+2^k} Pr\left(|\bar{w}(e) - \hat{w}_s(e)| \geq c_{m+2^k,s}\right) \\ &\leq \sum_{m=1}^{T}\sum_{k=0}^{\infty} 2^k \sum_{s=1}^{m+2^k} \frac{1}{(m+2^k)^5} \\ &\leq \sum_{m=1}^{T}\sum_{k=0}^{\infty} \frac{2^k}{(m+2^k)^4} \leq 1. \end{aligned}$$

Part 2. ***Regret conditioned on good events:*** *For ease of exposition, we define a mapping* π *that maps any round to the corresponding epoch based on the actual sequence of actions chosen, i.e., for all rounds* t *in epoch* $\mathcal{I}$*,* $\pi(t) = \mathcal{I}$*. Based on this, we define an event as follows:*

$$\mathcal{H}_t = \left\{ A_{\pi(t)} = \underset{A}{\operatorname{argmax}}\, f(A, U_t),\ \Delta_{A_{\pi(t)}} \leq 2 \sum_{e \in \tilde{A}_{\pi(t)}} c_{T,T_{t-1}(e)}, \Delta_{A_{\pi(t)}} > 0 \right\}, \quad (3.2)$$

where $\tilde{A}_{\pi(t)} = A^* \setminus A_{\pi(t)}$*. This event will be used in the sequel.*

Consider any epoch $\mathcal{I}$*, where a sub-optimal solution* $A_{\mathcal{I}}$ *is chosen. Let the start and end round of epoch* $\mathcal{I}$ *be denoted by* $\mathcal{I}^{start}$ *and* $\mathcal{I}^{end}$*, respectively. Since the system chose*

solution $A_{\mathcal{I}}$ over A^* at round $\mathcal{I}^{start}$, this implies:

$$\begin{aligned} f(A^*, U_t) &\leq f(A_{\mathcal{I}}, U_t) \\ \sum_{e \in A^* \setminus A_{\mathcal{I}}} U_t(e) &\leq \sum_{e \in A_{\mathcal{I}} \setminus A^*} U_t(e) \\ \sum_{e \in A^* \setminus A_{\mathcal{I}}} \bar{w}(e) &\leq \sum_{e \in A_{\mathcal{I}} \setminus A^*} \bar{w}(e) + 2 \sum_{e \in A_{\mathcal{I}} \setminus A^*} c_{t-1, T_{t-1}(e)} \\ \Delta_{A_{\mathcal{I}}} &\leq 2 \sum_{e \in A_{\mathcal{I}} \setminus A^*} c_{t-1, T_{t-1}(e)} \\ \Delta_{A_{\mathcal{I}}} &\leq 2 \sum_{e \in A_{\mathcal{I}} \setminus A^*} c_{T, T_{t-1}(e)}. \end{aligned} \tag{3.3}$$

This implies that at time $t = \mathcal{I}^{start}$, event $\mathcal{H}_t$ must have taken place. Consider the epoch as shown in Figure 3.2 for illustration. If the system chose solution $A_{\mathcal{I}}$ from rounds α_j to α_{j+3}, then by definition of the confidence bound it must be the case that for all time $t \in [\mathcal{I}^{start}, \alpha_{j+3}]$, $U_t(A_{\mathcal{I}}) \geq U_t(A^*)$. Alternatively, for this time interval the event $\mathcal{H}_t$ must hold. Since the solution changed after the execution at $\mathcal{I}^{end}$, it must be the case that the event $\mathcal{H}_t$ stopped being true for some $t \in [\alpha_{j+3}, \mathcal{I}^{end}]$. We denote the length of the interval for which the event $\mathcal{H}_t$ was true by $z_{\mathcal{I}}$. We shall refer to the rounds left in epoch $\mathcal{I}$ after $z_{\mathcal{I}}$ as wasted rounds of epoch $\mathcal{I}$. To proceed with the analysis, we rely on a Lemma from [13] used to bound the regret of the CombUCB$_1$ algorithm.

Lemma 3.2 (Kveton et al.[13]). *Let*

$$\mathcal{F}_t = \left\{ \Delta_{A_t} \leq 2 \sum_{e \in \tilde{A}_t} c_{T, T_{t-1}(e)}, \;\; \Delta_{A_t} > 0 \right\}$$

be an event as defined above where A_t denotes the action chosen by the CombUCB$_1$ algorithm in round t. Then,

$$\sum_{t=1}^{T} \Delta_{A_t} \mathbb{1}_{\{\mathcal{F}_t\}} \leq \sum_{e \in \tilde{\mathcal{N}}} \frac{96 K^{4/3}}{\Delta_{e,min}} \log T.$$

Note that conditioned on good events, the events $\mathcal{F}_t$ and $\mathcal{H}_t$ are identical. It is implicit in the CombUCB$_1$ algorithm that whenever the event $\mathcal{F}_t$ occurs, the chosen solution A_t is optimal with respect to the upper confidence bounds. In the case of dCombUCB$_4$ however, even when conditioned on good events, the chosen solution $A_{\pi(t)}$ at round t might not be optimal with respect to confidence bounds at time t. We shall denote such

an event by $\overline{\mathcal{H}}_t$ *i.e.*

$$\overline{\mathcal{H}}_t \;=\; \left\{ A_{\pi(t)} \neq \operatorname*{argmax}_A f(A, U_t), \bar{\xi}_t \right\}.$$

Using these definitions we can bound the regret incurred by dCombUCB$_4$ conditioned on good events as follows:

$$\begin{aligned}
&\sum_{t=1}^{T} \Delta_{A_{\pi(t)}} \cdot \mathbb{1}_{\{\bar{\xi}_t, \Delta_{A_{\pi(t)}} > 0\}} \\
&\leq \sum_{t=1}^{T} \Delta_{A_{\pi(t)}} \cdot \mathbb{1}_{\{\bar{\xi}_t, \mathcal{H}_t\}} + \sum_{t=1}^{T} \Delta_{A_{\pi(t)}} \cdot \mathbb{1}_{\{\bar{\xi}_t, \overline{\mathcal{H}}_t\}} \\
&\overset{(a)}{\leq} 2 \sum_{t=1}^{T} \Delta_{A_{\pi(t)}} \cdot \mathbb{1}_{\{\bar{\xi}_t, \mathcal{H}_t\}} \\
&\overset{(b)}{\leq} 2 \sum_{t=1}^{T} \Delta_{A_{\pi(t)}} \cdot \mathbb{1}_{\{\mathcal{F}_t\}} \\
&\overset{(c)}{\leq} \sum_{e \in \tilde{\mathcal{N}}} \frac{192 K^{4/3}}{\Delta_{e,min}} \log T.
\end{aligned} \tag{3.4}$$

The inequality (a) is evident from the fact that the number of rounds event $\overline{\mathcal{H}}_t$ *occurs is upper bounded by the number of rounds event* $\mathcal{H}_t$ *occurs. (b) is based on the observation that event* $\{\bar{\xi}_t, \mathcal{H}_t\}$ *is equivalent to event* $\mathcal{F}_t$ *as defined in Lemma 3.2. (c) follows directly from Lemma 3.2.*

Part 3. ***Regret due to computation and switching cost:*** *First assume that all computations occur conditioned on good events. We can express* $\chi(T) = \chi_1(T) + \chi_2(T)$ *where* $\chi_1(T)$ *and* $\chi_2(T)$ *denote the number of computations that resulted in a sub-optimal and an optimal solution being chosen respectively. Note that* $\chi_1(T)$ *can be upper bounded by the number of times the algorithm chose a sub-optimal solution, i.e.*

$$\begin{aligned}
\chi_1(T) \;&\leq \sum_{\substack{t=1 \\ A_{\pi(t)} \neq A^*}}^{T} \mathbb{1}_{\{\mathcal{H}_t\}} \\
&\leq \sum_{e \in \tilde{\mathcal{N}}} \frac{96 K^{4/3}}{\Delta_{e,min}^2} \log T,
\end{aligned} \tag{3.5}$$

where the second inequality is due to Theorem 3, [13] derived as part of analysis of CombUCB$_1$. To bound $\chi_2(T)$*, note that the number of computations that result in a transition from a sub-optimal to optimal solution is upper bounded by* $\chi_1(T)$*. Furthermore, for every such transition, there can be at most* $O(\log T)$ *computations without*

switching to a sub-optimal solution. Therefore, $\chi_2(T)$ *is bounded as:*

$$\chi_2(T) \;\leq\; \chi_1(T) \log T.$$

This bound is conditioned on good events. The number of computations on account of bad events can simply be bounded by the number of these bad events and can be bounded as:

$$\sum_{e \in \mathcal{N}} \sum_{t=1}^{n} \sum_{s=1}^{t} P\left[|\bar{w}(e) - \hat{w}_s(e)| \geq c_{t,s}\right] \;\leq\; \frac{\pi^2}{3} N.$$

Finally, the number of switches *can be bounded by* $2\chi_1(T)$*. Putting this all together, the regret due to computation and switching cost is bounded by:*

$$(C + 2S) \cdot \left[\sum_{e \in \tilde{\mathcal{N}}} \frac{96K^{4/3}}{\Delta^2_{e,min}} \log T(1 + \log T) + \frac{\pi^2 N}{3} \right].$$

□

3.4 Open Problem

In this chapter, we focused on the combinatorial multi-armed bandit problem with semi-bandit feedback. Since the rewards of the actions are stochastic, it allows the use of well understood techniques like the variants of UCB that we have used here. The problem however, is much more challenging for the case of adversarial rewards. For the classical problem, i.e. where the learner chooses only one action in a round, [33] give an algorithm with a switching regret guarantee of $\Theta(T^{2/3})$. This was supplemented by the lower bound by [34] who showed that no algorithm can guarantee a switching regret bound better than $T^{2/3}$. In this sense, the algorithm in [33] is optimal. The techniques for the vanilla version, however, do not extend to the CMAB problem studied here. While there exist algorithms [35] that guarantee regret bound of $O(T^{1/2})$ with semi-bandit feedback, they have not yet been extended to the switching costs model.

Part II

Pricing in Markets with Gross-Substitutes Utilities

Chapter 4

Motivation and Preliminaries

4.1 Dynamic Pricing in the Presence of Competition

In the subsequent chapters, and especially in Chapters 5 and 6, we investigate several pricing strategies in some prominent market models. A characterizing aspect of our study is the incorporation of the competitive nature of the market agents. In this section, we motivate our study in a general sense by sketching some of the past approaches taken and why they are inadequate in competitive environments.

The Internet has revolutionized the way goods are bought and sold. This has created a range of new possibilities to price the goods strategically and dynamically. This is especially true for online retail and apparel stores where the cost and effort to update prices has become negligible. This flexibility in pricing has propelled the research in *dynamic pricing* in the last decade or so, informally defined as the study of determining optimal selling prices in an unknown environment to optimize an objective, usually revenue. Coupled with the presence of digitally available and frequently updated sales data one may also view this as an (online) learning problem.

The inherent hurdles in dynamic pricing arise on account of *lack of information*. In the context of a single good, this could be the underlying demand function that maps a given price to the observed demand. Indeed, this problem has been studied in several models in the literature and strong results are now known for it. However, the problem becomes all the more challenging in a realistic setting where multiple sellers independently choose prices for their goods, and the demand observed by any single seller is a function of all the prices. For example, some fixed seller might observe completely different demands for the same price she uses for her items depending on the prices chosen by other sellers. Such a seller might falsely conclude of being in a dynamic environment even when the underlying demand function is static.

Several existing approaches for dynamic pricing assume a parametric form for the underlying demand function and choose a sequence of prices to learn the individual parameters by statistical estimation. This approach is commonly referred to as "learn-and-earn" in the literature [36, 37]. It would, however, be unrealistic in the presence of multiple sellers since that would imply learning highly nonlinear and possibly unstructured functions in high dimensions.

For this problem, we propose two different approaches depending on the market model under consideration. These approaches rely on some existing results in the domain of markets and equilibrium theory. We briefly introduce them in Section 4.2. In Chapter 7, in addition to these concepts, we also use sequential learning algorithms and the corresponding analysis for online convex optimization. These are reviewed in Section 4.3.

4.2 Markets and Equilibrium

A sizeable body of literature in economics has focused on the question of how a large number of seemingly unconnected decisions taken by strategic agents, namely the buyers and sellers in any large market, leads to a balancing of supply and demand and thereby facilitates an efficient allocation of goods in the market. Perhaps as one might expect, all of the classical works point to the same underlying reasoning: It is the pricing system in the market that inherently guides it towards efficiency. To better explain the concept of efficiency, let us first define a market. Specifically, since the subsequent chapters focus on *Fisher markets*, we shall restrict our discussion only to this class of markets.

Fisher Market: A Fisher market can be defined as a congregation of two types of agents, namely the buyers and the sellers. Let there be m buyers and n sellers. For ease of exposition assume each of the n sellers bring exactly one good to the market, where seller i brings w_i units of her good. This good is assumed to be infinitely divisible. The buyers on the other hand bring money to the market. The budget of buyer j is denoted by b_j. Furthermore, associated with each buyer is a utility function that quantifies the *value* the buyer derives from a given bundle of goods. We note here that the seller has no utility associated with the goods she brings. Similarly, the buyers derive utility only from the bundle of items they get and therefore, spends their entire budget. Let $\mathbf{p}$ denote the vector of prices chosen by the sellers. In response, every buyer j internally solves an optimization problem to compute the bundle of goods that would maximize her utility given the prices and subject to her budget constraint. This forms the *demand* $\mathbf{x}_j(\mathbf{p})$ of buyer j. We denote by $\mathbf{x}_{ij}(\mathbf{p})$ the demand of buyer j for item i. It is implicit in the definition of this market that the budget and the demand of a buyer for any good

is always non-negative. The *supply of good* i is w_i, and we set $\mathbf{w} = (w_i)_{i=1,\ldots,n}$. Let $\mathbf{z} = (z_i)_{i=1,\ldots,n}$ be the vector of *excess demand*, i.e., demand minus supply: $z_i = x_i - w_j$.

Note: For a given price vector $\mathbf{p}$, we refer to the demand of buyer j by $\mathbf{x}_j(\mathbf{p})$, whereas the cumulative demand observed by a seller i is denoted by $\mathbf{x}_i(\mathbf{p})$.

A pair $(\mathbf{x}^*, \mathbf{p}^*)$ is a *competitive* or *market equilibrium* if (1) each vector $\mathbf{x}_j^*$ is a demand of buyer j at prices $\mathbf{p}^*$, (2) for each good i with $p_i^* > 0$, demand is equal to suppy (i.e., $p_i^* \cdot z_i = 0$), and (3) for each good i with $p_i^* = 0$, demand is at most supply (i.e., $z_i \leq 0$). An equilibrium price vector $\mathbf{p}^*$ is also called a vector of *market clearing prices*. In general, such a price may not exist but for the utility functions we use in subsequent chapters, existence is guaranteed.

While the existence of a competitive equilibrium for such markets has been qualitatively proven in economics, the question of whether they can be *computed* efficiently remained until recently largely unaddressed. More recently, computation of competitive equilibrium has become a central area in algorithmic game theory, in which a variety of interesting algorithmic techniques have been applied successfully [38, 39]. Indeed, in a market where equilibrium price is known to exist, if one cannot compute the equilibrium even using computers, it is likely that such equilibria will fail to arise in reality too. While many computational techniques to compute competitive equilibria are inherently centralized, most large-scale markets lack a central authority that determines and dictates prices and allocation. Instead, prices of goods are updated in a distributed fashion. Towards this end, Walras [40] proposed a natural price adaptation process, called *tatonnement*, and has been studied as a continuous-time process in economics since the late 1950s.

The basic idea underlying the tatonnement process is as follows: if the demand observed for a good is more than the supply, i.e., if the good is over-demanded, increase the price of the good. In most natural settings, one would expect a decrease in the demand. On the other hand, if a good is under-demanded, decrease the price of the good to achieve the opposite effect. For general Arrow-Debreu exchange markets, price updates of this form converge to an equilibrium in markets with gross substitutes property [41], but might not converge in markets beyond this class [42]. In recent years, discrete-time tatonnement has been analyzed in a series of works in computer science. These results show fast convergence of this process in Fisher markets with utilities with constant elasticity of substitution (CES), even when the utilities imply that buyers have complementary preferences.

4.2.1 Gross Substitutes

Gross Substitutes markets are a prominent class of markets where positive results, like uniqueness and stability of competitive equilibrium, has been proven. Since this property is fundamental to the algorithms and analysis introduced in the subsequent chapters, we introduce it here. Two goods are said to be gross substitutes if an increase in the price of one good increases the demand for the other. Informally, a buyer can replace one good by another if its price increases so as to maximize the utility gains. Formally, it is defined as follows:

Definition 4.1. *A demand function $x(\mathbf{p})$ is said to satisfy the gross-substitutes property if, whenever $\mathbf{p}$ and $\mathbf{p}'$ are such that $p'_k > p_k$ and $p'_l = p_l$ for all $l \neq k$, then $x_l(\mathbf{p}') > x_l(\mathbf{p})$ for all $l \neq k$.*

4.2.2 CES Utilities

As mentioned in the previous section, strong positive results are known about the efficacy of the tatonnement update process to converge to equilibrium when the buyer utilities satisfy the CES property. Since we also extensively use this property in the Chapters 5 and 7, we introduce the property here for all subsequent references.

The utility u_j of buyer j is said to satisfy the constant elasticity of substitution (CES) property, when for any demand bundle x_j it is of the form,

$$u_j(\mathbf{x}_j) = \left(\sum_{j=1}^{m} a_{ij} \cdot (x_{ij})^{\rho} \right)^{1/\rho}, \tag{4.1}$$

where $1 \geq \rho > -\infty$ and all $a_{ij} \geq 0$. Utility functions with $1 > \rho > 0$ satisfy the gross-substitutes property and are our primary focus in Chapter 6. According to this property, if the price of any good i increases, then the demand for any other good j also increases. This property enables us to show several positive results. For $\rho < 1$ and $\rho \neq 0$, buyer j's demand for good i is

$$\hat{x}_{ij} \;=\; b_i \cdot \frac{(a_{ij})^{1-c}(p_j)^{c-1}}{\sum_{k=1}^{n}(a_{ik})^{1-c}(p_k)^{c}}, \quad \text{where} \quad c \;=\; \frac{\rho}{\rho - 1} \;.$$

Elasticity of demand: In the course of the analysis in subsequent chapters, we often need to use the definition of elasticity of demand (also, own price elasticity of demand). This is a market parameter that measures the responsiveness of the demand of an item for a given change in its price. Formally, it is the percentage change in demand for a

unit percentage change in the price. For price vector $\mathbf{p}$ the elasticity of demand is given by:

$$E_i(\mathbf{p}) \;=\; \frac{\partial x_i(\mathbf{p})/x_i(\mathbf{p})}{\partial p_i/p_i}.$$

Similarly, we can also define the cross-price elasticity of demand which captures the sensitivity of demand of a good to changes in the price of another good. The cross-price elasticity of good i with respect to good j is:

$$E_{ij}(\mathbf{p}) \;=\; \frac{\partial x_i(\mathbf{p})/x_i(\mathbf{p})}{\partial p_j/p_j}.$$

4.2.3 IGS Utilities

In Chapter 6, we assume a class of buyer utilities which do not have a closed form expression as in the case CES utilities but nevertheless satisfy the gross substitutes property. We define it here for subsequent references.

Definition 4.2 (Iso-elastic and Gross Substitutes (`IGS`) utility). *We say that a utility function is `IGS` when it satisfies the following conditions:*

a) The utility function satisfies the gross substitutes property and the resulting demand functions are continuous.

b) Increasing the price of any good i decreases the total spending on the item i.e. $p_i x_i(\mathbf{p})$.

c) The price elasticity of good i for any price vector $\mathbf{p}$ satisfies:

$$\left|\frac{\partial \ln x_i(\mathbf{p})}{\partial \ln p_j}\right| = E \qquad \forall j \in [1, n]$$

where $E > 1$ is a constant.

This model may be viewed as an approximate form of the CES utilities (with the parameter $\rho \in (0, 1)$) since they satisfy parts (a) and (b) in Definition 4.2. Note that for CES utilities, instead of a fixed constant as price elasticity, this parameter depended on the prices of all goods i.e. $\left|\frac{\partial \ln x_i(\mathbf{p})}{\partial \ln p_j}\right| = E_i(\mathbf{p})$.

4.3 Primer on Online Convex Optimization

In the following chapters, we use concepts and techniques pertaining to online convex optimization for our analysis. In this section, we give a brief primer on the basic model

studied, a benchmark to compare the performance of different learning algorithms, and one of the most well-studied class of algorithms.

Model: The problem consists of a learner and an adversary. The adversary chooses a sequence of convex functions $f_1, f_2, \ldots, f_T$. For our purposes, we assume the adversary is oblivious, i.e., the adversary chooses this sequence before the start of the algorithm. It therefore cannot be changed depending on the choices made by the algorithm. In any given round t, the learner chooses a distinct point $\mathbf{w}^t$ from a fixed convex set S before observing the loss function picked by the adversary for that round. Depending on this choice, the learner incurs a cost given by $f_t(\mathbf{w}^t)$ and receives as feedback the function f_t. The goal of the learner is to choose a sequence of points $\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^T$ such that $\sum_t f_t(\mathbf{w}^t)$ is minimized.

A systematic way to measure the performance of an algorithm in online learning problems is to analyze its regret. This benchmark measures the total cost incurred by the learner with respect to *any* single fixed action in hindsight. Formally, the regret with respect to any fixed action $\mathbf{w}^*$ is defined as:

$$\text{Regret}(\mathbf{w}^*) \;=\; \sum_{t=1}^{T} f_t(\mathbf{w}^t) \;-\; \sum_{t=1}^{T} f_t(\mathbf{w}^*).$$

Follow-the-Regularized-Leader (FTRL): FTRL is one of the standard algorithms studied for online learning problems in various models. It's appeal stems from the intuitive interpretation of the algorithm, which is to choose a point that simply minimizes the cost over all past rounds and an additional regularization term. This term stabilizes the dynamic by preventing big fluctuations in the actions taken. Formally, for a regularization function $R : S \to \mathbb{R}$ we can define the FTRL algorithm as follows:

$$\forall t: \qquad \mathbf{w}_t \;=\; \operatorname*{argmin}_{\mathbf{w} \in S} \sum_{i=1}^{t-1} f_i(\mathbf{w}) \;+\; R(\mathbf{w}).$$

For the special case, when $R(\mathbf{w}) = \frac{\|\mathbf{w}\|^2}{2\eta}$ the FTRL algorithm corresponds to the online gradient descent algorithm (also called greedy projection algorithm) by Zinkevich [3]. The learning algorithm can then also be described as follows:

1. Start with an initial point $\mathbf{w}^1 \in F$. Let $\{\eta_t\}$ be a sequence of learning rates.
2. In each round t, do:

$$\mathbf{w}^{t+1} \;\leftarrow\; \prod_{S} \left(x^t - \eta_t \cdot \nabla f_t(\mathbf{w}^t)\right).$$

Here ∇f_t denotes the gradient of the function at the chosen point and $\prod_S$ projects the resulting update back into the set S. Below, we state the regret bound of this algorithm and its analysis for completeness.

Theorem 4.3. *If $\eta_t = t^{-1/2}$, the regret of the greedy projection algorithm is:*

$$R(T) \;\leq\; \frac{\|S\|^2 \sqrt{T}}{2} \;+\; \left(\sqrt{T} - \frac{1}{2}\right) \|\nabla f\|^2 ,$$

where $\|S\|$ is the diameter i.e. $\max_{\mathbf{x},\mathbf{y}\in S} d(\mathbf{x},\mathbf{y})$ and $\|\nabla f\| = \sup_{\mathbf{w}\in S} \|\nabla f_t(\mathbf{w})\|$.

Proof. We first note that the learner observes the *value* of the loss incurred, i.e., $f_t(\mathbf{w}^t)$ as well as the gradient of the loss function. Let $\nabla f_t(\mathbf{w}^t) = g^t$. Since the loss function is convex, it follows that

$$f_t(\mathbf{w}^t) - f_t(\mathbf{w}^*) \;\leq\; g^t \cdot (\mathbf{w}^t - \mathbf{w}^*).$$

Summing over all rounds t, we have:

$$R(T) \;\leq\; \sum_t \; f_t(\mathbf{w}^t) - f_t(\mathbf{w}^*) \;\leq\; \sum_t \; g^t \cdot (\mathbf{w}^t - \mathbf{w}^*). \tag{4.2}$$

Let $\mathbf{y}^t$ denote the unprojected update at round t, i.e., $\mathbf{w}^t = \prod(\mathbf{y}^t)$. By the update rule of the algorithm,

$$\begin{aligned}
\mathbf{y}^{t+1} &= \mathbf{w}^t - \eta_t \cdot g^t \\
\mathbf{y}^{t+1} - \mathbf{w}^* &= \mathbf{w}^t - \mathbf{w}^* - \eta_t \cdot g^t \\
(\mathbf{y}^{t+1} - \mathbf{w}^*)^2 &= (\mathbf{w}^t - \mathbf{w}^*)^2 - 2\eta_t \underbrace{g^t \cdot (\mathbf{w}^t - \mathbf{w}^*)}_{\text{target expression}} + \; \eta_t^2 \cdot \left\|g^t\right\|^2 .
\end{aligned}$$

Since for all $\mathbf{y} \in \mathcal{R}^n$ and $\mathbf{w} \in F$, $(\mathbf{y} - \mathbf{w})^2 \;\geq\; \left(\prod(\mathbf{y}) - \mathbf{w}\right)^2$ and $\left\|g^t\right\| \leq \|\nabla f\|$ it follows that:

$$\begin{aligned}
(\mathbf{w}^{t+1} - \mathbf{w}^*)^2 &\leq (\mathbf{w}^t - \mathbf{w}^*)^2 - 2\eta_t g^t \cdot (\mathbf{w}^t - \mathbf{w}^*) + \eta_t^2 \cdot \|\nabla f\|^2 \\
g^t \cdot (\mathbf{w}^t - \mathbf{w}^*) &\leq \frac{1}{2\eta_t}\left((\mathbf{w}^t - \mathbf{w}^*)^2 - (\mathbf{w}^{t+1} - \mathbf{w}^*)^2\right) + \frac{\eta_t}{2} \cdot \|\nabla f\|^2 .
\end{aligned} \tag{4.3}$$

Using Inequalities 4.2 and 4.3 together,

$$
\begin{aligned}
& R(T) \\
& \leq \sum_t \frac{1}{2\eta_t} \left((\mathbf{w}^t - \mathbf{w}^*)^2 - (\mathbf{w}^{t+1} - \mathbf{w}^*)^2 \right) + \frac{\|\nabla f\|^2}{2} \sum_t \eta_t \\
& \leq \frac{(\mathbf{w}^1 - \mathbf{w}^*)^2}{2\eta_1} - \frac{(\mathbf{w}^{T+1} - \mathbf{w}^*)^2}{2\eta_T} + \frac{1}{2} \sum_{t=2}^{T} \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) (\mathbf{w}^t - \mathbf{w}^*)^2 + \frac{\|\nabla f\|^2}{2} \sum_t \eta_t \\
& \leq \left(\frac{1}{2\eta_1} + \frac{1}{2} \sum_{t=2}^{T} \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) \right) \|F\|^2 + \frac{\|\nabla f\|^2}{2} \sum_t \eta_t \\
& \leq \frac{\|S\|^2}{2\eta_T} + \frac{\|\nabla f\|^2}{2} \sum_t \eta_t .
\end{aligned}
$$

□

Chapter 5

Pricing via Tatonnement

5.1 Introduction

As described in Chapter 4, competitive equilibrium is the central solution concept when studying trading in large-scale market models. Moreover, it is also known that price update strategies based on the concept of tatonnement converge to equilibrium for a large class of markets. In this chapter, we investigate the question of whether such price update strategies are also individually rational for the sellers and the impact of these strategies on the sellers' revenue.

Tatonnement has been shown to realistically capture the behavior in laboratory experiments [43–45]. Furthermore, in recent years, discrete-time price update processes based on this concept are getting increasingly more attention. This is particularly the case for Fisher Markets with CES utilities, see for example [46, 47]. In these works, the authors have used several different price update methods and proved that the dynamics converge to equilibrium under their assumed market model. Although interesting in its own right, these updates have a strategic problem in the sense that it remains unclear why self-interested sellers should follow these protocols. Motivated by this issue, we study pricing from a revenue maximization perspective in repeated Fisher markets.

Specifically, we focus on the price update strategy used in Cole et al [46] as our tatonnement update. We show that for a static gross-substitutes market, apart from converging to equilibrium these dynamics have the additional property that if adopted by all sellers, the resulting sequence of prices yields a total revenue for each seller that is almost optimal in hindsight. In particular, each seller suffers only a constant total regret, i.e., a constant loss against the optimal total revenue achievable in hindsight via an individual best response in each of the T rounds. This result may be viewed in the

context of a recent line of work that studies novel performance guarantees for regret-minimizing players in repeated games [48–50]. The authors in these papers show that if each player chooses a regret minimizing algorithm belonging to a certain class, then this strategy results in a significantly improved external regret bound for each player. The results in this chapter are similar but differ in two important ways. First, instead of general no-regret learning we study the special but prominent adaptation process of tatonnement. Second, for such a process we provide very strong constant bounds in terms of the regret incurred with respect to the best responses in every round. This is a much stronger notion than the standard definition of regret, which is the loss with respect to that of a fixed action over all rounds. We also provide a brief conceptual argument as to why this adaptation process is a good approach – it is guaranteed to converge to a revenue-maximizing price for each seller individually when the prices of all other sellers stay fixed.

In addition to static markets, we also analyze the revenue guarantees of our tatonnement update in Fisher markets with dynamic supplies. This is especially interesting since for such markets the adaptation process itself is dynamic, since it depends on the supply observed in the current round. In the course of the analysis, we give a clear characterization of how the loss in revenue incurred by any seller depends on the variation in supplies and prove bounds which degrade smoothly with the total variation in supplies observed. When the total variation in supplies is large, for example when the supplies are chosen by an adversary, these bounds can be much worse than the one obtained by a fixed adaptation process, i.e. one using a fixed supply parameter. We show that a slight adaptation of the standard process using a supply estimator can address this problem by stabilizing the process and thereby leading to improved bounds on the cumulative loss of any seller.

Related Work

Our work studies discrete-time tatonnement price updates in Fisher markets with CES substitutes utilities. Such dynamics have received significant interest in algorithmic game theory over the last decade. In addition to guarantees on the convergence time [51], a focus has been the study of warehouses to store excess demands [52–54]. Moreover, discrete-time tatonnement has been shown to converge quickly to market equilibrium, even in Fisher markets without gross-substitutes property [55, 56]. However, these works focus squarely on the question of convergence to equilibrium and warehouse considerations, without regards to the incentives of sellers and the question of revenue efficiency of the price updates.

One of the distinctive features is our focus on revenue optimization in a market model explicitly incorporating competition. There has been prior work, for example [57, 58], focusing on dynamic pricing in models considering competition, but none of this literature studies a generalized market setting. Most approaches consider discrete choice models of demand, where a single consumer approaches and buys a discrete bundle of goods. The sellers in this model are also assumed to have a fixed inventory level which are not replenished. For a thorough survey of the existing literature, we refer the reader to [59].

In terms of incentive properties in Fisher markets, there has been some very recent work on forms of best-response dynamics with lookahead and changing beliefs [60]. These dynamics are grounded in modeling seller incentives and can be shown to converge to equilibrium. In contrast, we study the simple form of tatonnement and show that the resulting dynamics manage to provide near-optimal revenue.

Overview: This chapter investigates an established price adaptation strategy in Fisher markets with buyers that have CES utilities with gross-substitutes property. We start with markets with fixed supplies in Section 5.2. After presenting preliminaries in Section 5.2.1, the first main result is presented in Section 5.2.2: After T rounds of price updates, each seller has a total regret for his revenue that is at most a constant (Theorem 5.6). The constant term depends on a number of market parameters that stay invariant over time, such as the utility parameters and budgets of the agents and the initial prices. Moreover, based on these initial conditions a suitable fixed price space $\mathcal{P}$ is defined for the analysis. This price space is a function of the initial price, the utility functions and the parameters used in the price update, which are all invariant over time.

The technique used relies on establishing a bound on the revenue loss in a single round and connecting it to a potential function Ψ associated with the market. This potential function was proposed by [55] and may be interpreted as a measure of *distance* of the current price vector from the equilibrium. Specifically, it is shown that the revenue loss in a single round t for seller i can be tied using a time-invariant factor to its excess demand $|z_i^t|$, which in turn is shown to be related to Ψ via a β-smoothness property. By establishing these properties and using the fact that the price update rule guarantees a linear convergence rate in Ψ [55], we establish that the total accumulated revenue loss over all T rounds is bounded by a geometric series with constant parameters. Section 5.2.3 discusses how these price updates also justify the incentive considerations of a strategic seller by interpreting it as an update that myopically optimizes the revenue of the seller.

The above mentioned technique is extended to markets with dynamic supplies in Section 5.3 where there is a time-dependent supply w_j^t for each good j and each round t.

The loss in revenue in such markets is bounded using the same price adaption process as before. Section 5.3.2 focuses on a slightly modified price adaptation process that uses a supply estimator to choose the price in consecutive rounds. This supply estimator $\hat{w}_j^t$ is computed using the follow-the-regularized-leader strategy. For this modified process we obtain a bound that relates the revenue loss to an "regret" in supply, i.e., the total absolute norm difference of the supplies to the best single estimator of supply in hindsight (Theorem 5.12). Intuitively, this approach works well when the supply changes are adversarial.

5.2 Repeated Markets with Fixed Supplies

5.2.1 Preliminaries

We consider a market $\mathcal{M}$ consisting of a set $\mathcal{G}$ of n sellers and a set $\mathcal{B}$ of m buyers. Associated with each buyer $j \in \mathcal{B}$ is a fixed budget b_j and a utility function u_j (described below). We denote the cumulative buyer budget by $B = \sum_j b_j$. Each seller i offers a single, infinitely divisible good and strives to maximize revenue. We assume that the market operates in a synchronous, round-based fashion. The supply w_i of seller i is fixed throughout. The supplies of all sellers are succinctly represented by the vector $\mathbf{w}$.

In each round t, every seller i sets a fixed price p_i^t on her good, which yields a price vector $\mathbf{p}^t$. Depending on these prices, the buyers demand a utility-maximizing bundle, which yields an aggregate demand $x_i(\mathbf{p}^t)$ observed by seller i. After observing the demand, the seller meets this demand subject to availability. If a good is under-demanded (i.e., $x_i(\mathbf{p}^t) < w_i$), then the portion $w_i - x_i(\mathbf{p}^t)$ remains unsold. We assume goods are perishable, and unsold supply is discarded after each round.

For a given price vector $\mathbf{p}$, the demand results from buyers choosing a bundle of goods that maximizes their utility and is affordable. Let $\mathbf{y}_j = (y_{ji})_{i=1}^n$ denote an arbitrary bundle of goods for buyer j. The demand of buyer j is $\mathbf{y}_j(\mathbf{p}) = \text{argmax}\{u_j(\mathbf{y}_j) \mid \sum_i p_i y_{ji} \leq b_j\}$, and the total demand for good i is $\mathbf{x}_i(\mathbf{p}) = \sum_{j=1}^m y_{ji}(\mathbf{p})$. In this chapter, we assume that the utility of each buyer satisfies the gross substitutes CES property i.e, $0 < \rho < 1$. This property is discussed in detail in Chapter 4.

Log-Revenue Objective. Given this setup, for any feasible price vector $\mathbf{p}$, the revenue of seller i is $r_i(\mathbf{p}) = p_i \cdot \min\{x_i(\mathbf{p}), w_i\}$ where $x_i(\mathbf{p})$ and w_i are the demand observed and the supply of seller i. If the buyer utilities are as in (4.1), then the revenue function as mentioned above is not concave and as such is not amenable to optimization. Interestingly however, the revenue objective in log scale, also called the *log-revenue*,

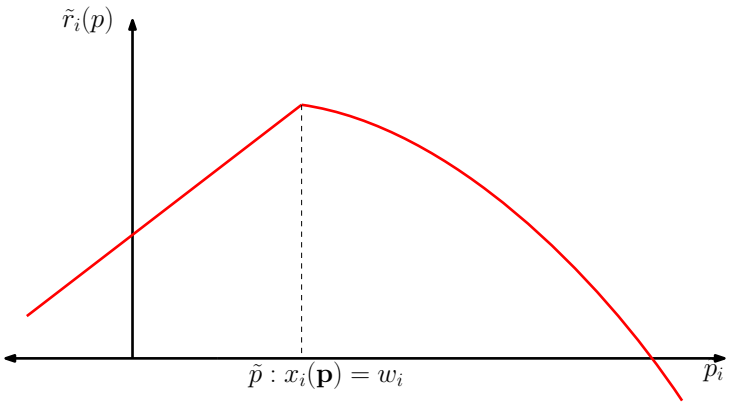


FIGURE 5.1: Revenue in log scale

$\tilde{r}_i(\mathbf{p}) = \tilde{p}_i + \min\{\tilde{x}_i(\mathbf{p}), \tilde{w}_i\}$ is concave (see Fig. 5.1). Furthermore, the revenue optimizing price of a seller is also the price for which her excess demand is zero. This implies that for a market in equilibrium, each seller obtains the maximum revenue possible given the prices of other goods. We make these ideas more formal in the following claim.

Claim 5.1. *Assuming the buyer utilities satisfy the CES property with* $0 < \rho < 1$*, then if the prices of all other goods are fixed to* p^t_{-i}*:*

(a) The log-revenue, $\tilde{r}_i(p_i, p^t_{-i})$*, of seller* i *is concave in* $\tilde{p}_i$*.*

(b) There exists a unique price p^*_i *that maximizes the revenue of seller* i*.*

(c) This maximum revenue is achieved when excess demand is zero.

Proof. We prove the concavity by a simple second derivative test. Taking the derivative of log-revenue with respect to $\tilde{p}_i$, we have: $\frac{\partial \tilde{r}_i(p_i, p^t_{-i})}{\partial \tilde{p}_i} = 1$ when for the chosen price, $x_i(\mathbf{p}) > w_i$ and

$$\frac{\partial \tilde{r}_i(p_i, p^t_{-i})}{\partial \tilde{p}_i} = 1 + \frac{\partial \tilde{x}_i(p)}{\partial \tilde{p}_i} = 1 - E_i(\mathbf{p})$$

where $E_i(\mathbf{p})$ denotes the own-price elasticity of demand for good i. By definition,$-E_i(\mathbf{p}) = \frac{\partial x_i / x_i}{\partial p_i / p_i}$ and can be expressed in terms of spending s_i, i.e., the fraction of the total money spent on item i is $-E_i(p_i) = -E + (E-1)s_i$ (see [61] for more details). Here E is the elasticity of substitution of the market.

$$\begin{aligned}
\frac{\partial^2 \tilde{r}_i(\mathbf{p}^t)}{\partial \tilde{p}_i^2} &= \frac{\partial}{\partial \tilde{p}_i}\left(1 - E + (E-1)s_i\right) \\
&= (E-1)\frac{\partial s_i}{\partial \tilde{p}_i} = p_i(E-1)\frac{\partial p_i x_i}{\partial p_i} \\
&= p_i(E-1)\left(x_i + p_i \cdot \frac{\partial x_i}{\partial p_i}\right) \\
&= p_i(E-1)\left(x_i - p_i \cdot E_i(p) \cdot \frac{x_i}{p_i}\right) \\
&= (E-1)\cdot s_i\left(1 - E_i(p)\right) \;<\; 0
\end{aligned}$$

The last inequality follows since $E_i(\mathbf{p}) > 1$ for any price vector $\mathbf{p}$. By the gross-substitutes property, keeping prices of all other items fixed, the demand for item i decreases monotonically with increase in p_i, i.e., there exists a unique price p_i^* which simultaneously ensures zero excess demand and maximum revenue. □

Competitive Equilibrium and Convex Potential Function. For the Fisher markets we consider in this chapter, there exists a convex potential function which can be interpreted as a *measure of distance* from the equilibrium configuration that guarantees market clearing. For such a market $\mathcal{M}$ with price and supply vectors $\mathbf{p}$ and $\mathbf{w}$ respectively, the potential function is defined as follows:

$$\begin{aligned}
\Psi_{\mathcal{M}}(\mathbf{w},\mathbf{p}) &= \Phi_{\mathcal{M}}(\mathbf{w},\mathbf{p}) - \Phi_{\mathcal{M}}(\mathbf{w},\mathbf{p}^*) \\
\text{where} \qquad \Phi_{\mathcal{M}}(\mathbf{w},\mathbf{p}) &= \mathbf{w}\cdot\mathbf{p} \;-\; \sum_{j=1}^{m} b_j \cdot \ln\left(\sum_{k=1}^{n}(a_{jk})^{1-c}(p_k)^c\right)^{1/c},
\end{aligned} \tag{5.1}$$

and $c = \frac{\rho}{\rho-1}$.

We denote by $\mathbf{p}^*$ an *equilibrium price vector*, i.e., a price vector at which the demand for each good is exactly equal to its supply. It was shown in [55] that Ψ is convex for all prices and minimized at $\mathbf{p}^*$. Hence, $\Psi_{\mathcal{M}}(\mathbf{w},\mathbf{p})$ can be interpreted as a distance measure. Henceforth, when clear from the context, we drop the subscript $\mathcal{M}$.

A notable property of this potential function is that its gradient with respect to a price vector is exactly the negative excess demand vector observed for that price, i.e., $\nabla_i \Psi(\mathbf{w},\mathbf{p}) = -z_i(\mathbf{p})$. We rely on this property in the next section to prove bounds on revenue loss.

If each seller in a market updates the price of her good according to the following *tatonnement update rule*

$$p_i^{t+1} \leftarrow p_i^t \exp\left(\gamma\left(x_i(\mathbf{p}^t) - w_i\right)\right) \quad \text{for all } i \in \mathcal{G}. \tag{5.2}$$

where γ is a suitable fixed parameter, then Ψ is guaranteed to decrease by a constant factor. Over a period of rounds, the potential converges linearly to the minimum. We state the result in a simplified form.

Theorem 5.2 (Theorem 43, [55])**.** *For all gross-substitute CES markets, for the sequence of prices* $\mathbf{p}^t$ *defined by the update step (5.2)*

$$\Psi\left(\mathbf{w}, \mathbf{p}^{t+1}\right) \leq (1-\delta) \cdot \Psi\left(\mathbf{w}, \mathbf{p}^t\right),$$

where $\mathbf{p}^*$ *is the equilibrium price vector and* δ *depends on the initial price vector* $\mathbf{p}^0$ *and market parameters.*

5.2.2 A Constant Bound on Total Revenue Loss

Depending on the initial price vector $\mathbf{p}^0$, we define a set $\mathcal{P}$ of prices such that for a market with static supply, the prices $\mathbf{p}^t$ chosen by the sellers and their corresponding best response prices $\mathbf{p}^{*,t}$ at any round $t \geq 1$ are in $\mathcal{P}$. Since the price updates are deterministic functions, it is clear that there always exists such a set $\mathcal{P}$. In the following, we use this fact to show a smoothness property on the potential function.

Proposition 5.3. *There is a constant* $\beta_{\mathcal{P},\mathcal{M}}$ *depending on the price set* $\mathcal{P}$ *and the market* $\mathcal{M}$ *such that the potential function* $\Psi_{\mathcal{M}}$ *is* $\beta_{\mathcal{P},\mathcal{M}}$*-smooth convex for all* $\mathbf{p} \in \mathcal{P}$*.*

We defer the proof of this proposition to Section 5.4.1. We now connect the loss in revenue of any seller i at round t to the *excess demand* vector observed. Let p_i^+ be the maximum price that seller i will set for her good in the course of the price updates. Moreover, we define

$$S_i(\mathcal{P}) = \max_{\mathbf{p} \in \mathcal{P}} \sum_j \frac{\sum_{k \neq i} a_{jk}^{1-c} \left(p_k^t\right)^c}{\sum_k a_{jk}^{1-c} \left(p_k^t\right)^c} \ .$$

Note that p_i^+ and S_i can be bounded by a fixed constant parametrized by the initial price vector $\mathbf{p}^0$.

Lemma 5.4. *Consider the price vector* $\mathbf{p}^t$ *and the excess demand vector* $\mathbf{z}^t$ *in round* t*. Then the loss in revenue of any seller* i *with respect to their best-response prices is bounded by* $\left\|\mathbf{z}^t\right\|_1 \cdot C_{\mathcal{P},\mathcal{M}}$*, where* $C_{\mathcal{P},\mathcal{M}} = \left(\frac{w_i (p_i^+)^2}{m - S_i(\mathcal{P})} + p_i^+\right)$ *is a constant depending on the space* $\mathcal{P}$ *of prices and the market* $\mathcal{M}$*.*

We defer the proof of this lemma to Section 5.4.1. As the final step, we use a known result connecting the gradient of a β-smooth convex function f to the optimum value of the function. We state the following proposition with a proof for completeness.

Proposition 5.5. *For any β-smooth convex function $f : \mathbb{R}^d \to \mathbb{R}$, the gradient at any point x is bounded by*

$$\|\nabla f(x)\|_1^2 \;\leq\; 2d\beta\,(f(x) - f^*)\,.$$

Proof. By β-smoothness we know that:

$$f(y) \;\leq\; f(x) + \nabla f(x)\cdot(y-x) + \frac{\beta}{2}\,\|y-x\|_2^2\,.$$

If $y = x - \frac{1}{\beta}\nabla f(x)$ then:

$$\begin{aligned} f\left(x - \frac{1}{\beta}\nabla f(x)\right) \;&\leq\; f(x) - \frac{\nabla f(x)\cdot\nabla f(x)}{\beta} + \frac{\beta}{2}\cdot\frac{\|\nabla f(x)\|_2^2}{\beta^2} \\ &=\; f(x) - \frac{\|\nabla f(x)\|_2^2}{2\beta}. \end{aligned}$$

Since f is a convex function, it follows that:

$$\frac{\|\nabla f(x)\|_2^2}{2\beta} \;\leq\; f(x) - f^*.$$

The proposition now follows by using the fact that $\|\cdot\|_1 \leq \sqrt{d}\,\|\cdot\|_2$. □

We are now ready to prove a bound on the cumulative revenue loss.

Theorem 5.6. *If each seller in the market uses price update rule (5.2), then the cumulative loss of any seller i over any number of rounds is bounded by*

$$L_T \;\leq\; \frac{2C_{\mathcal{P},\mathcal{M}}\sqrt{2n\beta_{\mathcal{P},\mathcal{M}}\cdot\Psi\left(\mathbf{w},\mathbf{p}^0\right)}}{\delta}.$$

Proof. The revenue loss of any seller i in round t can be bounded using Lemma 5.4 by

$$\begin{aligned} \ell_t \;&\leq\; C_{\mathcal{P},\mathcal{M}}\cdot\left\|\mathbf{z}^t\right\|_1 \\ &\overset{(a)}{\leq}\; C_{\mathcal{P},\mathcal{M}}\sqrt{2n\beta_{\mathcal{P},\mathcal{M}}\cdot\Psi\left(\mathbf{w},\mathbf{p}^t\right)} \\ &\overset{(b)}{\leq}\; C_{\mathcal{P},\mathcal{M}}\sqrt{2n\beta_{\mathcal{P},\mathcal{M}}\cdot\Psi\left(\mathbf{w},\mathbf{p}^0\right)}\cdot(1-\delta)^{t/2}\,, \end{aligned}$$

where $\mathbf{p}^0$ is the initial price. Inequality (a) follows from Proposition 5.5, inequality (b) from Theorem 5.2. Summing the loss over all rounds:

$$\begin{aligned} L_T &\leq \sum_t \ell_t \leq C_{\mathcal{P},\mathcal{M}} \sqrt{2n\beta_{\mathcal{P},\mathcal{M}} \cdot \Psi\left(\mathbf{w}, \mathbf{p}^0\right)} \cdot \sum_{t=1}^{\infty} (1-\delta)^{t/2} \\ &\leq \frac{2C_{\mathcal{P},\mathcal{M}} \sqrt{2n\beta_{\mathcal{P},\mathcal{M}} \cdot \Psi\left(\mathbf{w}, \mathbf{p}^0\right)}}{\delta}. \end{aligned}$$

□

5.2.3 Tatonnement and Myopic Revenue Optimization

In the previous section, we showed that a standard price adaptation process, which was previously proven to converge to equilibrium, also guarantees near-optimal revenue. Note, however, that this update strategy does not offer any justification for the incentive considerations of the strategic sellers from their localized perspective. Interestingly, as we show next, tatonnement can also be interpreted as an update that myopically optimizes the revenue of the sellers.

Observe that from the localized view of a seller, the log-revenue function (Fig. 5.1) is concave. Optimizing revenue with respect to this local view amounts to optimizing the log-revenue objective. Also, since the objective is concave, iterative optimization methods, such as gradient ascent are ideal candidates. However, a sequence of price updates that maximizes the log-revenue objective in this localized view would be myopic since the prices of all goods change every round leading to correspondingly different local views.

Nevertheless, direct optimization of this objective is not possible using gradient-based methods since the gradient information itself might not be available. If a seller, however, uses her excess demand $\mathbf{z}_i(\mathbf{p}) = \mathbf{x}_i(\mathbf{p}) - w_i$ as a *proxy* for the gradient, then the resulting price update step is exactly the price adaptation strategy used in the previous section:

$$\tilde{p}_i^{t+1} \leftarrow \tilde{p}_i^t + \gamma\left(x_i\left(\mathbf{p}^t\right) - w_i\right) \quad \Rightarrow \quad p_i^{t+1} \leftarrow p_i^t \exp\left(\gamma\left(x_i\left(\mathbf{p}^t\right) - w_i\right)\right).$$

It is possible to show that this price update rule converges linearly to a revenue-maximizing price, provided other prices do not change. We defer the proof of the following lemma to Section 5.4.2.

Lemma 5.7. *Assuming p_{-i} is fixed, the price update rule (5.2) of seller i converges linearly to p_i^*, where $p_i^* = \underset{p_i \in \mathcal{P}}{\operatorname{argmax}}\ \tilde{r}_i(p_i, p_{-i})$.*

In this sense, tatonnement may be interpreted as a gradient-ascent-style update from the perspective of any fixed seller, which myopically optimizes her revenue. In other words, such a price adaptation process aligns itself naturally to the strategic considerations of the individual sellers.

5.3 Repeated Markets with Dynamic Supplies

5.3.1 Preliminaries

In this section, we turn our attention to markets with dynamic supplies. A market $\mathcal{M}$ with dynamic supplies consists of the same set of buyers and sellers over T consecutive rounds. The supplies of goods may change between rounds. In particular, we assume that the supplies $\mathbf{w}^1, \mathbf{w}^2 \cdots \mathbf{w}^T$ for the T rounds are chosen by an oblivious adversary. As before, the subscript $\mathcal{M}$ captures the dependence on buyer utilities and budgets and is dropped when clear from context. In the rest of the section, we assume any norm to be a ℓ_1-norm, unless otherwise specified.

The tatonnement price updates react to excess demands and thereby attempt to stabilize the market. In this fashion, the price updates also can be seem as a predictor about the supply and demand conditions in the next round. Here we define a set $\mathcal{P}$ of prices parametrized by the initial price vector $\mathbf{p}^0$ and supplies $\{\mathbf{w}^t\}_{t=1}^T$, such that the price vector $\mathbf{p}^t$, chosen by the sellers and their corresponding best-response prices $\mathbf{p}^{*,t}$ at any round $t \geq 1$ are in $\mathcal{P}$. Let $P = \max_{\mathbf{p} \in \mathcal{P}} \|\mathbf{p}\|_\infty$ denote the maximum price that can be observed across all sellers. Recall that $B = \sum_j b_j$ is total money in the market. Using Corollary 5.19 (proof in Section 5.4.3), we can obtain a bound on the potential function in round t

$$\Psi\left(\mathbf{w}^t, \mathbf{p}^t\right) \;\leq\; (1-\delta)^{t-1}\,\Psi\left(\mathbf{w}^1, \mathbf{p}^1\right) + (P+B)\sum_{i=0}^{t-1}(1-\delta)^i \left\|\mathbf{w}^{t-i} - \mathbf{w}^{t-i-1}\right\|,$$

where δ is a constant depending on the market parameters and the price space $\mathcal{P}$. Summing over all rounds,

$$\begin{aligned}\sum_t \Psi\left(\mathbf{w}^t, \mathbf{p}^t\right) \;&\leq\; \Psi\left(\mathbf{w}^1, \mathbf{p}^1\right)\sum_{t=1}^{T}(1-\delta)^{t-1} + (P+B)\sum_{t=1}^{T}\sum_{i=0}^{t-1}(1-\delta)^i \left\|\mathbf{w}^{t-i} - \mathbf{w}^{t-i-1}\right\| \\ &\leq\; \frac{1}{\delta}\left((1-\delta)\Psi\left(\mathbf{w}^1, \mathbf{p}^1\right) \;+\; (P+B)\sum_{i=1}^{T}\left\|\mathbf{w}^i - \mathbf{w}^{i-1}\right\| \;\right).\end{aligned}$$

Using the connection between potential function and revenue outlined in the previous section, this directly implies bounds on the cumulative revenue loss of any seller i.

Proposition 5.8. *If each seller in a market with dynamic supplies uses the tatonnement update rule (5.2), then the cumulative loss in revenue of any seller is bounded by*

$$L_T \;\leq\; \sqrt{T}\cdot\sqrt{2nC^2_{\mathcal{P},\mathcal{M}}\cdot\beta_{\mathcal{P},\mathcal{M}}\cdot\frac{1}{\delta}\cdot\left(\sum_{t=1}^{T}(1-\delta)\Psi(\mathbf{w}^1,\mathbf{p}^1)+(P+B)\sum_{t=1}^{t}\|\mathbf{w}^i-\mathbf{w}^{i-1}\|\right)}\;.$$

Proof. By Proposition 5.5, the squared loss of any seller i at round t is bounded by

$$\ell_t^2 \;\leq\; 2nC^2_{\mathcal{P},\mathcal{M}}\cdot\beta_{\mathcal{P},\mathcal{M}}\cdot\Psi\left(\mathbf{w}^t,\mathbf{p}^t\right).$$

Summing this over all rounds and using the Cauchy-Schwarz inequality:

$$\begin{aligned}\sum_{t=1}^{T}\ell_t \;&\leq\; \sqrt{2nC^2_{\mathcal{P},\mathcal{M}}\cdot\beta_{\mathcal{P},\mathcal{M}}}\cdot\sum_{t=1}^{T}\sqrt{\Psi\left(\mathbf{w}^t,\mathbf{p}^t\right)}\\ &\leq\; \sqrt{2nC^2_{\mathcal{P},\mathcal{M}}\cdot\beta_{\mathcal{P},\mathcal{M}}\cdot\frac{T}{\delta}\cdot\left(\sum_{t=1}^{T}(1-\delta)\Psi(\mathbf{w}^1,\mathbf{p}^1)+(P+B)\sum_{t=1}^{t}\|\mathbf{w}^i-\mathbf{w}^{i-1}\|\right)}\;.\end{aligned}$$

□

Note that we lose the factor of $\sqrt{T}$ when simplifying the bound with Cauchy-Schwarz, which is not necessary if supplies are identical.

5.3.2 Tatonnement with Supply Estimation

The tatonnement rule relies only on the supplies in the current round to predict the price in the subsequent round. Instead, in markets where supplies are dynamic, and potentially adversarial, it can be profitable for sellers to rely on a learning algorithm to generate a supply estimator. Towards this end, we study tatonnement-style price update (as in Section 5.2) coupled with a supply-predicting learning algorithm. We show that such a price dynamic stabilizes the market, in terms of the prices offered by the sellers, and ensures that the average potential of the market stays bounded. Again, via the relation of revenue loss and potential function, we provide concrete bounds on the revenue loss for such markets.

The sellers use the same price adaptation approach except that the supply parameter used in the update step is now chosen from a learning algorithm. More precisely, the update step is:

$$p_j^{t+1} \;=\; p_j^t\cdot\exp\left(\gamma\left(x_j(\mathbf{p}^t)-\hat{w}_j^t\right)\right), \tag{5.3}$$

where $\hat{w}_j^t$ is obtained from a follow-the-regularized-leader strategy as below:

$$\hat{w}_j^t \;=\; \operatorname*{argmin}_{w\in\mathcal{W}} \sum_{i=1}^{t-1} \left|w_j^t - w\right| + \frac{w^2}{2\eta_t}. \tag{5.4}$$

As before, we define a set $\mathcal{P}$ of prices parametrized by an initial price vector $\mathbf{p}^0$ and supplies $\{\mathbf{w}^t\}_{t=1}^T$ such that the price vector $\mathbf{p}^t$ chosen by the sellers and their corresponding best response prices $\mathbf{p}^{*,t}$ at any round $t \geq 1$ are in $\mathcal{P}$. Moreover, $P = \max_{\mathbf{p}\in\mathcal{P}} \|\mathbf{p}\|_\infty$ is the maximum price that can be observed across all sellers.

Theorem 5.9. *Let $\mathbf{w}^1, \mathbf{w}^2 \cdots \mathbf{w}^T$ denote the supplies chosen by an oblivious adversary. If each seller in the market predicts her supply using update (5.4) and uses it in the price update (5.3), then*

$$\sum_{t=1}^{T} \Psi\left(\mathbf{w}^t, \mathbf{p}^t\right) \;\leq\; \sum_{t=1}^{T} (1-\delta)^{t-1}\, \Psi\left(\mathbf{w}^1, \mathbf{p}^1\right) \;+\; (P+B) \sum_{t=1}^{T} \left\|\mathbf{w}^t - \mathbf{w}^*\right\| \;+\; O\left(\sqrt{T}\right),$$

where δ is a constant depending on the market parameters and the price space $\mathcal{P}$.

Proof. Recall the definition of $\Psi(\mathbf{w}, \mathbf{p}) = \Phi(\mathbf{w}, \mathbf{p}) - \Phi^*(\mathbf{w}, \mathbf{p})$. For any vector $\hat{\mathbf{w}}^t$, we can express Φ at round t as follows:

$$\begin{aligned}\Phi\left(\mathbf{w}^t, \mathbf{p}^t\right) \;&=\; \mathbf{w}^t \cdot \mathbf{p}^t \;-\; \sum_{j=1}^{m} b_j \cdot \ln\left(\sum_{k=1}^{n} (a_{jk})^{1-c} (p_k)^c\right)^{1/c} \\ &=\; \Phi\left(\hat{\mathbf{w}}^t, \mathbf{p}^t\right) + \mathbf{p}^t \cdot \left(\mathbf{w}^t - \hat{\mathbf{w}}^t\right).\end{aligned}$$

A direct calculation (see Lemma 5.18 in Section 5.4.3) shows that

$$\Psi\left(\hat{\mathbf{w}}^t, \mathbf{p}^t\right) \;\leq\; (1-\delta)\left[\Phi\left(\hat{\mathbf{w}}^{t-1}, \mathbf{p}^{t-1}\right) - \Phi^*\left(\hat{\mathbf{w}}^{t-1}\right)\right] + (P+B)\left\|\hat{\mathbf{w}}^t - \hat{\mathbf{w}}^{t-1}\right\| \;. \tag{5.5}$$

Using this above,

$$\begin{aligned}&\Psi\left(\mathbf{w}^t, \mathbf{p}^t\right) \\ &\leq (1-\delta)\,\Psi\left(\hat{\mathbf{w}}^{t-1}, \mathbf{p}^{t-1}\right) + (P+B)\left\|\hat{\mathbf{w}}^t - \hat{\mathbf{w}}^{t-1}\right\| + \mathbf{p}^t \cdot \left(\mathbf{w}^t - \hat{\mathbf{w}}^t\right) + \Phi^*(\hat{\mathbf{w}}^t) - \Phi^*(\mathbf{w}^t) \\ &\overset{(a)}{\leq}\; (1-\delta)\,\Psi\left(\hat{\mathbf{w}}^{t-1}, \mathbf{p}^{t-1}\right) + (P+B)\left(\left\|\hat{\mathbf{w}}^t - \hat{\mathbf{w}}^{t-1}\right\| + \left\|\mathbf{w}^t - \hat{\mathbf{w}}^t\right\|\right) \\ &\overset{(b)}{\leq}\; (1-\delta)^{t-1}\,\Psi\left(\hat{\mathbf{w}}^1, \mathbf{p}^1\right) + (P+B)\left(\sum_{i=0}^{t-1} (1-\delta)^i \left\|\hat{\mathbf{w}}^{t-i} - \hat{\mathbf{w}}^{t-i-1}\right\|_1 + \left\|\mathbf{w}^t - \hat{\mathbf{w}}^t\right\|_1\right).\end{aligned}$$

The inequality (a) uses the fact that $\mathbf{w} \cdot \mathbf{p} \leq \|\mathbf{p}\|_\infty \|\mathbf{w}\|_1$ and Lemma 5.22. Inequality (b) follows by applying (5.5) recursively. Hence, the potential at round t is bounded as

a function of all supplies seen until round $t-1$. Summing the potential over all rounds:

$$\sum_{t=1}^{T} \Psi\left(\mathbf{w}^t, \mathbf{p}^t\right) \;\leq\; \sum_{t=1}^{T} (1-\delta)^{t-1}\, \Psi\left(\hat{\mathbf{w}}^1, \mathbf{p}^1\right)$$

$$+\;\; (P+B) \sum_{t=1}^{T} \left(\left\|\mathbf{w}^t - \hat{\mathbf{w}}^t\right\|_1 + \sum_{i=0}^{t-1} (1-\delta)^i \left\|\hat{\mathbf{w}}^{t-i} - \hat{\mathbf{w}}^{t-i-1}\right\|_1 \right)$$

$$\leq\; \sum_{t=1}^{T} (1-\delta)^{t-1}\, \Psi\left(\hat{\mathbf{w}}^1, \mathbf{p}^1\right) + (P+B) \sum_{t=1}^{T} \left\|\mathbf{w}^t - \mathbf{w}^*\right\| \;+\; C\sqrt{T},$$

where $\mathbf{w}^*$ is any fixed vector, and $C = \left(\frac{F^2+W^2}{2} \;+\; \frac{W\sqrt{n}}{\delta}\right)$, with $F = \max_t \left\|\mathbf{w}^t - \mathbf{w}^*\right\|_2$ and $W = \max_t \left\|\mathbf{w}^t\right\|_2$. The second inequality is from Corollary 5.20 in Section 5.4.4. □

The theorem shows that the *average* potential of the market (alternatively, the *distance* to an underlying "average equilibrium" state) is governed by the ℓ_1 distance of the supply vectors from a single optimal supply vector in hindsight, which the sellers could have chosen as part of their price updates.

Remark 5.10. In update 5.4, instead of $\frac{w^2}{2\eta_t}$ one could also use a different regularizer as long as it is strongly convex and satisfies the following property:

$$\sum_{t=1}^{T} \left(\left\|\mathbf{w}^t - \hat{\mathbf{w}}^t\right\| + \sum_{i=0}^{t-1} (1-\delta)^i \left\|\hat{\mathbf{w}}^{t-i} - \hat{\mathbf{w}}^{t-i-1}\right\| \right) \;\leq\; \sum_t \left\|\mathbf{w}^t - \mathbf{w}^*\right\| \;+\; O\left(\sqrt{T}\right).$$

This is the core property that ensures that the analysis goes through.

Remark 5.11. Recall that our price update is $p_j^{t+1} \leftarrow p_j^t \cdot \exp\left(\gamma z_j\right)$, where γ is a *constant*. In Cheung et al. [55] use a time-dependent step size

$$\gamma_j^t = \left[5 \cdot \max\left\{1, \frac{1-c_{min}}{2}\right\} \cdot \max\left\{1, x_j^t\right\}. \right]^{-1}.$$

However, their analysis holds also for a constant step size provided that it is small enough.

Given Theorem 5.9, we can now bound the cumulative loss in the revenue of any seller i as above. The proof is a straightforward adaptation of the one for Proposition 5.8.

Theorem 5.12. *If each seller in a market with dynamic supplies uses price update 5.3, then the cumulative loss in revenue of any seller is bounded by*

$$L_T \;\leq\; \sqrt{T} \cdot \sqrt{2nC_{\mathcal{P},\mathcal{M}}^2 \cdot \beta_{\mathcal{P},\mathcal{M}} \cdot \left(\frac{\psi\left(\mathbf{w}^1, \mathbf{p}^1\right)}{\delta} \;+\; (P+B) \sum_{t=1}^{T} \left\|\mathbf{w}^t - \mathbf{w}^*\right\| \;+\; O\left(\sqrt{T}\right) \right)}.$$

Remark 5.13. If $\sum_{t=1}^{T} \left\|\mathbf{w}^t - \mathbf{w}^*\right\| = O(T^\alpha)$ for $\alpha \in [1/2, 1)$, then one can observe that the cumulative loss incurred by any seller increases only sub-linearly in T. In particular, it follows that $\sum_{t=1}^{T} \ell_t \;\leq\; O\left(T^{(1+\alpha)/2}\right)$, i.e., this bound improves smoothly with the *benignity* of the supply sequence observed.

5.4 Omitted Proofs

5.4.1 A Constant Bound on Total Revenue Loss

Lemma 5.4. *Consider the price vector $\mathbf{p}^t$ and the excess demand vector $\mathbf{z}^t$ in round t. Then the loss in revenue of any seller i with respect to their best-response prices is bounded by $\left\|\mathbf{z}^t\right\|_1 \cdot C_{\mathcal{P},\mathcal{M}}$, where $C_{\mathcal{P},\mathcal{M}} = \left(\frac{w_i (p_i^+)^2}{m - S_i(\mathcal{P})} \;+\; p_i^+\right)$ is a constant depending on the space $\mathcal{P}$ of prices and the market $\mathcal{M}$.*

Proof. Let the price vector at round t be $\mathbf{p}^t$, and let $p_i^{*,t}$ denote the price that maximizes the revenue of seller i keeping the other prices p_{i-} fixed. Hence, $x_i\left(p_i^{*,t}, p_{-i}\right) = w_i$. The revenue loss of seller i at round t, ℓ_i^t, is given by

$$\begin{aligned} \ell_i^t \;&=\; p_i^{*,t} \cdot w_i \;-\; p_i^t \cdot \min\{w_i, x_i^t(\mathbf{p})\} \\ &\leq\; p_i^{*,t} \cdot w_i \;-\; p_i^t \left(w_i - \left|z_i^t\right|\right) \\ &\leq\; w_i(p_i^{*,t} - p_i^t) \;+\; p_i^+ \left|z_i^t\right|. \end{aligned} \tag{5.6}$$

Note here that $\left|z_i^t\right| \;\leq\; \left\|\mathbf{z}^t\right\|_1$ and therefore we need to bound only $p_i^{*,t} - p_i^t$. Suppose $p_i^{*,t} \leq p_i^t$, then the revenue loss can simply be upper bounded by $p_1^+ \cdot \left\|\mathbf{z}^t\right\|_1$. Hence, for the remainder of the proof we assume $p_i^{*,t} > p_i^t$

Note that $p_i^{*,t} > p_i^t$ is equivalent to $x_i(\mathbf{p}^t) > w_i$, which is equivalent to $z_i^t > 0$. Therefore,

$$z_i^t \;=\; x_i\left(p_i^t, p_{-i}^t\right) - x_i\left(p_i^{*,t}, p_{-i}^t\right).$$

Using the definition of CES functions, we can explicitly express the demands by

$$\begin{aligned} z_i^t \;&=\; \sum_j \left(\frac{a_{ji}^{1-c} \left(p_i^t\right)^{c-1}}{\sum\limits_{k \neq i} a_{jk}^{1-c} \left(p_k^t\right)^c + a_{ji}^{1-c} \left(p_i^t\right)^c} \;-\; \frac{a_{ji}^{1-c} \left(p_i^{*,t}\right)^{c-1}}{\sum\limits_{k \neq i} a_{jk}^{1-c} \left(p_k^t\right)^c + a_{ji}^{1-c} \left(p_i^{*,t}\right)^c} \right) \\ &=\; \sum_j \left(\frac{\left(p_i^t\right)^{c-1}}{K_j^t + \left(p_i^t\right)^c} \;-\; \frac{\left(p_i^{*,t}\right)^{c-1}}{K_j^t + \left(p_i^{*,t}\right)^c} \right) \end{aligned}$$

where $K_j^t = \frac{\sum_{k \neq i} a_{jk}^{1-c} \left(p_k^t\right)^c}{a_{ji}^{1-c}}$. By letting $d = -c$:

$$
\begin{aligned}
z_i^t &= \sum_j \left(\frac{\left(p_i^t\right)^{-d-1}}{K_j^t + \left(p_i^t\right)^{-d}} - \frac{\left(p_i^{*,t}\right)^{-d-1}}{K_j^t + \left(p_i^{*,t}\right)^{-d}} \right) \\
&= \sum_j \left(\frac{1}{p_i^t} \cdot \frac{1}{1 + K_j^t \left(p_i^t\right)^d} - \frac{1}{p_i^{*,t}} \cdot \frac{1}{1 + K_j^t \left(p_i^{*,t}\right)^d} \right) \\
&= \sum_j \left(\frac{1}{p_i^t} \left(1 - \frac{K_j^t \left(p_i^t\right)^d}{1 + K_j^t \left(p_i^t\right)^d} \right) - \frac{1}{p_i^{*,t}} \left(1 - \frac{K_j^t \left(p_i^{*,t}\right)^d}{1 + K_j^t \left(p_i^{*,t}\right)^d} \right) \right) \\
&= \sum_j \left[\left(\frac{1}{p_i^t} - \frac{1}{p_i^{*,t}} \right) + K_j^t \left(\frac{\left(p_i^{*,t}\right)^d \cdot \left(p_i^{*,t}\right)^{-1}}{1 + K_j^t \left(p_i^{*,t}\right)^d} - \frac{\left(p_i^t\right)^d \cdot \left(p_i^t\right)^{-1}}{1 + K_j^t \left(p_i^t\right)^d} \right) \right]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{p_i^t \cdot p_i^{*,t}} \sum_j \left[\left(p_i^{*,t} - p_i^t \right) + K_j^t \left(\frac{\left(p_i^{*,t}\right)^d \cdot p_i^t}{1 + K_j^t \left(p_i^{*,t}\right)^d} - \frac{\left(p_i^t\right)^d \cdot p_i^{*,t}}{1 + K_j^t \left(p_i^t\right)^d} \right) \right] \\
&= \frac{1}{p_i^t \cdot p_i^{*,t}} \sum_j \left[\left(p_i^{*,t} - p_i^t \right) + K_j^t \left(\frac{p_i^t}{\left(p_i^t\right)^{-d} + K_j^t} - \frac{p_i^{*,t}}{\left(p_i^{*,t}\right)^{-d} + K_j^t} \right) \right] \\
&\geq \frac{1}{p_i^t \cdot p_i^{*,t}} \sum_j \left[\left(p_i^{*,t} - p_i^t \right) + K_j^t \left(\frac{p_i^t}{\left(p_i^{*,t}\right)^{-d} + K_j^t} - \frac{p_i^{*,t}}{\left(p_i^{*,t}\right)^{-d} + K_j^t} \right) \right]
\end{aligned}
$$

This implies

$$
z_i^t \geq \frac{p_i^{*,t} - p_i^t}{p_i^t \cdot p_i^{*,t}} \sum_j \left[1 - \frac{K_j^t}{\left(p_i^{*,t}\right)^{-d} + K_j^t} \right] \geq \frac{p_i^{*,t} - p_i^t}{p_i^t \cdot p_i^{*,t}} \left(m - S_i(\mathcal{P}) \right).
$$

Rearranging this yields

$$
p_i^{*,t} - p_i^t \leq \frac{z_i^t \cdot (p_i^+)^2}{m - S_i(\mathcal{P})}
$$

and substituting this back in (5.6) proves the lemma. □

Proposition 5.14. *There is a constant $\beta_{\mathcal{P},\mathcal{M}}$ depending on the price set $\mathcal{P}$ and the market $\mathcal{M}$ such that the potential function $\Psi_{\mathcal{M}}$ is $\beta_{\mathcal{P},\mathcal{M}}$-smooth convex for all $\mathbf{p} \in \mathcal{P}$.*

Proof. Note that the Jacobian $\mathcal{J}_{\mathbf{z}}$ of $\nabla\Psi = -\mathbf{z}(\mathbf{p})$ can be expressed as:

$$\begin{bmatrix} -\frac{\partial x_1(\mathbf{p})}{\partial p_1} & -\frac{\partial x_2(\mathbf{p})}{\partial p_1} & \dots & -\frac{\partial x_n(\mathbf{p})}{\partial p_1} \\ -\frac{\partial x_1(\mathbf{p})}{\partial p_2} & \vdots & \ddots & -\frac{\partial x_n(\mathbf{p})}{\partial p_2} \\ \vdots & & & \\ -\frac{\partial x_1(\mathbf{p})}{\partial p_n} & \dots & & -\frac{\partial x_n(\mathbf{p})}{\partial p_n} \end{bmatrix} = \begin{bmatrix} -E_{11}\frac{x_1(\mathbf{p})}{p_1} & -E_{21}\frac{x_2(\mathbf{p})}{p_1} & \dots & -E_{n1}\frac{x_n(\mathbf{p})}{p_1} \\ -E_{12}\frac{x_1(\mathbf{p})}{p_2} & \vdots & \ddots & -E_{n2}\frac{x_n(\mathbf{p})}{p_2} \\ \vdots & & & \\ -E_{1n}\frac{x_1(\mathbf{p})}{p_n} & \dots & & -E_{nn}\frac{x_n(\mathbf{p})}{p_n} \end{bmatrix}$$

where E_{ij} denotes the elasticity of the demand of good i with respect to price of good j. We note that for CES substitutes utilities, the elasticity of demand of any good with respect to its own price is negative whereas with respect to prices of other goods is positive. More precisely, $E_{ii} = -E + (E-1)s_i$ and $E_{ij} = (E-1)s_j$, where E is the elasticity of substitution and s_i denotes the fraction of total money in the market spent on good i.

One can easily verify that the matrix is symmetric. We note that the diagonal terms are all positive. Since the potential function Ψ is known to be convex, $\mathcal{J}_{\mathbf{z}}$ is positive semi-definite. Next we show that $\mathcal{J}_{\mathbf{z}}$ is, in fact, positive definite and Ψ is strongly convex. To do so, we use the following result (see, e.g., [62, Chapter 7.6]).

Lemma 5.15 (Sylvester's criterion)**.** *A matrix is positive definite if all upper left $k \times k$ determinants of a symmetric matrix are positive.*

Now consider the determinant for the general case.

$$\det(A) = \underbrace{\frac{x_1 x_2 \cdots x_n}{p_1 p_2 \cdots p_n}}_{X} \cdot \begin{vmatrix} -E_{11} & -E_{21} & \dots & & -E_{n1} \\ -E_{12} & -E_{22} & -E_{32} & \dots & -E_{n2} \\ \vdots & \vdots & \vdots & & \vdots \\ -E_{1n} & -E_{2n} & \dots & -E_{n-1,n} & -E_{nn} \end{vmatrix}$$

$$= X \cdot \begin{vmatrix} E-(E-1)s_1 & -(E-1)s_1 & -(E-1)s_1 & \dots & -(E-1)s_1 \\ -(E-1)s_2 & E-(E-1)s_2 & -(E-1)s_2 & \dots & -(E-1)s_2 \\ \vdots & \vdots & \vdots & & \vdots \\ -(E-1)s_n & -(E-1)s_n & \dots & & E-(E-1)s_n \end{vmatrix}$$

$$= X(E-1)^n \prod_{i=1}^{n} s_i \begin{vmatrix} \frac{E}{(E-1)s_1}-1 & -1 & -1 & \dots & -1 \\ -1 & \frac{E}{(E-1)s_2}-1 & -1 & \dots & -1 \\ \vdots & \vdots & \vdots & & \vdots \\ -1 & -1 & \dots & -1 & \frac{E}{(E-1)s_n}-1 \end{vmatrix}$$

$$= X(E-1)^n \prod_{i=1}^{n} s_i \begin{vmatrix} \frac{E}{(E-1)s_1} & 0 & 0 & \dots & -1 \\ -\frac{E}{(E-1)s_2} & \frac{E}{(E-1)s_2} & 0 & \dots & -1 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & -\frac{E}{(E-1)s_n} & \frac{E}{(E-1)s_n}-1 \end{vmatrix}$$

We now execute transformations $C_1 \leftarrow C_1 - C_2$, $C_2 \leftarrow C_2 - C_3, \cdots, C_{n-1} \leftarrow C_{n-1} - C_n$ and take out factors $E/(E-1)$ from the resulting columns. This yields

$$\det(A) \;=\; XE^n(E-1)\prod_{i=1}^{n} s_i \begin{vmatrix} 1/s_1 & 0 & 0 & 0 & \dots & -1 \\ -1/s_2 & 1/s_2 & 0 & 0 & \dots & -1 \\ 0 & -1/s_3 & 1/s_3 & 0 & \dots & -1 \\ \vdots & \vdots & \vdots & & & \vdots \\ 0 & 0 & \dots & & -1/s_n & \frac{E}{(E-1)s_n} - 1 \end{vmatrix}$$

This determinant can be shown to be always greater than zero by inductively computing $i \times i$ *lower right* sub-determinants. As a base case, note that

$$A_{2\times 2} \;=\; \begin{vmatrix} 1/s_{n-1} & -1 \\ -1/s_n & \frac{E}{(E-1)s_n} - 1 \end{vmatrix} \;=\; \frac{E}{(E-1)s_n s_{n-1}} - \frac{1}{s_{n-1}} - \frac{1}{s_n} \;>\; 0.$$

For all $i > 2$, one can show that:

$$\begin{aligned} A_{i\times i} \;&=\; \frac{1}{s_{n-i}} \cdot A_{(i-1)\times(i-1)} \;-\; \frac{1}{s_n s_{n-1} \cdots s_{n-i+1}} \\ &=\; \frac{1}{s_n s_{n-1} \cdots s_{n-i}} \left(\frac{E}{E-1} \;-\; \sum_{j=n-i}^{n} s_j \right), \end{aligned}$$

and hence $A_{n\times n}$ is always positive. Interestingly, with the same set of determinant transformations, one can show that *all* upper left $k \times k$ sub determinants are also strictly greater than zero. By Lemma 5.15, we can therefore claim that $\mathcal{J}_{\mathbf{z}}$ is positive definite.

Since Ψ is strongly, and also strictly, convex and since Ψ is differentiable everywhere in $\mathcal{P}$, there must exist a constant $\beta_{\mathcal{P},\mathcal{M}} > 0$ such that Ψ is $\beta_{\mathcal{P},\mathcal{M}}$ smooth. □

5.4.2 Tatonnement and Myopic Revenue Optimization

Lemma 5.7. *Assuming p_{-i} is fixed, the price update rule (5.2) of seller i converges linearly to p_i^*, where $p_i^* = \underset{p_i \in \mathcal{P}}{\operatorname{argmax}}\ \tilde{r}_i(p_i, p_{-i})$.*

Proof. Let $\phi_i(\tilde{p}) = x_i(p, p_{-i}) - w_i$ for some unknown but fixed p_{-i}. The tatonnement update step (5.2) can then be written as: $\tilde{p}_i^{t+1} \;\leftarrow\; \tilde{p}_i^t + \gamma\phi_i(\tilde{p}_i^t)$ or more succinctly as

$\tilde{p}_i^{t+1} \leftarrow \mathcal{F}\left(\tilde{p}_i^t\right)$ where $\mathcal{F}$ is the operator defined as

$$\mathcal{F} : \mathbb{R} \to \mathbb{R} : \tilde{p} \mapsto \tilde{p} + \gamma\phi_i(\tilde{p}). \tag{5.7}$$

The following properties of the function ϕ_i, used in the subsequent claim, can be easily verified:

1. ϕ_i is differentiable over the range of prices $\mathcal{P}$.
2. $\tilde{p}_i < \tilde{p}_i' \Rightarrow \phi_i(\tilde{p}_i) > \phi_i(\tilde{p}_i')$, i.e. ϕ_i is a strictly decreasing function in $\tilde{p}_i$.
3. $\phi_i(\tilde{p}_i^*) = 0$, where $p_i^* = \underset{p_i \in \mathcal{P}}{\operatorname{argmax}}\, r_i(p_i, p_{-i})$.

Using these properties, we characterize the operator $\mathcal{F}$ as follows:

Claim 5.16. *The operator $\mathcal{F}$ defined in* (5.7) *satisfies the following properties:*

a) $\phi_i(\tilde{p}_i) = 0$ if and only if $\tilde{p}_i$ is a fixed point of operator $\mathcal{F}$.

b) For any two prices p^1 and p^2, $\left|\mathcal{F}(\tilde{p}^1) - \mathcal{F}(\tilde{p}^2)\right| \leq \rho\left|\tilde{p}^1 - \tilde{p}^2\right|$ for $\rho \in [0, 1)$, i.e. $\mathcal{F}$ is a contraction mapping.

Proof. Let $\tilde{p}_i$ be a price such that $\phi_i(\tilde{p}_i) = 0$. From the update rule it follows directly that $\mathcal{F}(\tilde{p}_i) = \tilde{p}_i$ and hence a fixed point. Conversely, if $\tilde{p}_i$ is a fixed point of operator $\mathcal{F}$, then $x = x + \gamma\phi_i(\tilde{p}_i)$ implying $\phi_i(\tilde{p}_i) = 0$ since $\gamma \neq 0$. For the second part,

$$\begin{aligned}
\left|\mathcal{F}(\tilde{p}^1) - \mathcal{F}(\tilde{p}^2)\right| &= \left|\tilde{p}^1 + \gamma\phi_i(\tilde{p}^1) \;-\; \tilde{p}^2 - \gamma\phi_i(\tilde{p}^2)\right| \\
&= \left|(\tilde{p}^1 - \tilde{p}^2) \;+\; \gamma\left(\phi_i(\tilde{p}^1) - \phi_i(\tilde{p}^2)\right)\right| \\
&\overset{(a)}{=} \left|(\tilde{p}^1 - \tilde{p}^2) \;+\; \eta\nabla\phi_i(\tilde{p}^z)(\tilde{p}^1 - \tilde{p}^2)\right| \\
&= \left|(\tilde{p}^1 - \tilde{p}^2)\left(1 + \gamma\nabla\phi_i(\tilde{p}^z)\right)\right| \\
&\overset{(b)}{\leq} \rho\left|\tilde{p}^1 - \tilde{p}^2\right| \text{ for } \rho \in [0, 1).
\end{aligned}$$

The equality (a) follows from the mean value theorem for scalar functions [63] and inequality (b) follows from the fact that ϕ_i is differentiable over the entire domain and is strictly decreasing. □

The lemma now follows directly from the Banach fixed-point theorem (See Chapter 3, [63] for more details), which states that if an operator $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ is a contraction

mapping in a metric space, i.e., $|\mathcal{F}(x) - \mathcal{F}(y)| \leq \rho\,|(x-y)|$ for $\rho \in [0,1)$, then $\mathcal{F}$ admits a unique fixed point and the iterate $\mathbf{x}^t$ satisfies:

$$\left\|\mathbf{x}^t - \mathbf{x}^*\right\| \;\leq\; \rho^t \left\|\mathbf{x}^0 - \mathbf{x}^*\right\|.$$

□

Remark 5.17. Excess demand is not the only possible proxy for the gradient that may be used by sellers. In principle, the sellers may use any proxy as long as the resulting ϕ_i function satisfies the same properties as in the lemma above. For example, the sellers may use $\ln\left(\frac{x_i(\mathbf{p})}{w_i}\right)$ resulting in the price update step $p_i^{t+1} \leftarrow p_i^t \left(x_i\left(\mathbf{p}\right)/w_i\right)^\gamma$. Indeed, this is very closely related[1] to the update rule studied by [53] to prove fast convergence of CES substitutes markets to equilibrium, albeit using a different potential function.

5.4.3 Bounding Potential with Dynamic Supplies

In this section, we present bounds on the convex potential function (5.1) of the market when each seller updates the prices of her goods using standard tatonnement. It is implicitly assumed here that the set of sellers, buyers and their budgets and utilities stay the same throughout. Let $Q_j(\mathbf{p}) = \ln\left(\sum_{k=1}^n (a_{jk})^{1-c}(p_k)^c\right)^{1/c}$.

Lemma 5.18. *Let $\mathbf{w}^{t+1}, \mathbf{w}^t$ and $\mathbf{p}^{t+1}, \mathbf{p}^t$ denote the supplies observed and the corresponding prices chosen according to the tatonnement rule (5.2) in consecutive rounds. Then*

$$\Psi\left(\mathbf{w}^{t+1}, \mathbf{p}^{t+1}\right) \;\leq\; (1-\delta)\,\Psi(\mathbf{w}^t, \mathbf{p}^t) - \Phi^*\left(\mathbf{w}^t\right) \;+\; (P+B)\left\|\mathbf{w}^{t+1} - \mathbf{w}^t\right\| \quad .$$

[1] The actual price update studied by [53] is $p_i^{t+1} \leftarrow p_i^t\left(1 + \lambda \min\left(1, \frac{x_i - w_i}{w_i}\right)\right)$.

Proof.

$$
\begin{aligned}
&\Psi(\mathbf{w}^{t+1}, \mathbf{p}^{t+1}) \\
&= \Phi(\mathbf{w}^{t+1}, \mathbf{p}^{t+1}) - \Phi^*(\mathbf{w}^{t+1}) \\
&\leq \left(\mathbf{w}^t \cdot \mathbf{p}^{t+1} + \sum_j b_j Q_j(\mathbf{p}^{t+1}) - \Phi^*(\mathbf{w}^t) \right) + \mathbf{p}^{t+1} \cdot \left| \mathbf{w}^{t+1} - \mathbf{w}^t \right| + \Phi^*(\mathbf{w}^t) - \Phi^*(\mathbf{w}^{t+1}) \\
&= \left(\Phi(\mathbf{w}^t, \mathbf{p}^{t+1}) - \Phi^*(\mathbf{w}^t) \right) \; + \; \mathbf{p}^{t+1} \cdot \left(\mathbf{w}^{t+1} - \mathbf{w}^t \right) \; + \; \Phi^*(\mathbf{w}^t) - \Phi^*(\mathbf{w}^{t+1}) \\
&\leq (1-\delta) \left[\Phi(\mathbf{w}^t, \mathbf{p}^t) - \Phi^*(\mathbf{w}^t) \right] \; + \; \left\| \mathbf{p}^{t+1} \right\|_\infty \left\| \mathbf{w}^{t+1} - \mathbf{w}^t \right\|_1 \; + \; \Phi^*(\mathbf{w}^t) - \Phi^*(\mathbf{w}^{t+1}) \\
&\leq (1-\delta)\, \Psi(\mathbf{w}^t, \mathbf{p}^t) \; + \; P \left\| \mathbf{w}^{t+1} - \mathbf{w}^t \right\|_1 \; + \; \Phi^*(\mathbf{w}^t) - \Phi^*(\mathbf{w}^{t+1})
\end{aligned}
$$

By Lemma 5.22, we have already shown that $\Phi^*(\mathbf{w}^t) - \Phi^*(\mathbf{w}^{t+1}) \;\leq\; B \left\| \mathbf{w}^{t+1} - \mathbf{w}^t \right\|_1$. The lemma follows from this. □

By a straightforward application of this lemma, one can also bound the potential at any round T.

Corollary 5.19. *Let $\{\mathbf{w}^i\}_{i=1}^T$ and $\{\mathbf{p}^i\}_{i=1}^T$ denote the sequence of supplies observed and the corresponding prices chosen according to the tatonnement rule (5.2) in consecutive rounds. Then:*

$$
\Psi\left(\mathbf{w}^T, \mathbf{p}^T\right) \;\leq\; (1-\delta)^{T-1}\, \Psi\left(\mathbf{w}^1, \mathbf{p}^1\right) + (P+B) \sum_{i=0}^{T-1} (1-\delta)^i \left\| \mathbf{w}^{T-i} - \mathbf{w}^{T-i-1} \right\|.
$$

5.4.4 Regret-Style Bounds for Tatonnement with Supply Estimation

Recall that $\mathbf{w}^*$ is any fixed vector and $C = \left(\frac{F^2+W^2}{2} + \frac{W\sqrt{n}}{\delta} \right)$, with $F = \max_t \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2$ and $W = \max_t \left\| \mathbf{w}^t \right\|_2$.

Corollary 5.20. *If all sellers use a fixed step size $\eta = 1/\sqrt{T}$ for their supply prediction(5.4), then*

$$
\sum_{t=1}^{T} \left\| \mathbf{w}^t - \hat{\mathbf{w}}^t \right\| \; + \; \sum_{i=0}^{t-1} (1-\delta)^i \left\| \hat{\mathbf{w}}^{t-i} - \hat{\mathbf{w}}^{t-i-1} \right\| \;\leq\; \sum_{t=1}^{T} \left\| \mathbf{w}^t - \mathbf{w}^* \right\| \; + \; C\sqrt{T}.
$$

This is a direct consequence of the following lemma that gives a bound for decreasing step size.

Lemma 5.21. *If the supply parameter in the tatonnement update is chosen as in Equation 5.4, then*

$$\sum_{t=1}^{T}\left(\left\|\mathbf{w}^t-\hat{\mathbf{w}}^t\right\| \;+\; \sum_{i=0}^{t-1}(1-\delta)^i\left\|\hat{\mathbf{w}}^{t-i}-\hat{\mathbf{w}}^{t-i-1}\right\|\right)$$
$$\leq\; \sum_t\left\|\mathbf{w}^t-\mathbf{w}^*\right\| \;+\; \frac{F^2}{2\eta_T}+\frac{W^2}{2}\sum_t\eta_t \;+\; W\cdot\sum_{t=1}^{T}\sum_{i=0}^{t}(1-\delta)^i\,\eta_t,$$

where $F=\max_t\left\|\mathbf{w}^t-\mathbf{w}^*\right\|_2$ *and* $W=\max_t\left\|\mathbf{w}^t\right\|_2$.

Proof. Let $f^t(\mathbf{w})=\left\|\mathbf{w}^t-\mathbf{w}\right\|$ be the loss function. Since the sellers use Equation 5.4 to choose $\hat{\mathbf{w}}^t$, this corresponds to online gradient descent being used on the sequence of functions $\{f^t\}_t$[2]. Our goal, however, is to bound the following *modified* regret:

$$R_T \;=\; \sum_{t=1}^{T}\left(f^t(\hat{\mathbf{w}}^t)+\sum_{i=0}^{t-1}(1-\delta)^i\left\|\hat{\mathbf{w}}^{t-i}-\hat{\mathbf{w}}^{t-i-1}\right\|\right)-f^t(\mathbf{w}^*), \tag{5.8}$$

which accounts for the *cost* of all previous *switches* with geometrically decreasing weights, in addition to the standard regret. From the analysis of online gradient descent [3],

$$\sum_t f^t(\hat{\mathbf{w}}^t)-f^t(\mathbf{w}^*) \;\leq\; \frac{F^2}{2\eta_T} \;+\; \frac{W^2}{2}\sum_t\eta_t, \tag{5.9}$$

where $W=\max_t\left\|\nabla f^t(\hat{\mathbf{w}}^t)\right\|_2=\max_t\left\|\mathbf{w}^t\right\|_2$ and $F=\max_t\left\|\mathbf{w}^t-\mathbf{w}^*\right\|_2$. One can directly use the online gradient descent update to bound $\left\|\hat{\mathbf{w}}^k-\hat{\mathbf{w}}^{k-1}\right\|_1$ for any round k, i.e.

$$\begin{aligned}\hat{\mathbf{w}}^{k+1} &= \hat{\mathbf{w}}^k-\eta_k\cdot\nabla f^k(\hat{\mathbf{w}}^k)\\ \left\|\hat{\mathbf{w}}^{k+1}-\hat{\mathbf{w}}^k\right\|_1 &= \eta_k\left\|\nabla f^k(\hat{\mathbf{w}}^k)\right\|_1\\ &= \eta_k\left\|\mathbf{w}^k\right\|_1 \;\leq\; \sqrt{n}\cdot\eta_k\left\|\mathbf{w}^k\right\|_2 \;\leq\; \sqrt{n}W\cdot\eta_k.\end{aligned}$$

Therefore,

$$\sum_{t=1}^{T}\sum_{i=0}^{t-1}(1-\delta)^i\left\|\hat{\mathbf{w}}^{t-i}-\hat{\mathbf{w}}^{t-i-1}\right\|_1 \;\leq\; \sqrt{n}\sum_{t=1}^{T}\sum_{i=0}^{t}(1-\delta)^i\,\eta_i\left\|\mathbf{w}^i\right\|_2.$$

[2]This is because the FTRL algorithm on a convex function with $\frac{w^2}{2\eta}$ as regularizer is known to be equivalent to online gradient descent. See [64] for more details.

Using this and (5.9) in our modified regret (5.8)

$$R_T \;\le\; \frac{F^2}{2\eta_T} \;+\; \frac{W^2}{2}\sum_t \eta_t \;+\; \sqrt{n}W\sum_{t=1}^{T}\sum_{i=0}^{t}(1-\delta)^i\,\eta_i.$$

□

Lemma 5.22. *For any supply vectors* $\mathbf{w}_1, \mathbf{w}_2 \in \mathcal{W}$,

$$\varphi^*(\mathbf{w}_1) - \varphi^*(\mathbf{w}_2) \;\le\; B\,\|\mathbf{w}_1 - \mathbf{w}_2\|_1\,.$$

Proof. Let $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$ denote the equilibrium price vectors for supplies $\mathbf{w}_1$ and $\mathbf{w}_2$, respectively.

$$\begin{aligned}
\varphi^*(\mathbf{w}_2) &= \mathbf{w}_2\mathbf{p}_2^* \;-\; f(\mathbf{p}_2^*) \\
&\ge \mathbf{w}_1\mathbf{p}_2^* \;-\; f(\mathbf{p}_2^*) \;-\; \mathbf{p}_2^*\cdot(\mathbf{w}_2 - \mathbf{w}_1) \\
&\ge \varphi^*(\mathbf{w}_1) \;-\; \|\mathbf{p}_2^*\|_\infty\,\|\mathbf{w}_2 - \mathbf{w}_1\|_1 \qquad \text{(by definition of } \varphi^*(\mathbf{w}_1)) \\
&\ge \varphi^*(\mathbf{w}_1) \;-\; B\,\|\mathbf{w}_2 - \mathbf{w}_1\|_1\,,
\end{aligned}$$

where $f(\mathbf{p}) = b\ln\left[\left(\sum_{k=1}^{n}(a_k)^{1-c}(p_k)^c\right)^{1/c}\right]$. The last inequality holds since each equilibrium price is bounded above by B, the total amount of money in the market. □

Chapter 6

Pricing via Regret Learning

6.1 Introduction

In the previous chapter we described how the well-studied price update procedure based on tatonnement not only leads the market to equilibrium but also optimizes the revenue of every seller with respect to the optimal price in hindsight. Since the behaviour of tatonnement is well understood in Fisher markets with substitutes CES utilities, we were able to use its properties together with convexity of the associated potential function to bound the loss in revenue of any seller. On second thoughts, one may argue that the assumption that *every* seller in the market follows the protocol prescribed by the tatonnement update exactly is too strong. In this chapter, we follow exactly this line of argument and explore alternative price update dynamics that can deliver similar results.

As in the previous chapter, we view the market as a set of strategic agents (the sellers) choosing successive actions (prices) in order to maximize their utility (revenue) and focus on using the existing rich tool-kit of *agnostic learning* in game-theoretic models to prove fast convergence to optimal prices. The advantages of an agnostic learning approach are multifold: Firstly, it does not rely on the precise parametric form of the underlying demand function, and secondly it can be easily extended to the case when the market parameters may change across rounds. The downside, however, being that in the best case of static markets with clean parametric representation, the algorithms might converge to optimal prices only asymptotically [65, 66]. Consequently, to measure the performance of the actions (prices) chosen by such a learning algorithm we typically compare it to a certain benchmark sequence of actions and the *regret bound* represents the loss incurred by the algorithm for not having chosen the benchmark sequence instead. This is the same benchmark as used in the previous chapter.

We base our dynamic pricing approach on the work by Syrgkanis et al [2], where the authors prove that in a game with multiple agents if each agent uses a regret-minimizing algorithm with a suitable step-size parameter and satisfying a certain technical property, then the individual regret of each agent is bounded by $O(T^{1/4})$ where T is the total number of rounds. Although the main result is proved in the discrete action setting, the authors show that the same technique can be extended to agents with continuous action sets as well. In a nutshell, these algorithms *anticipate* the utility vector for the forthcoming round and choose a price such that the cumulative utility over all previous rounds and the forthcoming one is maximized. The regret bound thus obtained holds with respect to the single best price in hindsight and is one of the benchmarks we use to measure the performance of our approach.

Related Work

The problem of learning an optimal pricing policy for various demand models and inventory constraints has been researched extensively in the last decade. However, many consider the problem of a single good with no *competition effects*. Several works [65, 67–70] study a parametric family of demand functions and design an optimal pricing policy by estimating the unknown parameters by standard techniques such as linear regression or maximum likelihood estimation. In addition, there are works [36, 71, 72] that consider Bayesian and non-parametric approaches.

Closer to the theme of this chapter there has also been a considerable amount of research about dynamic pricing in models incorporating competition, eg., [73–75]. However, most of these works consider discrete choice models of demand, where a single consumer approaches and buys a discrete bundle of goods. Moreover, they assume that every seller has a fixed inventory level in the beginning and is not replenished during the course of the algorithm. We, on the other hand, consider demand originating from a general mass of consumers with large volumes in which case, the items may be considered divisible. For a more thorough survey of the existing literature we refer the reader to [76].

6.2 Model and Preliminaries

We consider a market with n sellers, each selling a single good to a general population of consumers. We assume that the market operates in a round-based fashion. In each round t every seller i chooses a price p_i^t for her good. The supply w_i of seller i stays the same every round. No left-over supply from previous rounds is carried over (which is the case for example for perishable goods). Depending on the resulting price vector

$\mathbf{p}^t = (p_i^t)_i$, each seller observes a certain demand for her item given by $x_i(\mathbf{p}^t)$. These observed demands are governed by an underlying utility function of the consumers. To ensure that the problem is well defined we assume that the optimal revenue of any seller i for any profile $\mathbf{p}_{-i}$ of prices chosen by others is bounded in $[r, R]$. Intuitively, this is equivalent to saying that the set of allowed prices and supplies are such that revenue of any seller is not arbitrarily small or large.

We measure the performance of the pricing strategy used by the seller in terms of regret. Formally, the regret of an algorithm after T rounds is defined as the loss with respect to the single best action (here price) in hindsight. For example, if $\{r_i^t(p_i)\}_t$ denotes the sequence of revenue functions faced by the seller i then the regret with respect to the sequence of prices $\{p_i^t\}_{t=1}^T$ is defined as: $R_T = \sum_t r_i^t(p_i^*) - r_i^t(p_i^t)$ where $p_i^* = \operatorname{argmax}_p \sum_t r_i^t(p)$. Analogously, one can also define *dynamic regret* as the regret incurred with respect to a dynamic benchmark sequence. For example, if $p_1^*, p_2^* \cdots p_T^*$ is the sequence of prices against which we measure the loss of our algorithm, then dynamic regret is defined as:

$$R_T(p_1^*, p_2^* \cdots p_T^*) = \sum_t r_i^t(p_t^*) - r_i^t(p_i^t)$$

Log-Revenue Objective: Along the same lines as in previous chapter, we take an indirect approach to the problem of revenue optimization by optimizing the log-revenue objective instead of the actual revenue. For completeness, we define it again here:

$$\ln r_i(\mathbf{p}) = \ln \left[p_i \min \left\{ x_i(\mathbf{p}), w_i \right\} \right].$$

6.3 Regret Learning with CES Utilities

In this section, we demonstrate the kind of regret bounds that can be achieved in gross-substitutes CES markets (i.e., with the parameter $\rho \in (0, 1)$). We showed in the previous chapter (Section 5.2.1), that the log-revenue curve for CES utilities is concave with the gradient being a function of the price elasticity of demand. To ensure that the problem is well-defined we assume that the price elasticity of demand for any item i and any price vector $\mathbf{p}$ is bounded in $[E_{min}, E_{max}]$. Since the gradient of the log-revenue objective for any price p chosen by the seller is not known, direct application of the online gradient ascent[1] algorithm by Zinkevich [77] is not possible. Nevertheless, in what follows, we show that one can modify the algorithm and its analysis to recover the $O(\sqrt{T})$ regret bound. This modification of the algorithm only relies on the information whether the actual gradient is positive or negative. In the context of a seller, this simply corresponds

[1] See chapter 4 for an elaborate description.

to the sign of the excess demand. Before proceeding with the algorithm, we start with a claim for general convex functions with modified feedback.

Claim 6.1. *Consider a sequence of convex functions $f_1, f_2 \cdots f_T$ satisfying the following condition:*

$$g \;\leq\; |\nabla f_t(x)| \;\leq\; G \qquad \forall t \in [T], x \in \mathcal{X}.$$

Suppose for the action x_t chosen in round t and for $\gamma = \frac{G}{g}$, we receive as feedback $\nabla g_t(x_t) \in \left[\frac{\nabla f_t(x_t)}{\gamma}, \gamma \nabla f_t(x_t)\right]$, then the regret bound of OGD for step-size $\eta_t = 1/\sqrt{t}$ is given by $R_T \leq \gamma\sqrt{T}$.

Proof. The update rule of the OGD algorithm when the feedback $\nabla f_t(x_t)$ is available is given by: $\mathbf{x}_{t+1} = \Pi\left[\mathbf{x}_t - \eta_t \cdot \nabla f_t(\mathbf{x}_t)\right]$ where $\Pi(\cdot)$ is the euclidean projection operator. Since we use $\nabla g_t(x_t)$ instead, we would get a different sequence of decision points according to the update step as follows:

$$\mathbf{x}'_{t+1} = \Pi\left[\mathbf{x}'_t - \eta_t \cdot \nabla g_t(\mathbf{x}'_t)\right].$$

Since $\nabla g_t(\mathbf{x}'_t) \in \left[\frac{\nabla f_t}{\gamma}, \gamma \nabla f_t\right]$, we can re-write the same update step as:

$$\mathbf{x}'_{t+1} = \Pi\left[\mathbf{x}'_t - \eta'_t \cdot \nabla f_t(\mathbf{x}'_t)\right],$$

where $\eta'_t \in \left[\frac{\eta_t}{\gamma}, \gamma\eta_t\right]$ is such that $\eta_t \cdot \nabla g_t(\mathbf{x}_t) = \eta'_t \cdot \nabla f_t(\mathbf{x}_t)$. Therefore, we get the same sequence of steps by using ∇f_t but with a difference step size sequence. The claim follows from the same analysis as in Zinkevich [77] and replacing η_t by η'_t. □

This property allows us to use OGD even with *imperfect* gradient feedback, upto a multiplicative constant, to obtain regret bounds that are also within this same factor. Since the exact gradient in the case when $x_i(\mathbf{p}) < w_i$ is not available to the algorithm, we modify the feedback gradient based on the demand observed,

$$\frac{\partial \tilde{r}_i}{\partial \tilde{p}_i} = \begin{cases} 1 - E_i(p) & \Longrightarrow \; -1, \qquad \text{for } p_i : x_i(\mathbf{p}) < w_i \\ 1 & \Longrightarrow \; 1, \qquad \text{for } p_i : x_i(\mathbf{p}) \geq w_i \end{cases} \tag{6.1}$$

i.e., we work around this problem by choosing as feedback the gradient -1 whenever $x_i(\mathbf{p}) < w_i$ and $+1$ otherwise.

Theorem 6.2. *If any player i uses OGD on the log-revenue curve with $\eta_t = t^{-1/2}$ with the adjusted gradient feedback as in Equation* (6.1)*, then the cumulative loss in revenue*

of seller i is bounded by

$$\sum_t r_i^t(p_i^*) - r_i^t(p^t) \;=\; O\left(R \cdot \max\left\{E_{max} - 1, \frac{1}{E_{min} - 1}\right\} T^{1/2}\right),$$

where $p_i^* = \underset{p_i}{\operatorname{argmax}} \sum_t \tilde{r}_i(p_i, p_{-i}^t)$.

Proof. The price elasticity of demand for any item i at any price vector $\mathbf{p}$ satisfies $1 < E_{min} < |E_i(\mathbf{p})| < E_{max}$. Hence for the case when $x_i(\mathbf{p}) < w_i$, the gradient of log-revenue curve satisfies:

$$E_{min} - 1 \;\leq\; \left|\frac{\partial \tilde{r}_i}{\partial \tilde{p}_i}\right| \;\leq\; E_{max} - 1.$$

Using the same idea as in Claim 6.1, we can pretend to be using OGD on the actual log-revenue curve with a correspondingly modified step size $\eta' \in \left[(E_{max} - 1)\eta, \frac{\eta}{E_{min}-1}\right]$. The following bound then follows directly:

$$\sum_t \tilde{r}_i(p^*, p_{-i}^t) - \tilde{r}_i(p_i^t, p_{-i}^t) \;=\; O\left(\max\left\{E_{max} - 1, \frac{1}{E_{min} - 1}\right\} T^{1/2}\right),$$

where $p^* = \underset{p_i}{\operatorname{argmax}} \sum_t r_i(p_i, p_{-i}^t)$. The left-hand side of the above inequality can be further lower bounded:

$$\begin{aligned}
\sum_t \tilde{r}_i(p^*, p_{-i}^t) - \tilde{r}_i(p_i^t, p_{-i}^t) \;&=\; -\sum_t \ln\left(1 + \frac{r_i^t(p_i^t) - r_i^t(p^*)}{r_i^t(p^*)}\right) \\
&\geq\; \sum_t \frac{r_i^t(p_i^*) - r_i^t(p^t)}{r_i^t(p^*)} \\
&\geq\; \sum_t \frac{r_i^t(p_i^*) - r_i^t(p^t)}{R}.
\end{aligned}$$

□

This bound shows that for a broad class of learning dynamics, the regret incurred by any seller increases monotonically (as $O(\sqrt{T})$) with the number of rounds. Note that this is in stark contrast to our finding in the Chapter 5, where we showed that for a different class of price dynamics, the regret incurred with respect to the best-response prices (a stricter benchmark) by any seller in such a market is bounded by a constant. Although the tatonnement dynamics guarantee good bounds for static markets, it is arguably not the most natural price update dynamics. Furthermore, these guarantees require the market model to be static, which also limits its usefulness. A natural question, is then

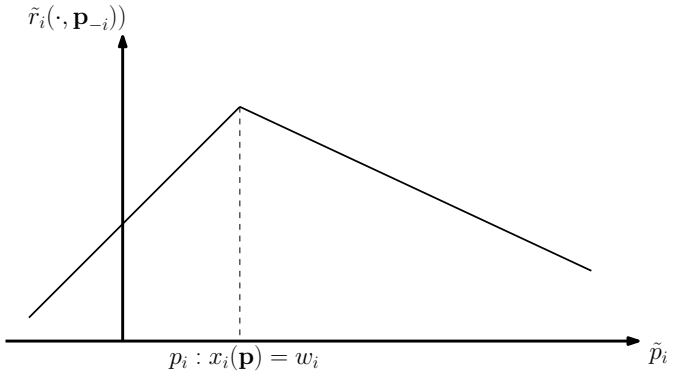


FIGURE 6.1: Log-revenue for `IGS` utilities

to ask: *Do natural price-update strategies exist that guarantee better regret bounds than the OGD algorithm?*

In the next section, we show that it is indeed possible to achieve a middle ground, i.e. where the sellers may choose their price update strategies from a sufficiently broad set of dynamics and at the same time achieve a better regret bound than the one we just showed.

6.4 Regret Learning with IGS Utilities

In this section, we consider the same market model as before (Section 6.2) except that the buyer utilities belong to the set of `IGS` utilities. For the definition of these utilities, we refer the reader to Section 4.2.3 in Chapter 4. Using the definition of `IGS` utility functions (Section 4.2.3) we can derive the following straightforward fact used directly in the rest of the chapter. The proposition follows from the definition of log-revenue function and the price elasticity of demand.

Proposition 6.3. *The gradient of the log-revenue function $\tilde{r}_i(\tilde{p}_i)$ satisfies:*

$$\frac{\partial \tilde{r}_i}{\partial \tilde{p}_i} = \begin{cases} 1 - E & \text{for } p_i : x_i(\mathbf{p}) < w_i \\ 1 & \text{for } p_i : x_i(\mathbf{p}) \geq w_i \end{cases}$$

This proposition implies that the log-revenue function for seller i, keeping prices of all other items fixed, takes a rather simple form as seen in figure 6.1.

6.4.1 Game Theoretic Interpretation

We start our investigation into this problem by observing that the revenue optimization problem in a market (as defined in Section 6.2) is equivalent to agents in a game using

learning algorithms locally to optimize their utility, where this utility is a function of the *strategies* of all agents in the game. Problems of this flavor have already been studied in different game-theoretic settings but are not applicable in a black-box fashion to our problem on account of the market specific constraints. Specifically, the log-revenue objective although concave is not smooth, an assumption used in almost all gradient-based learning algorithms. This calls for a different approach than the ones taken in the idealized settings.

With this context in mind, we start from the result of [2], where it is proved that if all players in a game use learning algorithms satisfying a certain technical property, called the RVU property (see Definition 6.4), then the regret incurred by each individual agent is $O(T^{1/4})$. A natural question is then: Can we use the same technique in our revenue optimization problem in markets?

Definition 6.4 (RVU property, [2]). *We say that a vanishing regret algorithm satisfies the Regret bounded by Variation in Utilities (RVU) property with parameters $\alpha > 0$ and $0 < \beta \leq \gamma$ and a pair of dual norms $(\|\cdot\|, \|\cdot\|_*)$ if its regret on any sequence of utilities $\mathbf{u}^1, \mathbf{u}^2, \dots \mathbf{u}^T$ is bounded by:*

$$\sum_{t=1}^{T} \left\langle \mathbf{p}^* - \mathbf{p}^t \,\middle|\, \mathbf{u}^t \right\rangle \;\leq\; \alpha \;+\; \beta \sum_{t=1}^{T} \left\| \mathbf{u}^t - \mathbf{u}^{t-1} \right\|_* \;-\; \gamma \sum_{t=1}^{T} \left\| \mathbf{p}^t - \mathbf{p}^{t-1} \right\|.$$

Although this property is defined for linear utility functions, we can extend this definition to concave utilities by using the gradient of the utility with respect to p_i as proxy for $\mathbf{u}^t$. In the context of our problem

$$\tilde{r}_i^t(p_i^*) - \tilde{r}_i^t(p_i^t) \;\leq\; \left\langle \mathbf{p}^* - \mathbf{p}^t \,\middle|\, \frac{\partial \tilde{r}_i}{\partial \tilde{p}_i} \right\rangle.$$

As noted in [2], the standard online learning algorithms such as Online Mirror Descent (generalization of OGD) and Follow-the-Regularized-Leader (FTRL) do not satisfy the RVU property. However, Rakhlin and Sridharan [78] and Syrgkanis et al. [2] have developed modified versions of these algorithms, namely Optimistic Mirror Descent (OMD) and Optimistic FTRL (OFTRL) respectively, that do satisfy this property,

Proposition 6.5 (Informal, [2]). *Let D denote a measure of the diameter of the decision space. Then:*

1. *The OMD algorithm using step size η satisfies the RVU property with constants $\alpha = D/\eta$, $\beta = \eta$ and $\gamma = 1/(8\eta)$*
2. *The OFTRL algorithm using step size η satisfies the RVU property with constants $\alpha = D/\eta$, $\beta = \eta$ and $\gamma = 1/(4\eta)$*

In the context of continuous games, the utility function (alternatively, the objective) of each player should additionally satisfy some *regularity* conditions. For ease of presentation, we shall refer to the player objectives satisfying these conditions as *regular objectives* and are defined, in a general sense, as follows:

Definition 6.6 (Regular Objective). *Let the strategy space of each player i be denoted by $S_i \in \mathbb{R}^d$ and the combined strategy space by $\mathcal{S} = S_1 \times S_2 \times \cdots S_n$. Let $\mathbf{w} = (\mathbf{s}_i)_{i=1}^n$ denote the combined strategy profile where the strategy of each player $s_i \in S_i$. An objective function $f_i : \mathcal{S} \to \mathbb{R}$ of a player i is said to be regular if it satisfies the following conditions:*

1. *(Concave in player strategy) For each player i and for each profile of opponent strategies $\mathbf{s}_{-i}$, the function $f_i(\cdot, \mathbf{s}_{-i})$ is concave in $\mathbf{s}_i$.*
2. *(Lipschitz Gradient) For each player i, the gradient of the objective with respect to i, $\delta_i(\mathbf{s}) = \nabla_i f_i(\mathbf{s})$ is L-Lipschitz continuous with respect to the L1-norm. i.e.*

$$\left\|\delta_i(\mathbf{s}) - \delta_i(\mathbf{s}')\right\|_* \;\leq\; L \cdot \left\|\mathbf{s} - \mathbf{s}'\right\|.$$

6.4.2 Smoothed Log-Revenue Curve

One of the foremost requirements to apply the analysis based on the `RVU` property is that the utility function should be smooth, specifically, the gradient of the objective should be L-Lipschitz continuous.[2] Clearly, as seen in Figure 6.1, this is not the case with our log-revenue objective. We work around this problem by using a *smoothed* gradient feedback.

Definition 6.7 (Smoothed Gradient Feedback). *For any fixed seller i and price vector $\mathbf{p}_{-i}$, we define the smoothed gradient for player i, $\delta_{i,\mathsf{X}_i}(\cdot)$, as follows:*

$$\delta_{i,\mathsf{X}_i}(p_i) = \begin{cases} 1, & \text{for } p_i : x_i(\mathbf{p}) > w_i \\ 1 - E, & \text{for } p_i : x_i(\mathbf{p}) < \mathsf{X}_i \\ 1 + \frac{E(\tilde{x}_i(\mathbf{p}) - \tilde{w}_i)}{\tilde{w}_i - \tilde{\mathsf{X}}_i}, & \text{otherwise} \end{cases}$$

where X_i is a threshold parameter for seller i.

For ease of notation, we shall denote $\delta_{i,\mathsf{X}_i}(p_i)$ by simply δ_i when clear from context. For purposes of analysis, we parametrize the threshold parameter of seller i as $\mathsf{X}_i = \frac{w_i}{\exp(\epsilon r)}$

[2] Informally, this is required to ensure that small changes in prices do not lead to large changes in utility gradient.

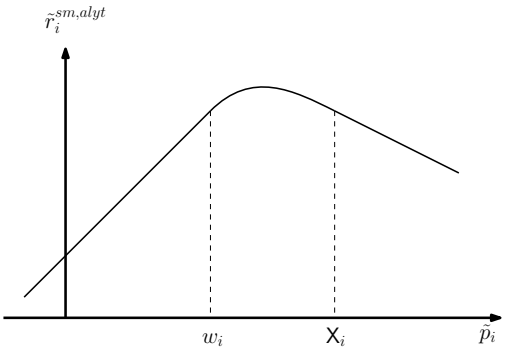


FIGURE 6.2: Smoothed log-revenue from an analytical standpoint

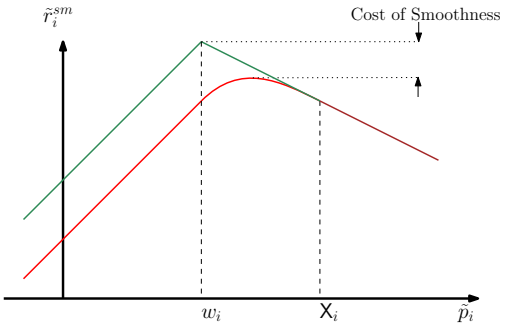


FIGURE 6.3: Smoothed vs actual log-revenue curve

where ϵ is a small constant and r is a lower bound on optimal revenue of seller i. Also, henceforth we shall refer to the *actual* revenue curve by $\tilde{r}(\cdot)$ and the algorithm's view of smoothed revenue curve by $\tilde{r}^{sm}(\cdot)$.

Lemma 6.8. *The smoothed revenue objective, $\tilde{r}_i^{sm}(\mathbf{p})$, for any seller i is regular.*

The proof of this lemma can be found in Section 6.7.1.

6.4.3 Cost of Smoothness

Since our learning algorithm only uses the smoothed gradient feedback the resulting regret bound also holds only for the smoothed view of the log-revenue curve. Therefore, the optimal price in this smoothed view would be the price for which the smoothed gradient is zero, although this price is clearly suboptimal for the actual revenue curve. (See Fig 6.3). To prove bounds with respect to the actual revenue curve, we need to draw connections between the smoothed and actual revenue for any fixed price.

Lemma 6.9. *For any seller i and fixed $\mathbf{p}_{-i}$ and for any fixed price p chosen by seller i:*

$$0 \;\leq\; \tilde{r}_i(p, \mathbf{p}_{-i}) \;-\; \tilde{r}_i^{sm}(p, \mathbf{p}_{-i}) \;\leq\; \epsilon r$$

Theorem 6.10. *Suppose each seller i uses the OFTRL algorithm on the log-revenue objective using the smoothed gradient feedback and threshold demand $\mathsf{X}_i = \frac{w_i}{\exp(\epsilon r)}$. Let $p_i^{**} = \operatorname{argmax}_p \sum_t \tilde{r}_i^t(p)$ denote the optimal price in hindsight with respect to the log-revenue objective. Then the actual loss in revenue is bounded by:*

$$\sum_{t=1}^{T} (1 - \epsilon R)\, r_i^t(p_i^{**}) - r_i^t(p^t) \;=\; O\left(\left(\frac{R^2 E^2}{\epsilon r}\right)^{1/2} T^{1/4}\right) \;-\; \epsilon R T.$$

Proof. Since $\tilde{r}_i^{sm}(p_i, \mathbf{p}_{-i})$ satisfies the regularity condition (Definition 6.6), if each seller uses a learning algorithm satisfying the RVU property, then the individual regret satisfies:

$$\begin{aligned}\sum_t \tilde{r}_i^{sm}(p_i^{**}, \mathbf{p}_{-i}^t) \;-\; \tilde{r}_i^{sm}(p_i^t, \mathbf{p}_{-i}^t) \;&\leq\; \sum_t \tilde{r}_i^{sm}(\bar{p}_i^*, \mathbf{p}_{-i}^t) \;-\; \tilde{r}_i^{sm}(p_i^t, \mathbf{p}_{-i}^t) \\ &\leq\; \sum_t \left\langle \delta_{i,\mathsf{X}_i}(\mathbf{p}^t) \,\middle|\, \tilde{\bar{p}}_i^* - \tilde{p}_i^t \right\rangle,\end{aligned}$$

where $\bar{p}_i^* = \operatorname*{argmax}_p \sum_t \tilde{r}_i^{sm}(p, \mathbf{p}_{-i}^t)$. For ease of notation, we denote $\delta_{i,\mathsf{X}_i}(\mathbf{p}^t)$ by δ_i^t. Using Lemma 6.9 to lower bound the left-hand-side above:

$$\begin{aligned}\sum_t \tilde{r}_i^{sm}(p^{**}, \mathbf{p}_{-i}^t) \;-\; \tilde{r}_i^{sm}(p_i^t, \mathbf{p}_{-i}^t) \;&\geq\; \sum_t \left(\tilde{r}_i^t(p_i^{**}) \;-\; \epsilon r\right) \;-\; \tilde{r}_i(p_i^t) \\ &\geq\; \sum_t (1-\epsilon)\, \tilde{r}_i(p_i^{**}) \;-\; \tilde{r}_i(p_i^t).\end{aligned} \tag{6.2}$$

The last inequality holds since r is a lower bound on revenue. We still have to prove an upper bound on the expression $\sum_t \left\langle \delta_i^t \,\middle|\, \tilde{\bar{p}}_i^* - \tilde{p}_i^t \right\rangle$. Since our learning algorithm satisfies the RVU property, by Definition 6.4 it follows that:

$$R_T \leq \alpha \;+\; \beta \sum_{t=1}^{T} \left|\delta_i^t - \delta_i^{t-1}\right|^2.$$

Since the smoothed gradient $\delta_i(\mathbf{p})$ for any seller is L-Lipschitz continuous (Lemma 6.19), for $L = \frac{E^2}{\epsilon r}$ we can bound $\left|\delta_i^t - \delta_i^{t-1}\right|^2$ by

$$\begin{aligned}\left|\delta_i^t - \delta_i^{t-1}\right|^2 \;&\leq\; L^2 \left(\sum_j \left|p_j^t - p_j^{t-1}\right|\right)^2 \\ &\leq\; L^2 n \sum_j \left|p_j^t - p_j^{t-1}\right|^2.\end{aligned}$$

In addition to the fact that OFTRL satisfies the RVU property, it is also known that the algorithm satisfies a *stability* property (Lemma 20, [2]), i.e., $\left|p_j^t - p_j^{t-1}\right| \leq 2\eta$ where η is the step-size parameter of the algorithm.

We can now bound the regret by: $R_T \leq \alpha \;+\; 4n^2\beta L^2\eta^2 T$. Finally, substituting the RVU parameters of the algorithm (Proposition 7, [2]) $\alpha = D/\eta$, $\beta = \eta$ and $\gamma = 1/4\eta$ with $\eta = (Ln)^{-1/2}T^{-1/4}$ we get:

$$R_T \leq D/\eta + 4\eta^3 L^2 n^2 T = O(\sqrt{Ln}(D+4)T^{1/4}).$$

Combining this with Equation 6.2 and substituting the value of L we get:

$$\sum_{t=1}^{T} (1-\epsilon)\, \tilde{r}_i^t(p_i^{**}) - \tilde{r}_i^t(p_i^t) \;\leq\; O\left(\left(\frac{E^2}{\epsilon r}\right)^{1/2} \cdot T^{1/4}\right).$$

Rearranging the inequality and using same steps as in the proof of Lemma 6.2:

$$\begin{aligned}
\sum_t \frac{r_i^t(p_i^{**}) - r_i^t(p^t)}{r_i^t(p^{**})} &\leq O\left(\left(\frac{E^2}{\epsilon r}\right)^{1/2} \cdot T^{1/4}\right) + \epsilon \sum_{t=1}^{T} \tilde{r}_i^t(p_i^{**}) \\
\sum_t r_i^t(p_i^{**}) - r_i^t(p^t) &\leq O\left(\left(\frac{E^2 R^2}{\epsilon r}\right)^{1/2} \cdot T^{1/4}\right) + \epsilon R \sum_{t=1}^{T} \tilde{r}_i^t(p_i^{**}) \\
&\leq O\left(\left(\frac{R^2 E^2}{\epsilon r}\right)^{1/2} T^{1/4}\right) + R\epsilon \sum_{t=1}^{T} (r_i^t(p_i^{**}) - 1)
\end{aligned}$$

$$\sum_t (1-\epsilon R)\, r_i^t(p_i^{**}) - r_i^t(p^t) \;\leq O\left(\left(\frac{R^2 E^2}{\epsilon r}\right)^{1/2} T^{1/4}\right) - \epsilon R T$$

□

Similar bounds can be shown in the case when sellers use the Optimistic Mirror Descent (OMD) algorithm.

Remark 6.11. Here we compare the total revenue obtained to the total revenue with respect to the fixed price $p^{**} = \operatorname{argmax}_p \sum_t \tilde{r}_i^t(p)$ i.e. the price in hindsight that optimizes the cumulative log-revenue objective and not necessarily the revenue objective itself. We note that since the revenue function need not be concave, it is not immediately clear how to characterize the resulting cumulative revenue function and the price optimizing it. For this reason, we are using the price that optimizes the cumulative log-revenue.

6.5 Learning with a Dynamic Benchmark

A bound on the loss of revenue of a seller with respect to the single price p_i^{**} in hindsight is a comparatively weak benchmark. In order to prove a regret bound with respect to a stronger benchmark, we shall focus on a more constrained sequence of benchmark prices. In what follows, we define a class of learning algorithms whose guarantees apply to *any* game setting where strategic players use regret minimization to maximize their own utility. For generality, we define this class for any sequence of concave utility functions

$\{u_i^t(\cdot)\}_t$. In the following section, we shall specialize this guarantee to the context of revenue optimization in markets.

Definition 6.12 (DRVU property). *We say that a vanishing regret algorithm satisfies the Dynamic Regret bounded by Variation in Utilities (DRVU) property with parameters $\alpha, \rho > 0$ and $0 < \beta \leq \gamma$ and a pair of dual norms $(\|\cdot\|, \|\cdot\|_*)$ if its regret on any sequence of utilities $\mathbf{u}^1, \mathbf{u}^2, \dots \mathbf{u}^T$ with respect to the benchmark sequence $\{p_i^{*,t}\}_t$ is bounded by:*

$$\sum_{t=1}^{T} \left\langle \mathbf{p}^{*,t} - \mathbf{p}^t \,\middle|\, \mathbf{u}^t \right\rangle \;\leq\; \alpha \;+\; \beta \sum_{t=1}^{T} \left\|\mathbf{u}^t - \mathbf{u}^{t-1}\right\|_*^2 \;+\; \rho \sum_{t=1}^{T} \left\|\mathbf{p}^{*,t} - \mathbf{p}^{*,t-1}\right\| \;-\; \gamma \sum_{t=1}^{T} \left\|\mathbf{p}^t - \mathbf{p}^{t-1}\right\|.$$

This definition is an extension of the `RVU` property. The difference is in the term $\rho \sum_t \left\|\mathbf{p}^{*,t} - \mathbf{p}^{*,t-1}\right\|$ that quantifies the hardness of learning with respect to a dynamic strategy. As for the `RVU` property, this property is defined with respect to linear utilities and can be extended to concave utilities by standard arguments.

Theorem 6.13 (Informal)**.** *The OMD algorithm, with step size η and suitably chosen parameters, satisfies the DRVU property with constants $\alpha = D_1/\eta$, $\rho = D_2/\eta$ $\beta = \eta$ and $\gamma = 1/(8\eta)$ for constants D_1 and D_2.*

For purposes of readability we defer the proof of this theorem to Section 6.7.3. This section also contains a more detailed discussion on the optimistic mirror descent algorithm (OMD), as used in the lemma. Using this new definition we can now extend almost all of the results in [2] to corresponding results for dynamic regret. We state the following claim for concreteness.

Corollary 6.14. *Let $C_T = \sum_t \| p_i^{*,t} - p_i^{*,t-1} \|$ denote the cumulative change in benchmark strategies of player i. If all players use algorithms satisfying the DRVU property, then the regret incurred by any player i satisfies:*

$$\sum_t u_i^t \left(p_i^t, p_{-i}^t\right) - \; u_i^t \left(p_i^{*,t}, p_{-i}^t\right) \;=\; O\left((1 + C_T) T^{1/4}\right)$$

6.5.1 Revenue Optimization in Dynamic Markets

Dynamic Market Model: We define a dynamic market $\mathcal{M} = (M_1, M_2 \cdots M_T)$, as a sequence of markets with the same set of sellers and buyers, with the same IGS utility

functions as in Definition 4.2 but with a dynamic supply vector i.e. we characterize the dynamicity of the market by the sequence of supply vectors $\mathbf{w}_1, \mathbf{w}_2 \cdots \mathbf{w}_T$. In order to achieve a strong dynamic regret bound, we shall assume that the income elasticity parameter of the market is equal to one. This is a standard assumption in many market models and is also satisfied by CES utilities.

In this section, we connect the dynamic regret of any seller i to the inherent instability of the market by choosing the sequence of *equilibrium prices* for seller i at each round as the benchmark sequence, i.e. $\{p_i^{eq,t}\}_{t=1}^T$. Since the supply vector may change every round, the equilibrium prices may also correspondingly change. These changes in equilibrium prices completely capture the inherent instability of the market. For example, if the supply stays the same every round, then this benchmark is the same as choosing the equilibrium price in each round. On the other hand, if the supply fluctuates wildly from one round to the next, then so do the equilibrium prices and there is no hope of achieving a sub-linear regret bound. That is, the resulting dynamic regret bound captures the inherent market instability. In what follows, we derive a relationship connecting the changes in equilibrium prices, which is also our benchmark, to the changes in supply observed. The main theorem bounding the loss in revenue can then be derived using a similar approach as in Theorem 6.10.

Lemma 6.15. *For some gross-substitutes market let* $\mathbf{p}^{old}$, $\mathbf{x}^{old}$ *and* $\mathbf{p}^{new}$, $\mathbf{x}^{new}$ *denote the price and the resulting demand vectors, respectively.*

(a) If $\mathbf{p}^{new} = \mathbf{p}^{old}(1+\epsilon)$ *then,* $\mathbf{x}^{new} = \frac{\mathbf{x}^{old}}{1+\epsilon}$.

(b) If $\mathbf{p}^{new} = \frac{\mathbf{p}^{old}}{(1+\epsilon)}$ *then,* $\mathbf{x}^{new} = \mathbf{x}^{old}(1+\epsilon)$.

Proof. We only prove Part (a) here. Part (b) follows from identical steps. Note that increasing the prices of all items by a factor of $(1+\epsilon)$ is equivalent to decreasing the income of all buyers by the same factor. Let the income of player i be denoted by I_i. Then for any buyer i, $I_i^{new} = \frac{I_i^{old}}{(1+\epsilon)}$. Further, for gross-substitutes markets with CES utilities, it is known that the income elasticity parameter, ϵ_I, for any player is exactly equal to 1. By definition of income elasticity:

$$\epsilon_I = \frac{\frac{\mathbf{x}^{new}-\mathbf{x}^{old}}{\mathbf{x}^{old}}}{\frac{I^{new}-I^{old}}{I^{old}}} = \frac{\frac{\mathbf{x}^{new}-\mathbf{x}^{old}}{\mathbf{x}^{old}}}{\frac{-\epsilon}{1+\epsilon}} = 1$$

Rearranging, $\mathbf{x}^{new} = \frac{\mathbf{x}^{old}}{1+\epsilon}$. □

Lemma 6.16. *Suppose the supply vector changes from* $\mathbf{w}_{old} = (w_i)_i$ *to* $\mathbf{w}_{new} = (w_i', w_{-i})$. *Let* $\mathbf{p}^{eq,old}$ *and* $\mathbf{p}^{eq,new}$ *be the equilibrium price vectors corresponding to the old and new supply vectors, respectively.*

(a) If $w_i' = w_i \cdot (1+\epsilon)$ *then,*

$$1 \;\leq\; \max_j \frac{p_j^{eq,old}}{p_j^{eq,new}} \;\leq\; (1+\epsilon)$$

(b) If $w_i' = \frac{w_i}{1+\epsilon}$ *then,*

$$1 \;\geq\; \min_j \frac{p_j^{eq,old}}{p_j^{eq,new}} \;\geq\; \frac{1}{1+\epsilon}$$

Proof. Consider the case where $w_i' = w_i \cdot (1+\epsilon)$. To prove a contradiction, assume that for some player j, $\frac{p_j^{eq,old}}{p_j^{eq,new}} = z \;>\; (1+\epsilon)$. By equilibrium condition,

$$\begin{aligned} w_j^{new} &= x_j\left(p_j^{eq,new}, p_{-j}^{eq,new}\right) \\ &= x_j\left(\frac{p_j^{eq,old}}{z}, p_{-j}^{eq,new}\right) \\ &\overset{(a)}{\geq} x_j\left(\frac{p_j^{eq,old}}{z}, \frac{p_{-j}^{eq,old}}{z}\right) \\ &\overset{(b)}{=} z \cdot x_j\left(\mathbf{p}^{eq,old}\right) = z \cdot w_j^{old}. \end{aligned} \tag{6.3}$$

The inequality (a) follows from the definition of gross substitutes markets. Equality (b) is the direct application of Lemma 6.15. Since this is a contradiction, we conclude that $\max_j \frac{p_j^{eq,old}}{p_j^{eq,new}} \;\leq\; (1+\epsilon)$.

For the lower bound suppose that for some item j, $\frac{p_j^{eq,old}}{p_j^{eq,new}} = z_2 \;<\; 1$. Then,

$$\begin{aligned} w_j^{new} = \; x_j(\mathbf{p}^{eq,new}) &= x_j\left(\frac{p_j^{eq,old}}{z_2}, p_{-j}^{eq,new}\right) \\ &\overset{(a)}{\leq} x_j\left(\frac{p_j^{eq,old}}{z_2}, \frac{p_{-j}^{eq,old}}{z_2}\right) \\ &\overset{(b)}{=} x_j(\mathbf{p}^{eq,old}) \cdot z_2 = \; w_j^{old} \cdot z_2, \end{aligned}$$

which is a contradiction for any $z_2 < 1$. The inequalities (a) and (b) follow the same reasoning as in Inequality (6.3). This implies that for an increase in supply of item i, the price of no item j increases and the maximum decrease in the price of any item j is at most a factor of $(1+\epsilon)$. By analogous arguments, we can prove the result for the case when the supply decreases. □

Corollary 6.17. *Let $\|\cdot\|_1$ denote the 1-norm. If the supply vector changes from $\mathbf{w}^t$ to $\mathbf{w}^{t+1}$, where the supply of each item may change independently, then:*

$$\max_j |\tilde{p}_j^{eq,t+1} - \tilde{p}_j^{eq,t}| \;\leq\; \left\|\tilde{\mathbf{w}}^{t+1} - \tilde{\mathbf{w}}^t\right\|_1$$

Proof. First note that we can re-write the result of Lemma 6.16 in log scale as:

$$\max_j |\tilde{p}_j^{eq,t+1} - \tilde{p}_j^{eq,t}| \;\leq\; |\tilde{w}_i^{t+1} - \tilde{w}_i^t|,$$

where we assumed that the supply of only item i changed. Now, for any two supply vectors $\mathbf{w}^{t+1}$ and $\mathbf{w}^t$, consider the switch from $\mathbf{w}^t$ to $\mathbf{w}^{t+1}$ sequentially in a pre-defined order while keeping the supplies of remaining sellers fixed during this switch. From Lemma 6.16, we know that for each such intermediate step, where the supply of only item j changes, the maximum change in equilibrium is at most $|\tilde{w}_j^{t+1} - \tilde{w}_j^t|$. The cumulative change in equilibrium can then simply be upper bounded by the sum of these individual changes. □

Theorem 6.18. *Let $W_T = \sum_t \left\|\tilde{\mathbf{w}}^t - \tilde{\mathbf{w}}^{t-1}\right\|_1$ denote the cumulative change in the market in terms of changes in supplies. Suppose each seller i uses the OMD algorithm on the log-revenue function with smoothed gradient feedback and threshold demand $\mathsf{X}_i^t = \frac{w_i^t}{\exp(\epsilon r)}$. Let $\{p_i^{eq,t}\}_t$ denote the sequence of equilibrium prices for seller i. Then:*

$$\sum_{t=1}^{T} (1 - \epsilon R)\, r_i^t(p_i^{eq,t}) - r_i^t(p^t) \;\leq\; O\left(\left(\frac{R^2E^2}{\epsilon r}\right)^{1/2} \cdot (1 + W_T)T^{1/4}\right)$$

Proof. This bound can be derived using almost the same steps as in Theorem 6.10 and using Corollary 6.17 to account for the cumulative change in benchmark prices. □

6.6 Experimental Evaluation

We analyze the performance of our modified OGD and Optimistic Mirror Descent (OMD) algorithms in the case where the consumer utility functions satisfy the CES property. We are able to do this since by choosing the utility parameters appropriately, the CES utilities approximately satisfies the definition of IGS. In our simulations, we show that the OMD algorithm indeed performs as proved in our analysis.

We consider the scenario with 2 items and the value of $E = 2.5$. We assume that the market is static in that each seller has a supply of one unit every round and uses the threshold parameter $\mathsf{X}_i = 0.9$. We observe that the modified OGD algorithm converges quickly to the neighbourhood of the optimal price but then keeps oscillating around

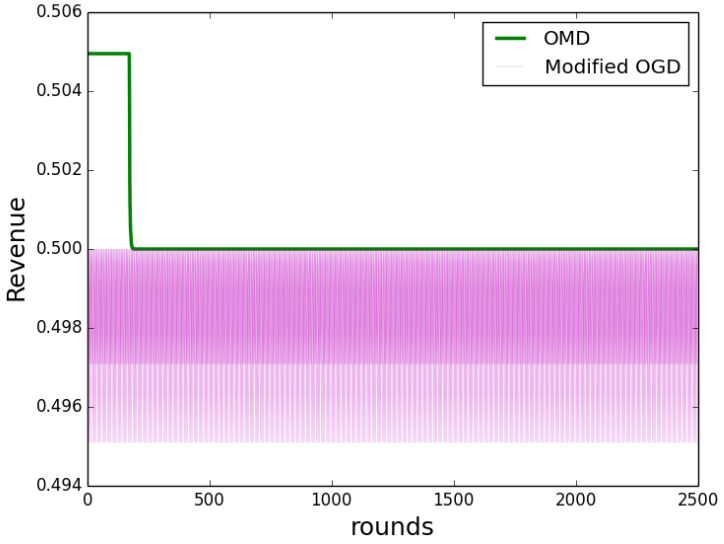


FIGURE 6.4: Modified OGD vs OMD

it. This is expected since in this neighbourhood the observed gradients might change abruptly. The OMD algorithm on the other hand takes a while before it comes close to the neighbourhood but once there converges to optimum quickly. As described in the analysis, this is precisely the reason for using the smoothed gradient feedback.

6.7 Omitted Proofs

6.7.1 Proof of Lemma 6.8

The proof depends on the Lipschitz continuity property proved in the following lemma.

Lemma 6.19. *For any seller i, the gradient of the smoothed revenue curve with threshold demand $\mathsf{X}_i = \frac{w_i}{\exp(\epsilon r)}$ satisfies $\frac{E^2}{\epsilon r}$-Lipschitz continuity, i.e.,*

$$\left\|\delta_i(\mathbf{p}^1) - \delta_i(\mathbf{p}^2)\right\|_* \leq \frac{E^2}{\epsilon r} \cdot \left\|\tilde{p}^1 - \tilde{p}^2\right\|. \tag{6.4}$$

Proof. It is known that (Lemma 24, [2]) if for all j,

$$\left\|\delta_i(\mathbf{p}^1) - \delta_i(p_j^2, \mathbf{p}_{-j}^1)\right\|_* \leq \frac{E^2}{\epsilon r} \cdot \left\|\tilde{p}_j^1 - \tilde{p}_j^2\right\|$$

then $\delta_i(\cdot)$ satisfies Inequality (6.4). We shall first prove the case when j is equal to i.

This is equivalent to proving $\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_i} \leq \frac{E^2}{\epsilon r}$ since the revenue curve is differentiable. By observation, we note that the maximum change in smoothed gradient $\delta_i(\mathbf{p})$ occurs for

prices when $\mathsf{X}_i \leq x_i(\mathbf{p}) \leq w_i$. This implies:

$$\begin{aligned}\left|\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_i}\right| &\leq \left|\frac{\partial}{\partial \tilde{p}_i}\left(1 + \frac{E(\tilde{x}_i(\mathbf{p}) - \tilde{w}_i)}{\tilde{w}_i - \tilde{\mathsf{X}}_i}\right)\right| \\ &= \left|\frac{E}{\tilde{w}_i - \tilde{\mathsf{X}}_i} \cdot \frac{\partial}{\partial \tilde{p}_i}\tilde{x}_i(\mathbf{p})\right| \\ &= \left|\frac{E}{\epsilon r} \cdot -E_i(\mathbf{p})\right| \\ \left|\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_i}\right| &\leq \frac{E^2}{\epsilon r}.\end{aligned}$$

In a similar way, we can show that the smoothed gradient of seller i is Lipschitz continuous also with respect to the price of any other seller j, i.e., $\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_j} \leq \frac{E^2}{\epsilon r}$. Using the same arguments as above, we get:

$$\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_j} \leq \frac{E}{\tilde{w}_i - \tilde{\mathsf{X}}_i} \cdot \left|\frac{\partial}{\partial \tilde{p}_j}\tilde{x}_i(\mathbf{p})\right|.$$

The cross derivative term $\frac{\partial \tilde{x}_i(\mathbf{p})}{\partial \tilde{p}_j}$ is exactly the *cross-price elasticity* of item i with respect to item j. We denote it by $E_{ij}(\mathbf{p})$. By definition of `IGS` utility functions this is exactly E. Therefore,

$$\left|\frac{\partial \delta_i(\mathbf{p}^1)}{\partial \tilde{p}_j}\right| \leq \frac{E^2}{\epsilon r}.$$

□

Lemma 6.8. *The smoothed revenue objective, $\tilde{r}_i^{sm}(\mathbf{p})$, for any seller i is regular.*

The lemma now follows directly from Lemma 6.19 the fact that $\tilde{r}_i^{sm}(\mathbf{p})$ is concave in $\tilde{p}_i$.

6.7.2 Proof of Lemma 6.9

To prove Lemma 6.9, we first quantify the difference in optimal revenues with respect to the log-revenue and its smoothed version.

Lemma 6.20. *For a fixed $\mathbf{p}_{-i}$ let p_{i,X_i} denote the price such that $x_i(p_{i,\mathsf{X}_i}, \mathbf{p}_{-i}) = \mathsf{X}_i$, where $\mathsf{X}_i = \frac{w_i}{\exp(\epsilon r)}$ is the threshold demand of seller i. Then:*

$$\tilde{r}_i(p_i^*, \mathbf{p}_{-i}) \; - \; \tilde{r}_i(p_{i,\mathsf{X}_i}, \mathbf{p}_{-i}) \; = \; \frac{E-1}{E} \cdot \epsilon r$$

Proof. The lemma follows directly from the following two observations:

1. For any price $p_i > p_i^*$, where p_i^* is the revenue maximizing price of seller i, chosen by seller,

$$\tilde{r}_i(p_i^*) - \tilde{r}_i(p_i) \;=\; (E-1)(\tilde{p}_i - \tilde{p}_i^*).$$

   This follows from our assumption that the gradient of log-revenue curve for any price $p_i > p_i^*$ is a constant equal to $-(E-1)$.

2. For $\tilde{p}_{i,\mathsf{X}_i}$ and $\tilde{p}_i^*$ as defined above, the following holds:

$$\tilde{p}_{i,\mathsf{X}_i} - \tilde{p}_i^* \;=\; \frac{\epsilon r}{E}.$$

   This can be shown by the following sequence of utilities.

$$\begin{aligned}
\tilde{r}_i(p_i^*) &= \tilde{r}_i(p_{i,\mathsf{X}_i}) + (E-1)(\tilde{p}_{i,\mathsf{X}_i} - \tilde{p}_i^*) \\
\tilde{p}_i^* + \tilde{w}_i &= \tilde{p}_{i,\mathsf{X}_i} + \tilde{x}_i(p_{i,\mathsf{X}_i}) + (E-1)(\tilde{p}_{i,\mathsf{X}_i} - \tilde{p}_i^*) \\
\tilde{w}_i - \tilde{x}_i(p_{i,\mathsf{X}_i}) &= E(\tilde{p}_{i,\mathsf{X}_i} - \tilde{p}_i^*)
\end{aligned}$$

   Since $\tilde{x}_i(p_{i,\mathsf{X}_i}) = \tilde{\mathsf{X}}_i$, it follows that

$$\tilde{w}_i - \tilde{x}_i(p_{i,\mathsf{X}_i}) \;=\; \ln\left(\frac{w_i}{x_i(p_{i,\mathsf{X}_i})}\right) \;=\; \epsilon r.$$

   The lemma follows from this.

□

We are now ready to bound the difference between the actual revenue and the smoothed revenue for any seller i and price p_i.

Lemma 6.9. *For any seller i and fixed $\mathbf{p}_{-i}$ and for any fixed price p chosen by seller i:*

$$0 \;\leq\; \tilde{r}_i(p, \mathbf{p}_{-i}) \;-\; \tilde{r}_i^{sm}(p, \mathbf{p}_{-i}) \;\leq\; \epsilon r$$

Proof. The left hand-side of the inequality follows directly from our construction of smoothed gradient. For the right-hand side we observe that the difference between the revenue values of the two curves is maximum at p_i^*. Hence, in the following, we shall focus on bounding $\tilde{r}_i(p_i^*) \;-\; \tilde{r}_i^{sm}(p_i^*)$. Note that the gradient of the smoothed revenue function changes gradually from $-(E-1)$ to 1 in the price range p_{i,X_i} to p_i^* and in the worse case, might change abruptly, i.e.

$$\begin{aligned}
\tilde{r}_i^{sm}(p^*) &\geq \tilde{r}_i^{sm}(p_{i,\mathsf{X}_i}) \;-\; (\tilde{p}_{i,\mathsf{X}_i} - \tilde{p}_i^*) \\
&\geq \tilde{r}_i^{sm}(p_{i,\mathsf{X}_i}) \;-\; \frac{\epsilon r}{E}
\end{aligned}$$

Using Lemma 6.20 and using the fact that $\tilde{r}_i^{sm}(p_{i,\mathsf{X}_i}) = \tilde{r}_i(p_{i,\mathsf{X}_i})$:

$$\begin{aligned} \tilde{r}_i(p_i^*) \;-\; \tilde{r}_i(p_{i,\mathsf{X}_i}) \;&=\; \frac{E-1}{E}\cdot \epsilon r \\ \tilde{r}_i(p_i^*) \;-\; \left(\tilde{r}_i^{sm}(p_i^*) + \frac{\epsilon r}{E}\right) \;&\leq\; \frac{E-1}{E}\cdot \epsilon r \\ \tilde{r}_i(p_i^*) \;-\; \tilde{r}_i^{sm}(p_i^*) \;&\leq \epsilon r \end{aligned}$$

□

6.7.3 Optimistic Mirror Descent and the DRVU Property

Optimistic Mirror Descent (OMD): Consider the following online convex optimization problem: Let $\mathcal{F}$ be the convex set of actions of the learner. In each round t, the learner chooses an action $\mathbf{x}_t$ and observes a linear utility function $\mathbf{u}^t$[3]. The goal of the agent is to maximize her utility, i.e. $\sum_t \langle \mathbf{x}_t \,|\, \mathbf{u}^t \rangle$. Let $\mathcal{R}$ be a 1-strongly convex function with respect to some norm $\|\cdot\|$ on $\mathcal{F}$. Suppose the agent has a prediction, M_t, about the forthcoming utility vector in round t. The OMD algorithm incorporates this information into the decision process by the following interleaved sequence:

$$\mathbf{x}_t = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{F}} \;\eta_t \langle \mathbf{x} \,|\, M_t \rangle + D_{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t-1}) \qquad \mathbf{y}_t = \operatorname*{argmin}_{\mathbf{y}\in\mathcal{F}} \;\eta_t \langle \mathbf{y} \,|\, \mathbf{u}^t \rangle + D_{\mathcal{R}}(\mathbf{y}, \mathbf{y}_{t-1})$$

where $D_{\mathcal{R}}$ is the Bregman divergence with respect to $\mathcal{R}$ and $\{\eta_t\}$ is the sequence of step-sizes that can be chosen adaptively.

Theorem 6.21 (Rakhlin and Sridharan [79])**.** *The loss incurred by a learning agent in round t under Optimistic Mirror Descent by choosing action $\mathbf{x}_t \in \mathcal{F}$ with respect to any feasible strategy $\mathbf{x}^*$ is upper bounded by:*

$$\begin{aligned} \langle \mathbf{x}_t - \mathbf{x}^{*,t} \,|\, \mathbf{u}^t \rangle \;\leq\; \left\|\mathbf{u}^t - M_t\right\| \|\mathbf{x}_t - \mathbf{y}_t\| \;+\; \frac{1}{\eta}\left[D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_{t-1}) - D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_t)\right] \\ -\; \frac{1}{2\eta}\left[\|\mathbf{x}_t - \mathbf{y}_t\|^2 \;+\; \left\|\mathbf{x}_t - \mathbf{y}_{t-1}\right\|^2\right] \end{aligned}$$

Fact 6.22. *For any $\rho > 0$ and any numbers a and b: $a \cdot b \leq \frac{\rho}{2}a^2 \;+\; \frac{1}{2\rho}b^2$.*

Fact 6.23. *For any points $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{F}$,*

$$D_{\mathcal{R}}(\mathbf{x}, \mathbf{z}) - D_{\mathcal{R}}(\mathbf{y}, \mathbf{z}) \;\leq\; D_{\mathcal{F}} \|\mathbf{x} - \mathbf{y}\|$$

where $D_{\mathcal{F}} = \max_{a,b\in\mathcal{F}} \|\mathbf{a} - \mathbf{b}\|$.

[3]For simplicity of presentation, we assume the utility function is linear

Fact 6.24. *For any points* $\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{y}_0 \in \mathcal{F}$*,*

$$\sum_t \left\|\mathbf{x}_t - \mathbf{x}_{t-1}\right\|^2 \;\leq\; 2\left(\sum_t \left\|\mathbf{x}_t - \mathbf{y}_0\right\|^2 + \left\|\mathbf{x}_{t-1} - \mathbf{y}_0\right\|^2\right)$$

Theorem 6.13. *The dynamic regret of an agent under Optimistic Mirror Descent with* $M_t = \mathbf{u}^{t-1}$ *with respect to the benchmark sequence of strategies* $\{\mathbf{x}^{*,t}\}_t$ *is upper bounded by:*

$$R_T \;\leq\; \frac{R}{\eta} + \frac{D_{\mathcal{F}}}{\eta}\sum_t \left\|\mathbf{x}^{*,t} - \mathbf{x}^{*,t-1}\right\| + \eta\sum_t \left\|\mathbf{u}^t - \mathbf{u}^{t-1}\right\|_*^2 - \frac{1}{8\eta}\sum_t \left\|\mathbf{x}_t - \mathbf{x}_{t-1}\right\|^2$$

where $R = \sup_x D_{\mathcal{R}}(\mathbf{x}, \mathbf{y}_0)$ *and* $D_{\mathcal{F}} = \max_{a,b\in\mathcal{F}} \|\mathbf{a} - \mathbf{b}\|$.

Proof. By Theorem 6.21 instantiated for $M_t = \mathbf{u}^{t-1}$, we have:

$$\begin{aligned}\left\langle \mathbf{x}_t - \mathbf{x}^{*,t} \,\middle|\, \mathbf{u}^t \right\rangle \;\leq\; \left\|\mathbf{u}^t - \mathbf{u}^{t-1}\right\| \|\mathbf{x}_t - \mathbf{y}_t\| \;+\; \frac{1}{\eta}\left[D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_{t-1}) - D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_t)\right] \\ -\; \frac{1}{2\eta}\left[\|\mathbf{x}_t - \mathbf{y}_t\|^2 \;+\; \left\|\mathbf{x}_t - \mathbf{y}_{t-1}\right\|^2\right]\end{aligned}$$

Using Fact 6.22 and by choosing $\rho = 2\eta$, we can bound the first part of the expression as:

$$\left\|\mathbf{u}^t - \mathbf{u}^{t-1}\right\|_* \|\mathbf{x}_t - \mathbf{y}_t\| \;\leq\; \eta\left\|\mathbf{u}^t - \mathbf{u}^{t-1}\right\|_*^2 + \frac{1}{4\eta}\|\mathbf{x}_t - \mathbf{y}_t\|^2 \tag{6.5}$$

Next, we can sum and rearrange the Bregman divergence terms to get:

$$\sum_{t=1}^{T} D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_{t-1}) - D_R(\mathbf{x}^{*,t}, \mathbf{y}_t) \;\leq\; R + \left(\sum_t D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_{t-1}) - D_R(\mathbf{x}^{*,t-1}, \mathbf{y}_{t-1})\right)$$

where $R = \sup_x D_R(\mathbf{x}, \mathbf{y}_0)$. Using Fact 6.23 in above inequality we get:

$$\sum_{t=1}^{T} D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_{t-1}) - D_{\mathcal{R}}(\mathbf{x}^{*,t}, \mathbf{y}_t) \;\leq\; R + \sum_t D\left\|\mathbf{x}^{*,t} - \mathbf{x}^{*,t-1}\right\| \tag{6.6}$$

Finally, we bound the last part of the expression using Fact 6.24 and observing that in OMD algorithm we choose $\mathbf{x}_0 = \mathbf{y}_0 = \operatorname{argmin}_x R(x)$.

$$\frac{1}{4\eta}\left[\|\mathbf{x}_t - \mathbf{y}_t\|^2 + \left\|\mathbf{x}_t - \mathbf{y}_{t-1}\right\|^2\right] \;\geq\; \frac{1}{8\eta}\sum_t \left\|\mathbf{x}_t - \mathbf{x}_{t-1}\right\|^2.$$

□

Chapter 7

Tracing Equilibrium in Dynamic Markets

A fundamental concept governing trade in large markets is the notion of competitive or market equilibrium. Several classical works, especially in economics, accept the existence of such equilibria as an axiom. Although the existence and characterization of equilibrium in markets has been a topic of interest for many years, only in the last decade or so has there been a renewed interest in the questions of what are some natural dynamics that might drive these markets to equilibrium. Such computational aspects of competitive equilibria is one of the central themes in algorithmic game theory, mainly for the prominent class of Fisher markets. As mentioned in Chapter 4, there are approaches based on distributed adaptation processes, namely tatonnement , that converge to equilibrium in a Fisher market setting. In this same chapter, we also saw how tatonnement provides an explanation how decentralized price adjustment can lead a market into an equilibrium state, thereby providing additional justification for the concept. Recently, several works derived a detailed analysis and proved fast convergence of discrete-time tatonnement in markets [8–11,13,14].

Most of the analysis of tatonnement-based dynamics till now assumes that the market and its properties (agents, budgets, utilities, supplies of goods) remain static and unchanged over time. In fact, to the best of our knowledge, all of the existing work on computation of market equilibrium in algorithmic game theory assumes that the market is essentially a static environment. In contrast, in many (if not all) applications of markets, the market itself is subject to dynamic change, in the sense that supplies of goods changes over time, agents have different budgets at their disposal that they can spend, or the preferences of agents expressed via utility functions evolve over time. Analyzing and quantifying the impact of dynamic change in markets is critical to understand

the robustness of market equilibrium in general, and of price adaptation dynamics like tatonnement in particular.

This chapter continues the algorithmic study of dynamic markets in the form of dynamically evolving environments and specifically focuses on the performance of dynamic adaptation processes like tatonnement. We analyze a discrete-time process, where in each round t tatonnement provides a price for each good, which is then updated using the excess demand for each good. In each round t the excess demand comes from a possibly different, arbitrarily perturbed market. This dynamic nature of markets gives rise to a number of interesting issues. Notably, even when in each round t the market has a unique equilibrium, over time this equilibrium becomes a dynamic object. As such, exact market equilibria can rarely or never be reached. Instead, we consider how tatonnement can trace the equilibrium by maintaining a small distance (in terms of suitably defined notions of distance), which also results in approximate clearing conditions.

In this chapter we study the same model as one in Chapter 5: Fisher markets with buyer utilities exhibiting the constant elasticity of substitution (CES) property. In this versatile framework, we analyze the impact of changes in supply of goods, budgets of agents, and their utility parameters. The adaptation approaches equilibrium conditions, but since the equilibrium is moving, prices and allocations follow and chase the equilibrium point over time. Our analysis provides distance bounds, which can be seen as a quantification of the extent of out-of-equilibrium trade due to the interplay of market perturbation and adaptation of agents.

On a technical level, our analysis primarily focuses on quantifying the impact of perturbation in market parameters on the associated potential function. Our main result then follows from the convergence guarantees for static markets. Moreover, this general approach is shown to constitute a powerful framework to analyze a large variety of protocols and dynamics that are well-understood in static systems, when these systems become subject to dynamic perturbation. This is demonstrated using a rather simple but concrete load balancing system with dynamic machine speeds.

Overview: In this chapter we investigate the effectiveness of the tatonnement price adaptation strategy in terms of its ability to maintain an approximate market-clearing property in dynamic markets. To this end, we focus on a Fisher market with CES utilities consisting of the same set of sellers and buyers but which undergoes a *perpetual* change in one of the market parameters, i.e., supply of sellers, buyer budgets or their utilities.

To quantify the deviation of the market from its market clearing state we use a convex potential function introduced in [46]. In Section 7.2, we quantify the impact of perturbation in the supply of goods, the budgets, and the utility function of buyers. These bounds reveal that the change is often a rather mild additive change in this market potential. Together with the fact that tatonnement decreases the potential multiplicatively, we see that the price adaptation is indeed able to incorporate and adapt to the changes quickly. Overall, the dynamics can trace the equilibrium point up to a distance that evolves from the change in a small number of recent rounds.

The technique we apply for markets can be executed much more generally for a class of dynamical systems, which we outline in Section 7.4. These systems have a set of control parameters (e.g., prices in markets, or strategic decisions in games) and system parameters (e.g., supplies or utilities in markets, or payoff values in games). Moreover, these systems admit a Lyapunov function, and a round-based adaptation process for the control parameters (e.g., tatonnement in markets, or best-response dynamics in classes of games) that multiplicatively decreases the Lyapunov function in a single round. Our results provide a bound on the value of the Lyapunov function when the system parameters are subject to dynamic change. We explicitly discuss such a system in Section 7.4.1.

Related Work

Decentralized adaptation processes such as tatonnement are important due to their simple nature and plausible applicability in real markets. Arrow, Block and Hurwitz [80] showed that a continuous version of tatonnement converges to an equilibrium for markets satisfying the weak gross substitutes (WGS) property. Several algorithmic advances since then provide new insights in analyzing tatonnement [46, 81]. Cole and Fleischer [82] proposed the *ongoing market model*, in which *warehouses* are introduced to allow out-of-equilibrium trade, and prices are updated in tatonnement-style *asynchronously*, to provide an *in-market* process which might capture how real markets work. There has been significant recent interest in further aspects of ongoing markets or asynchronous tatonnement [47, 83–85].

Notions of games and markets with perturbation and dynamic change are only very recently starting to receive increased interest in algorithmic game theory. For example, recent work has started to quantify the average performance of simple auctions and regret-learning agents in combinatorial auctions with dynamic buyer population [86, 87]. In these scenarios, however, equilibria are probabilistic objects and convergence in the

static case can only be shown in terms of regret on average in hindsight. Moreover, the main goal is to bound the price of anarchy.

7.1 Model and Preliminaries

Along the same lines as the model studied in Chapter 5, we consider a Fisher market with n *goods* and m *buyers*, each having CES utility functions with gross-substitutes property. See Chapter 4 for a detailed discussion on this. The notation for the demand and utility functions of the buyers also follows this chapter. The vector of budgets is denoted by $\mathbf{b} = (b_i)_{i=1,\dots,m}$, where $B = \sum_i b_i$ is the total budget in the market.

Dynamic Markets. We study a slightly different model of dynamic market than the ones in previous chapters. In our model, we consider a dynamic market where in the beginning of each round $t = 1, \dots, T$ our tatonnement dynamics propose a vector of prices $\mathbf{p}^t$. Dynamic market parameters like budgets $\mathbf{b}^t$, supplies $\mathbf{w}^t$ and utility functions $\mathbf{u}^t$ are manifested, which can be different from their value in previous rounds $0, \dots, t-1$. Agents request a demand bundle based on the prices $\mathbf{p}^t$ and market $\mathcal{M}^t = (\mathbf{u}^t, \mathbf{b}^t, \mathbf{w}^t)$, which yields a vector of excess demands $\mathbf{z}^t$. Then the system proceeds to the next round $t + 1$.

We first provide a basic insight that lies at the core of the analysis and manages to lift convergence results for a class of static markets to a bound for dynamic markets from that class. Formally, assume that the following properties hold:

Potential: There is a non-negative potential function $\Phi(\mathcal{M}, \mathbf{p})$, for every market $\mathcal{M} = (\mathbf{u}, \mathbf{b}, \mathbf{w})$ and every price vector $\mathbf{p}$. It holds $\Phi(\mathcal{M}, \mathbf{p}) = 0$ if and only if $\mathbf{p}$ is a vector of clearing prices for market $\mathcal{M}$.

Price-Improvement: The price dynamics satisfy $\Phi(\mathcal{M}, \mathbf{p}^t) \leq (1 - \delta) \cdot \Phi(\mathcal{M}, \mathbf{p}^{t-1})$, for some $1 \geq \delta > 0$ and every market $\mathcal{M}$.

Market-Perturbation: The market dynamics satisfy $\Phi(\mathcal{M}^t, \mathbf{p}) \leq \Phi(\mathcal{M}^{t-1}, \mathbf{p}) + \Delta^t$, for some values $\Delta^t \geq 0$ and every price vector $\mathbf{p}$.

Proposition 7.1. *Suppose the price and market dynamics satisfy the Potential, Price-Improvement, and Market-Perturbation properties. Then*

$$\Phi(\mathcal{M}^T, \mathbf{p}^T) \quad \leq \quad (1 - \delta)^T \cdot \Phi(\mathcal{M}^0, \mathbf{p}^0) \; + \; \sum_{t=1}^{T-1} (1 - \delta)^{T-\tau} \Delta^\tau \; .$$

Let $\Delta = \max_{t=1,\dots,T} \Delta^t$*, then it follows for any* $t \leq T$

$$\Phi(\mathcal{M}^T, \mathbf{p}^T) \quad \leq \quad \sum_{\tau=t+1}^{T} (1-\delta)^{T-\tau}\Delta^\tau \; + \; \frac{(1-\delta)^{T-t}}{\delta} \cdot \Delta \; + \; (1-\delta)^T \cdot \Phi(\mathcal{M}^0, \mathbf{p}^0) \; .$$

The proof follows by a direct application of the three properties. We prove it for a much more general class of dynamic systems with Lyapunov functions in Section 7.4.

Consider the three terms in the latter bound for Φ. The first term captures the impact of *recent* changes to the market. The second term bounds the effect of all *older* changes. The third term decays exponentially over time. Hence, when the process runs long enough, the potential is only affected by *recent changes* of the market, while all older changes can be accumulated into a constant term based on Δ and δ. Intuitively, the price dynamics follows the evolution of the equilibrium up to a "distance" of Δ/δ in the potential function value. Hence, if market perturbation Δ is small and price improvement δ is large, the process succeeds to maintain market clearing conditions almost exactly.

7.2 Dynamic Fisher Markets via Convex Potential

In this section, we focus on dynamic Fisher markets through the lens of a convex potential function proposed in [46]. This potential has a natural interpretation as a parameter quantifying the violation of market clearing conditions. In what follows, we use this property of the potential function to quantify the deviations induced by perturbation in market parameters like supply, budgets and buyer utilities.

The convex potential function for a static CES Fisher market is [46]

$$\Psi_{\mathsf{CPF}}(\mathcal{M}, \mathbf{p}) \; = \; \sum_{j=1}^{n} w_j \cdot p_j \; - \; \sum_i b_i \cdot \ln Q_i(\mathbf{p}), \quad \text{where } \; Q_i(\mathbf{p}) \; = \; \left(\sum_{k=1}^{n} (a_{ik})^{1-c} (p_k)^c \right)^{1/c} .$$

Note that $Q_i(\mathbf{p})$ is independent of the supplies of goods and the budgets of buyers; it can be interpreted as the minimum amount of money buyer i needs to use to earn one unit of utility [88]. Since the minimum value of $\Psi_{\mathsf{CPF}}(\mathcal{M}, \mathbf{p})$ is not zero in general we use a normalized version $\Phi_{\mathsf{CPF}}(\mathcal{M}, \mathbf{p}) \; := \; \Psi_{\mathsf{CPF}}(\mathcal{M}, \mathbf{p}) - \Psi^*_{\mathsf{CPF}}(\mathcal{M})$, where $\Psi^*_{\mathsf{CPF}}(\mathcal{M}) := \min_{\mathbf{p}} \Psi_{\mathsf{CPF}}(\mathcal{M}, \mathbf{p})$ to apply our general framework.

We study the following tatonnement price-update rule:

$$p_j^{t+1} \; \leftarrow \; p_j^t \cdot \exp\left(\gamma z_j^t\right), \tag{7.1}$$

where γ is a constant depending on market parameters.

Let $\Psi^*_{\mathsf{CPF}}(\mathcal{M})$ denote the minimum value of the function $\Psi_{\mathsf{CPF}}(\mathcal{M})$. The following theorem, stated in a simplified format from [46], demonstrates the Price-Improvement property.

Theorem 7.2 ([46])**.** *Let* $\mathbf{p}^0$ *denote the initial prices and* $\mathbf{p}^*$ *denote the mark et equilibrium. Suppose prices are updated according to the rule* (7.1)*. If* $\min_j p^0_j/p^*_j \geq q > 0$*, then there exists* $\delta = \delta(q, \lambda) > 0$ *such that for any time* $t \geq 0$*, it holds* $\Psi_{CPF}(\mathcal{M}, \mathbf{p}^{t+1}) - \Psi^*_{CPF}(\mathcal{M}) \leq (1-\delta) \cdot (\Psi_{CPF}(\mathcal{M}, \mathbf{p}^t) - \Psi^*_{CPF}(\mathcal{M}))$*.*

For our dynamic environment, we denote the market at time t by $\mathcal{M}^t = (\mathbf{u}^t, \mathbf{b}^t, \mathbf{w}^t)$, and

$$\Psi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}^t) = \sum_{j=1}^{n} w^t_j \cdot p^t_j - \sum_i b^t_i \cdot \ln Q^t_i(\mathbf{p}^t), \text{ where } Q^t_i(\mathbf{p}) = \left(\sum_{k=1}^{n} (a^t_{ik})^{1-c} (p_k)^c\right)^{1/c}.$$

Let $\Psi^{*,t}_{\mathsf{CPF}} = \min_{\mathbf{p}} \Psi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p})$, and $\Phi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}) = \Psi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}) - \Psi^{*,t}_{\mathsf{CPF}}$.

In the following sections we establish the Market-Perturbation property for the cases when the supplies, budgets and utility functions are dynamic.

7.2.1 Dynamic Supply

In this section, we consider the case when the supplies are changing, while buyers' budgets and utility functions are fixed. Thus, the function Q^t_i and budget b^t_i does not change over time, and we write Q_i and b_i instead.

Proposition 7.3. *A market with changing supplies, keeping other parameters fixed, satisfies the market perturbation property with* $\Delta^t = (P + B) \sum_j |w^{t+1}_j - w^t_j|$*.*

Proof.

$$
\begin{aligned}
&\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;=\; \Psi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) - \Psi^{*,t+1}_{\mathsf{CPF}} \\
\leq\; &\left(\sum_{j=1}^{n} w_j^t \cdot p_j^{t+1} \;-\; \sum_i b_i \cdot \ln Q_i(p^{t+1}) \;-\; \Psi^{*,t}_{\mathsf{CPF}} \right) \;+\; \sum_{j=1}^{n} p_j^{t+1} \cdot \left| w_j^{t+1} - w_j^t \right| \\
&+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}) \\
=\; &\left[\Psi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}^{t+1}) - \Psi^{*,t}_{\mathsf{CPF}} \right] \;+\; \sum_{j=1}^{n} p_j^{t+1} \cdot \left| w_j^{t+1} - w_j^t \right| \;+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}) \\
\leq\; &(1-\delta) \cdot \left[\Psi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}^t) - \Psi^{*,t}_{\mathsf{CPF}} \right] \;+\; P \cdot \sum_{j=1}^{n} \left| w_j^{t+1} - w_j^t \right| \;+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}) \\
=\; &(1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}^t) \;+\; P\|\boldsymbol{\varepsilon}^t\| \;+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}).
\end{aligned}
$$

The last term in the above expression can be bounded as follows: Let $\mathbf{p}^{*,t+1}$ denote the price vector which attains the minimum value of $\Psi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p})$. Then

$$
\begin{aligned}
\Psi^{*,t+1}_{\mathsf{CPF}} \;&=\; \sum_{j=1}^{n} w_j^{t+1} \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i \cdot \ln Q_i(\mathbf{p}^{*,t+1}) \\
&\geq\; \sum_{j=1}^{n} w_j^t \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i \cdot \ln Q_i(\mathbf{p}^{*,t+1}) \;-\; \sum_{j=1}^{n} \mathbf{p}_j^{*,t+1} \cdot \left| w_j^{t+1} - w_j^t \right| \\
&\geq\; \Psi^{*,t}_{\mathsf{CPF}} \;-\; \sum_{j=1}^{n} \mathbf{p}_j^{*,t+1} \cdot \left| w_j^{t+1} - w_j^t \right| \qquad \text{(by definition of } \Psi^{*,t}_{\mathsf{CPF}}) \\
&\geq\; \Psi^{*,t}_{\mathsf{CPF}} \;-\; B\|\boldsymbol{\varepsilon}^t\|_1.
\end{aligned}
$$

The last inequality holds since each equilibrium price is bounded above by B, the total amount of money in the market. Thus, $(\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}})$ is bounded above by $B\|\boldsymbol{\varepsilon}^t\|$. Summarizing,

$$
\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;\leq\; (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^t, \mathbf{p}^t) \;+\; (P+B)\|\boldsymbol{\varepsilon}^t\|_1,
$$

i.e., $\Delta^t = (P+B)\|\boldsymbol{\varepsilon}^t\|_1$.

□

7.2.2 Dynamic Budgets

In this section, we consider the case when the buyers' budgets are changing, while supplies and buyers' utility functions are fixed.

Proposition 7.4. *A market with changing buyers' budgets, keeping other parameters fixed, satisfies the market perturbation property with* $\Delta^t = C' \sum_i |b_i^{t+1} - b_i^t|$. *for a constant* C'.

Proof.

$$\begin{aligned}
&\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;=\; \Psi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) - \Psi^{*,t+1}_{\mathsf{CPF}} \\
=\;& \left(\sum_{j=1}^{n} w_j \cdot p_j^{t+1} \;-\; \sum_i b_i^t \cdot \ln Q_i(\mathbf{p}^{t+1}) \;-\; \Psi^{*,t}_{\mathsf{CPF}} \right) \;-\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln Q_i(\mathbf{p}^{t+1}) \\
&+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}) \\
=\;& \left[\Psi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t+1}) - \Psi^{*,t}_{\mathsf{CPF}} \right] \;-\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln Q_i(\mathbf{p}^{t+1}) \;+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}) \\
\leq\;& (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;-\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln Q_i(\mathbf{p}^{t+1}) \;+\; (\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}}).
\end{aligned}$$

Using a similar approach as in the previous section, we can bound $(\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}})$.

$$\begin{aligned}
\Psi^{*,t+1}_{\mathsf{CPF}} \;&=\; \sum_{j=1}^{n} w_j \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i^{t+1} \cdot \ln Q_i(\mathbf{p}^{*,t+1}) \\
&=\; \sum_{j=1}^{n} w_j \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i^{t} \cdot \ln Q_i(\mathbf{p}^{*,t+1}) \;-\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln Q_i(\mathbf{p}^{*,t+1}) \\
&\geq\; \Psi^{*,t}_{\mathsf{CPF}} \;-\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln Q_i(\mathbf{p}^{*,t+1}).
\end{aligned}$$

Combining the above two inequalities yields

$$\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;\leq\; (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;+\; \sum_i (b_i^{t+1} - b_i^t) \cdot \ln \frac{Q_i(\mathbf{p}^{*,t+1})}{Q_i(\mathbf{p}^{t+1})}.$$

Cheung et al. [46, Section 6.3] showed that in the static market setting, if the initial prices are neither too high nor too low, then $\frac{Q_i(\mathbf{p}^{*,t+1})}{Q_i(\mathbf{p}^{t+1})}$ has time-independent upper and lower bounds. In the dynamic market setting, we assume that there exists a constant $C \geq 1$ such that the budget of each buyer i changes within the range $[b_i^0/C, C \cdot b_i^0]$. Let U^*, L^* be the time-independent upper and lower bounds derived in [46], for the static market setting with $\mathbf{b} = (b_1^0, \ldots, b_m^0)$. Following the argument in [46], their upper bound on p_k^{t+1} can be carried through to the dynamic market setting by increasing by a factor of C, while their lower bound on p_k^{t+1} can be carried through to the dynamic market setting by shrinking by a factor of $1/C$; these hold similarly for the equilibrium prices. Thus,

for the dynamic market setting, we have time-independent upper and lower bounds on $\frac{Q_i(\mathbf{p}^{*,t+1})}{Q_i(\mathbf{p}^{t+1})}$ of values $C^2 \cdot U^*$ and L^*/C^2 respectively. Thus, by setting

$$C' \;:=\; \max\left\{\; \left|\ln(C^2 \cdot U^*)\right| \;,\; \left|\ln(L^*/C^2)\right| \;\right\},$$

we have

$$\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;\leq\; (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;+\; C' \cdot \sum_i \left|b_i^{t+1} - b_i^t\right|,$$

i.e., $\Delta^t = C' \cdot \sum_i \left|b_i^{t+1} - b_i^t\right|$.

□

7.2.3 Dynamic Buyer Utility

In this section, we consider the case when the buyers' utility function are changing, while supplies and budgets are fixed. In this case, changes to utility functions induce changes to the functions Q_i^t.

Proposition 7.5. *A market with changing buyers' utility functions, keeping other parameters fixed, satisfies the market perturbation property with* $\Delta^t = 2B \ln \chi^t$*, where* $\chi^t = \max_{i,j}((\chi_{ij}^t)^{-1/\rho}, (\chi_{ij}^t)^{1/\rho})$*. Here* χ_{ij} *denotes the multiplicative change in utility value* a_{ij}*.*

Proof.

$$\begin{aligned}
&\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;=\; \Psi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) - \Psi_{\mathsf{CPF}}^{*,t+1} \\
=\; &\left(\sum_{j=1}^{n} w_j \cdot p_j^{t+1} \;-\; \sum_i b_i \cdot \ln Q_i^t(\mathbf{p}^{t+1}) \;-\; \Psi_{\mathsf{CPF}}^{*,t}\right) \;-\; \sum_i b_i \cdot \ln \frac{Q_i^{t+1}(\mathbf{p}^{t+1})}{Q_i^t(\mathbf{p}^{t+1})} \\
&\quad +\; (\Psi_{\mathsf{CPF}}^{*,t} - \Psi_{\mathsf{CPF}}^{*,t+1}) \\
=\; &\left[\Psi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t+1}) - \Psi_{\mathsf{CPF}}^{*,t}\right] \;-\; \sum_i b_i \cdot \ln \frac{Q_i^{t+1}(\mathbf{p}^{t+1})}{Q_i^t(\mathbf{p}^{t+1})} \;+\; (\Psi_{\mathsf{CPF}}^{*,t} - \Psi_{\mathsf{CPF}}^{*,t+1}) \\
\leq\; &(1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;-\; \sum_i b_i \cdot \ln \frac{Q_i^{t+1}(\mathbf{p}^{t+1})}{Q_i^t(\mathbf{p}^{t+1})} \;+\; (\Psi_{\mathsf{CPF}}^{*,t} - \Psi_{\mathsf{CPF}}^{*,t+1}).
\end{aligned}$$

$(\Psi^{*,t}_{\mathsf{CPF}} - \Psi^{*,t+1}_{\mathsf{CPF}})$ can be bounded as follows:

$$\begin{aligned}
\Psi^{*,t+1}_{\mathsf{CPF}} &= \sum_{j=1}^{n} w_j \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i \cdot \ln Q_i^{t+1}(\mathbf{p}^{*,t+1}) \\
&= \sum_{j=1}^{n} w_j \cdot \mathbf{p}_j^{*,t+1} \;-\; \sum_i b_i \cdot \ln Q_i^{t}(\mathbf{p}^{*,t+1}) \;-\; \sum_i b_i \cdot \ln \frac{Q_i^{t+1}(\mathbf{p}^{*,t+1})}{Q_i^{t}(\mathbf{p}^{*,t+1})} \\
&\geq \Psi^{*,t}_{\mathsf{CPF}} \;-\; \sum_i b_i \cdot \ln \frac{Q_i^{t+1}(\mathbf{p}^{*,t+1})}{Q_i^{t}(\mathbf{p}^{*,t+1})}.
\end{aligned}$$

Combining yields

$$\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;\leq\; (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;+\; \sum_i b_i \cdot \ln \left(\frac{Q_i^{t+1}(\mathbf{p}^{*,t+1})}{Q_i^{t}(\mathbf{p}^{*,t+1})} \cdot \frac{Q_i^{t}(\mathbf{p}^{t+1})}{Q_i^{t+1}(\mathbf{p}^{t+1})} \right).$$

Starting from the initial utility values, each a_{ij} can in each round be changed by some multiplicative factor χ_{ij}^t. Let $\chi^t = \max_{i,j}((\chi_{ij}^t)^{-1/\rho}, (\chi_{ij}^t)^{1/\rho})$ and $\chi = \max_t \chi^t$. Note that $(1-c)/c = -1/\rho$, so $1/\chi^t \leq Q_i^{t+1}(\mathbf{p})/Q_i^t(\mathbf{p}) \leq \chi^t$ for any price vector $\mathbf{p}$. Thus,

$$\left| \ln \left(\frac{Q_i^{t+1}(\mathbf{p}^{*,t+1})}{Q_i^{t}(\mathbf{p}^{*,t+1})} \cdot \frac{Q_i^{t}(\mathbf{p}^{t+1})}{Q_i^{t+1}(\mathbf{p}^{t+1})} \right) \right| \;\leq\; 2 \ln \chi^t,$$

and hence

$$\Phi_{\mathsf{CPF}}(\mathcal{M}^{t+1}, \mathbf{p}^{t+1}) \;\leq\; (1-\delta) \cdot \Phi_{\mathsf{CPF}}(\mathcal{M}^{t}, \mathbf{p}^{t}) \;+\; 2B \ln \chi^t,$$

i.e., $\Delta^t = 2B \ln \chi^t$. □

7.3 Connections to Bounds on Revenue Loss

Up until now, we have focused on the tatonnement price update with goal of analyzing its robustness to arbitrary changes in market parameters. In the previous sections, we showed that indeed, on account of the fact that tatonnement converges linearly to equilibrium, even in the case when market parameters are subject to perturbation, tatonnement ensures that the market stays in a state of *approximate* equilibrium, where the state of the market is measured with respect to a convex potential function. This approximation however naturally depends on the magnitude of these perturbations.

The reader may recall that in Chapter 5, we established a connection between the value of the potential function in any given round to the loss incurred by any seller in the same round. For the same tatonnement updates as considered here, we showed a bound on the loss in the revenue of any seller, albeit in static markets. One can however, as well borrow this analysis and plug-in the value of the potential of a perturbed market

to derive analogous bounds on the loss incurred by any seller. We note here, that this bound would now be a function of the magnitude of the changes in the potential function induced by the perturbations encountered. The magnitude of these changes is exactly what we have analyzed in Sections 7.2.

7.4 Parametrized Lyapunov Dynamical Systems

In this section, we prove a general theorem, which includes as special case the bound shown for markets in Proposition 7.1. Our focus here are dynamical systems, in which time is discrete and represented by non-negative integers. Note, however, that the formulation below can be easily generalized to settings with continuous time.

We assume that the dynamical system can be described by two sets of parameters. There is a set of *control variables* that can be adjusted by an algorithm or a protocol. In addition, there is a set of *system parameters* that can change in each round in an adversarial way. For example, in our analysis of markets in the previous section, the control variables are prices of goods, whereas system parameters can be supplies of goods, budgets of agents, or utility parameters. As another example, in games the control variables could be the strategy choices of agents, whereas system parameters are utility and payoff values of states.

The classical theory of dynamical systems often studies the behaviour of systems with static system parameters. However, dynamical systems with varying system parameters often arise in practice (see Section 7.4.1 for an example). Here, we propose a simple framework to analysis Lyapunov dynamical systems with varying system parameters. More formally, the dynamical system L is described by an initial *control variable vector* $\mathbf{p}^0 \in \mathbb{R}^n$ and an *evolution rule* $F : \mathbb{R}^n \to \mathbb{R}^n$, which specifies how the control variables are adjusted. For each time $t \geq 1$, we have $\mathbf{p}^t = F(\mathbf{p}^{t-1})$.

The system L is called a *Lyapunov dynamical system* (LDS) if it admits a Lyapunov function $G : \mathbb{R}^n \to \mathbb{R}^+$ such that

(a) for every fixed point (equilibrium) $\mathbf{p}$ of F with $F(\mathbf{p}) = \mathbf{p}$ it holds $G(\mathbf{p}) = 0$;

(b) for every $\mathbf{p} \in \mathbb{R}^n$ it holds $G(F(\mathbf{p})) \leq G(\mathbf{p})$.

An LDS L is called *linearly converging* (LCLDS) if it further satisfies

(c) there exists a *decay parameter* $\delta = \delta(L) > 0$ such that for any $\mathbf{p} \in \mathbb{R}^n$,

$$G(F(\mathbf{p})) \leq (1 - \delta) \cdot G(\mathbf{p}).$$

Let $\mathcal{L}$ be a family of dynamical systems, while each dynamical system $L_{\mathbf{s}} \in \mathcal{L}$ is parametrized by a *system parameter* vector $\mathbf{s} \in \mathbb{R}^d$. The family $\mathcal{L}$ is called a family of *parametrized, linearly converging LDS* (PLCLDS) if each $L_{\mathbf{s}} \in \mathcal{L}$ is an LCLDS and $\delta(\mathcal{L}) = \inf_{L_{\mathbf{s}} \in \mathcal{L}} \delta(L_{\mathbf{s}}) > 0$. For each $L_{\mathbf{s}}$, we denote its evolution rule by $F_{\mathbf{s}}$ and its Lyapunov function by $G_{\mathbf{s}}$.

In many scenarios, particularly in agent-based dynamical systems, the control variables $\mathbf{p}$ change by the evolution rule that expresses, e.g., the sequential decisions of the agents, but the system parameters $\mathbf{s}$ can change in an exogeneous (or even adversarial) way. However, in many cases the impact of changes in a single time step is rather mild. The following theorem states our *recovery result* by relating the Lyapunov value to the magnitude of changes in each step. Intuitively, it characterizes the "distance" that the evolution rule maintains to a fixed point over the course of the dynamics.

Theorem 7.6. *Let $\mathcal{L}$ be a PLCLDS with $\delta \equiv \delta(\mathcal{L}) > 0$, let $\mathbf{s}^0, \mathbf{s}^1, \ldots, \mathbf{s}^T$ denote the system parameter vectors at times $0, 1, \cdots, T$, respectively, and let $\Phi(\mathbf{s}^t, \mathbf{p}^t) = G_{\mathbf{s}^t}(\mathbf{p}^t)$. Suppose that for every $t = 1, \ldots, T$ the system parameters $\mathbf{s}^{t-1}, \mathbf{s}^t \in \mathbb{R}^d$ invoke a change such that for every $\mathbf{p} \in \mathbb{R}^n$, we have $\Phi(\mathbf{s}^t, \mathbf{p}) \leq \Phi(\mathbf{s}^{t-1}, \mathbf{p}) + \Delta^t$. The initial control variable vector is denoted by $\mathbf{p}^0$, and the system evolves such that for every $t \geq 1$ we have $\mathbf{p}^t = F_{\mathbf{s}_{t-1}}(\mathbf{p}^{t-1})$. Then*

$$\Phi(\mathbf{s}^T, \mathbf{p}^T) \;\leq\; (1-\delta)^T \cdot \Phi(\mathbf{s}^0, \mathbf{p}^0) \;+\; \sum_{t=1}^{T} (1-\delta)^{T-t} \cdot \Delta^t \;.$$

Let $\Delta = \max_{t=1,\ldots,T} \Delta^t$, then it follows for any $t \leq T$

$$\Phi(\mathbf{s}^T, \mathbf{p}^T) \quad\leq\quad \sum_{\tau=t+1}^{T} (1-\delta)^{T-\tau} \Delta^\tau \;+\; \frac{(1-\delta)^{T-t}}{\delta} \cdot \Delta \;+\; (1-\delta)^T \cdot \Phi(\mathbf{s}^0, \mathbf{p}^0) \;.$$

Proof. For any time $t \geq 1$,

$$\begin{aligned}
\Phi(\mathbf{s}^t, \mathbf{p}^t) \;&=\; G_{\mathbf{s}_t}(\mathbf{p}^t) \;\leq\; G_{\mathbf{s}_{t-1}}(\mathbf{p}^t) + \Delta^t \\
&=\; G_{\mathbf{s}_{t-1}}(F_{\mathbf{s}_{t-1}}(\mathbf{p}^{t-1})) + \Delta^t \\
&\leq\; (1-\delta) \cdot G_{\mathbf{s}_{t-1}}(\mathbf{p}^{t-1}) + \Delta^t \quad=\; (1-\delta) \cdot \Phi(\mathbf{s}^{t-1}, \mathbf{p}^{t-1}) + \Delta^t.
\end{aligned}$$

Iterating the above recurrence yields the first result. For the second result, note that

$$\begin{aligned}
\sum_{\tau=1}^{t} (1-\delta)^{T-\tau} \Delta^\tau \;&\leq\; \Delta (1-\delta)^T \cdot \sum_{\tau=1}^{t} \left(\frac{1}{1-\delta}\right)^\tau \\
&=\; \Delta \cdot \frac{(1-\delta)^{T+1}}{\delta} \cdot \left(\left(\frac{1}{1-\delta}\right)^{t+1} - \frac{1}{1-\delta} \right) \;<\; \Delta \cdot \frac{(1-\delta)^T}{\delta} \cdot \left(\frac{1}{1-\delta}\right)^t . \;\square
\end{aligned}$$

In the scenarios where $\sum_{t=1}^{T} \Delta^t = O(T^\alpha)$ for small constant α, we have the following corollary.

Corollary 7.7. *In the setting of Theorem 7.6, if* $\sum_{t=1}^{T} \Delta^t = \mathcal{O}(T^\alpha)$ *for some constant* $\alpha > 0$*, then for any constant* $\beta > 0$*,*

$$\Phi(\mathbf{s}^T, \mathbf{p}^T) \leq \sum_{\tau=T-\left\lceil \frac{\alpha+\beta}{\delta} \log T \right\rceil+1}^{T} \Delta^\tau + \mathcal{O}(T^{-\beta}) + (1-\delta)^T \cdot \Phi(\mathbf{s}^0, \mathbf{p}^0).$$

As $T \to \infty$, the last two terms of the above inequality diminish. The bound is dominated by the first term, which describes the impact of the changes in the *recent* $\mathcal{O}\left(\frac{\log T}{\delta}\right)$ steps.

7.4.1 Load Balancing with Dynamic Machine Speed

Consider a setting with n distinct machines connected to each other to form an arbitrary network. For ease of notation, we label the machines as m_i for $i = 1$ to n. Each machine m_i can process jobs at speed s_i. Jobs/tasks, assumed to be infinitely divisible, of total weight M are arbitrarily distributed over the network. Our goal is to design a decentralized load balancing algorithm with the objective that the total processing time over all machines is minimized.

Algorithm 8 Diffusion

```
for t = 1 to T do
    for each machine m_i do
        f_i^(t) ← total processing time on m_i
        broadcast f_i^(t) to all j ∈ nbd(m_i)
        for all j ∈ nbd(m_i) do
            if f_i^(t) > f_j^(t) then
                Send P_ij(f_i^(t) − f_j^(t))s_i load to j
            end if
        end for
    end for
end for
```

Before proceeding, we set up some notation. $\mathbf{s}$ denotes the vector of machine speeds. $\boldsymbol{\ell}^{(t)} = (\ell_i^{(t)})_i$ denotes the vector of loads and $\mathbf{f}^{(t)} = (f_i^{(t)})_i$ the corresponding finishing times at round t. We assume throughout that the total load stays constant i.e. $\sum_i \ell_i^{(t)} = M$. For machine speed $\mathbf{s}$, $f^{*,\mathbf{s}}$ denotes the corresponding vector of finishing times in the balanced state, i.e., a state where the finishing time of all machines is the same.

Algorithm 8 is based on the diffusion principle [89], where if a machine has more jobs than its neighbours, then some jobs diffuse to the neighbour. In our context, since the

goal is to equalize the finishing times of all machines, the number of jobs that diffuse is proportional to the difference in the finishing times. The proportionality constant depends on the connecting edge. Specifically, in the algorithm that follows we use a diffusivity matrix P satisfying the following conditions: (a) $P_{ii} \geq 1/2$ (b) $P_{ij} > 0$ iff (i, j) is an edge in G. (c) P is symmetric and stochastic, i.e., for every machine m_i, $\sum_j P_{ij} = 1$.

If each machine m_i uses the load balancing protocol as described above, then the finishing time of machine m_i at time $t + 1$ is:

$$\begin{aligned} f_i^{(t+1)} &= \frac{\ell_i^{(t+1)}}{s_i} = \frac{1}{s_i}\left(\ell_i^{(t)} - \sum_{j=1}^{n} P_{ij}\left(\frac{\ell_i^{(t)}}{s_i} - \frac{\ell_j^{(t)}}{s_j}\right)s_i\right) \\ &= \frac{1}{s_i}\left(\ell_i^{(t)} - \sum_{j=1}^{n} P_{ij}\ell_i^{(t)} + \sum_{j=1}^{n} P_{ij}\frac{\ell_j^{(t)}}{s_j}s_i\right) \\ &= \frac{1}{s_i}\left(\sum_{j=1}^{n} P_{ij} f_j^{(t)} s_i\right) = \left(P f^{(t)}\right)_i. \end{aligned}$$

It therefore follows that $\mathbf{f}^{(t+1)} = P\mathbf{f}^{(t)}$. Further, in the balanced state $\mathbf{f}^*$, since the finishing time of all machines is the same,

$$(P\mathbf{f}^*)_i = \sum_{j=1}^{n} P_{ij} f_j^* = f_i^* \sum_{j=1}^{n} P_{ij} = f_i^*,$$

i.e., $P\mathbf{f}^* = \mathbf{f}^*$. If we denote the *error* in round $t + 1$ by $\mathbf{e}^{(t+1)}$, then:

$$\mathbf{e}^{(t+1)} = \mathbf{f}^{(t+1)} - \mathbf{f}^* = P(\mathbf{f}^{(t)} - \mathbf{f}^*) = P\mathbf{e}^{(t)},$$

i.e., the same transformations apply to the error vector as well. Since P is a symmetric matrix, it has n eigenvalues $\lambda_1, \lambda_2 \cdots \lambda_n$ and linearly independent corresponding eigenvectors. By the theory of Markov chains, it is also known that $1 = |\lambda_1| \geq |\lambda_2| \geq \cdots |\lambda_n|$. Since P scales the length of $\mathbf{e}^{(t)}$ by a factor of at most $|\lambda_2|$:

$$\left\|\mathbf{e}^{(t+1)}\right\| = \left\|P\mathbf{e}^{(t)}\right\| \leq |\lambda_2|\left\|\mathbf{e}^{(t)}\right\| \Rightarrow \left\|\mathbf{e}^{(t+1)}\right\| \leq |\lambda_2|^t \left\|\mathbf{e}^{(0)}\right\|. \tag{7.2}$$

For a given speed vector $\mathbf{s}$, one can define the "potential" as the normed distance: $\left\|\mathbf{f}^{(t)} - \mathbf{f}^{*,\mathbf{s}}\right\|_1$. This measures the imbalance in the network in terms of the finishing times. From (7.2), since the error vector $\mathbf{e}$ converges to zero linearly, the potential at the balanced state is zero. Note that this load balancing setting is an instance of the Lyapunov dynamical system introduced in Section 7.4. Specifically, the speed vector $\mathbf{s}$ is the *system parameter*, the evolution function $F(\boldsymbol{\ell}^{(t)})$ is the diffusion process

as described in Algorithm 8 and the potential as mentioned above corresponds to the Lyapunov function $G_{\mathbf{s}}(\boldsymbol{\ell}^{(t)}) = G^t_{\mathbf{s}}$. Note that by (7.2) it follows that $G^{t+1}_{\mathbf{s}} \leq |\lambda_2|^t G^0_{\mathbf{s}}$. In the following, all norms are assumed to be L1 norms.

Proposition 7.8. *For a speed vector* $\mathbf{s}$ *and an arbitrary load profile vector* $\boldsymbol{\ell}$*, let* $\mathbf{f}$ *denote the corresponding finishing time vector. For a Lyapunov function defined as* $G_{\mathbf{s}} = \|\mathbf{f} - \mathbf{f}^{*,\mathbf{s}}\|$*, if the speed vector changes to* $\mathbf{s}'$ *for the same load profile, then:*

$$G_{\mathbf{s}'} \leq G_{\mathbf{s}} + Mn\left|\frac{1}{\|\mathbf{s}'\|} - \frac{1}{\|\mathbf{s}\|}\right|.$$

Proof. For a change in the speed vector to $\mathbf{s}'$ with the same load profile, the Lyapunov function is given by:

$$G_{\mathbf{s}'} = \left\|\mathbf{f} - \mathbf{f}^{*,\mathbf{s}'}\right\| \leq \|\mathbf{f} - \mathbf{f}^{*,\mathbf{s}}\| + \left\|\mathbf{f}^{*,\mathbf{s}'} - \mathbf{f}^{*,\mathbf{s}}\right\| = G_{\mathbf{s}} + \left\|\mathbf{f}^{*,\mathbf{s}'} - \mathbf{f}^{*,\mathbf{s}}\right\|. \tag{7.3}$$

Let ℓ_i denote the load on machine m_i. Using the underlying symmetry, we can claim that the load on any machine m_i in the balanced state and its corresponding finishing time are $\ell^*_i = \frac{s_i \cdot M}{\sum_k s_k}$ and $f^{*,\mathbf{s}}_i = \frac{\ell^*_i}{s_i} = \frac{M}{\sum_k s_k}$ respectively. It then follows that:

$$\left\|\mathbf{f}^{*,\mathbf{s}'} - \mathbf{f}^{*,\mathbf{s}}\right\| = \sum_i \left|\frac{\ell'^*_i}{s'_i} - \frac{\ell^*_i}{s_i}\right| = \sum_i \left|\frac{M}{\sum_k s'_k} - \frac{M}{\sum_k s_k}\right| = Mn\left|\frac{1}{\|\mathbf{s}'\|} - \frac{1}{\|\mathbf{s}\|}\right|.$$

□

To formalize the problem, let $\mathcal{LB}(N, M)$ be a family of load balancing environments where N denotes the network of underlying machines and M the total weight of jobs. Each individual environment $LB_{\mathbf{s}} \in \mathcal{LB}(N, M)$ is parameterized by the machine-speed vector $\mathbf{s}$. The corresponding potential (also Lyapunov) function is denoted by $G_{\mathbf{s}}$.

Proposition 7.9. *Let* $\mathcal{LB}(N, M)$ *be a family of load balancing environments on* n *machines with the corresponding diffusivity matrix being* P_N*. Let* $\mathbf{s}^0, \mathbf{s}^1, \cdots, \mathbf{s}^T$ *denote the vector of machine speeds at times* $0, 1 \cdots T$ *respectively. If we denote by* λ_2 *the second largest eigenvalue of* P_N *and* $\Phi(\mathbf{s}^t, \boldsymbol{\ell}^t) := G_{\mathbf{s}^t}(\boldsymbol{\ell}^t)$*, then*

$$\Phi(\mathbf{s}^T, \boldsymbol{\ell}^T) \leq |\lambda_2|^T \cdot \Phi(\mathbf{s}^0, \boldsymbol{\ell}^0) + Mn\sum_{t=1}^{T} |\lambda_2|^{T-t} \cdot \left|\frac{1}{\|\mathbf{s}^{t+1}\|} - \frac{1}{\|\mathbf{s}^t\|}\right| .$$

Proof. The result follows from the fact that $G^{t+1}_{\mathbf{s}} \leq |\lambda_2|\, G^t_{\mathbf{s}}$ and Theorem 7.6. □

Since Φ is a measure of load imbalance in the network in terms of finishing times, the above theorem implies that if the change in the speed vectors across rounds is small, then the imbalance at time T is small and depends largely on the most recent changes.

Chapter 8

Conclusions

In this thesis, we focused on a rather broad class of problems ranging from product recommendations to a website visitor to pricing strategies in large markets. In all of the problems considered, we identified a key property of the model and designed approaches that take advantages of it. For example, in Chapter 2, we exploited the trend structure that is often inherent in several loss models. Similarly, in Chapter 3, we were able to design efficient algorithms by making use of the stochastic nature of channel performance. The approaches used in both these chapters although inspired from solutions to the classical problems, need significant changes to achieve stronger performance guarantees. This suggests that to design algorithms for sequential learning problems based on non-standard models, it is crucial to identify and characterise the key properties of the model, since black-box approaches are often too broad and cannot account for the model subtleties.

The insights obtained from the pricing problems we study are much deeper. Firstly, we note that the tatonnement update method, which has been widely studied in both algorithms and economics circle, has not yet shown to be individually rational i.e., it is not clear that this (and exactly this) price update is in the best interest of each seller. In chapters 5 and 7, we demonstrated that this price update not only leads to fast convergence to equilibrium, or approximate convergence in the case of dynamic markets, but also ensures revenue optimality. One can therefore argue that the tatonnement update method, as studied in this thesis, is a result of the revenue optimizing actions of the sellers. Our results, however, only apply to the gross-substitutes class of markets. Proving similar connections for a larger class of markets is still open. For the class of dynamic markets, we observed that some specialized forms of no-regret dynamics and prediction techniques for supply estimation can also be used. Although their performance is

weaker than the tatonnement update for the specific class of CES utility functions, these techniques are applicable to a larger class of functions.

Bibliography

[1] B. Kveton, Z. Wen, A. Ashkan, H. Eydgahi, and B. Eriksson. Matroid bandits: Fast combinatorial optimization with learning. *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, pages 420–429, 2014.

[2] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. Fast convergence of regularized learning in games. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, pages 2989–2997, Cambridge, MA, USA, 2015. MIT Press. URL `http://dl.acm.org/citation.cfm?id=2969442.2969573`.

[3] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proc. 20th Intl. Conf. Machine Learning (ICML'03)*, pages 928–936, 2003.

[4] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[5] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[6] Jean-Yves Audibert and Sébastien Bubeck. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11(Oct):2785–2836, 2010.

[7] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[8] Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.

[9] Yevgeny Seldin and Aleksandrs Slivkins. One practical algorithm for both stochastic and adversarial bandits. In *International Conference on Machine Learning*, pages 1287–1295, 2014.

[10] Elad Hazan and Satyen Kale. Better algorithms for benign bandits. *Journal of Machine Learning Research*, 12(Apr):1287–1311, 2011.

[11] Robin Allesiardo and Raphaël Féraud. Exp3 with drift detection for the switching bandit problem. In *Data Science and Advanced Analytics (DSAA), 2015. 36678 2015. IEEE International Conference on*, pages 1–7. IEEE, 2015.

[12] Taishi Uchiya, Atsuyoshi Nakamura, and Mineichi Kudo. Algorithms for adversarial bandit problems with multiple plays. In *Algorithmic learning theory*, pages 375–389. Springer, 2010.

[13] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pages 535–543, 2015.

[14] Jean-Yves Audibert, Sébastien Bubeck, and Gábor Lugosi. Regret in online combinatorial optimization. *Mathematics of Operations Research*, 39(1):31–45, 2013.

[15] Levente Kocsis and Csaba Szepesvári. Discounted ucb. In *2nd PASCAL Challenges Workshop*, pages 784–791, 2006.

[16] Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*, 2008.

[17] T Ryan Hoens, Robi Polikar, and Nitesh V Chawla. Learning from streaming data with concept drift and imbalance: an overview. *Progress in Artificial Intelligence*, 1(1):89–101, 2012.

[18] Joao Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. Learning with drift detection. In *Brazilian Symposium on Artificial Intelligence*, pages 286–295. Springer, 2004.

[19] Sébastien Bubeck and Nicolo Cesa-Bianchi. *Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems*, volume 5. 2012. doi: 10.1561/2200000024.

[20] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, 58(301):13–30, 1963.

[21] Y. Gai, B. Krishnamachari, and R. Jain. Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. *IEEE Symposium on New Frontiers in Dynamic Spectrum*, 2010.

[22] O. Avner and S. Mannor. Multi-user lax communications: a multi-armed bandit approach. *IEEE INFOCOM*, 2016.

[23] Naumaan Nayyar, Dileep Kalathil, and Rahul Jain. Decentralized learning for multi-player multi-armed bandits. *CDC*, 2012.

[24] Naumaan Nayyar, Dileep Kalathil, and Rahul Jain. On regret-optimal learning in decentralized multi-player multi-armed bandits. 2015.

[25] Jonathan Rosenski, Ohad Shamir, and Liran Szlak. Multi-player bandits - a musical chairs approach. *33rd International Conference on Machine Learning (ICML)*, 2016.

[26] L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems, November 2005, Volume 11, Issue 3*, pages 387–434, 2005.

[27] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Minimizing regret with label efficient prediction. *IEEE Transactions on Information Theory, vol. 51*, 2005.

[28] Y. Abbasi-Yadkori, D. Pal, and C. Szepesvari. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems*, 2011.

[29] P. Auer. Using confidence bounds for exploitationexploration trade-offs. *Journal of Machine Learning Research*, 2002.

[30] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 2002.

[31] J. Langford and T. Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. *Advances in Neural Information Processing Systems (NIPS)*, 2007.

[32] G. Neu and G. Bartok. An efficient algorithm for learning with semi-bandit feedback. *24th International Conference on Algorithmic Learning Theory*, pages 234–248, 2013.

[33] Nicolo Cesa-Bianchi, Ofer Dekel, and Ohad Shamir. Online learning with switching costs and other adaptive adversaries. In *Advances in Neural Information Processing Systems*, pages 1160–1168, 2013.

[34] Ofer Dekel, Jian Ding, Tomer Koren, and Yuval Peres. Bandits with switching costs: T 2/3 regret. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 459–467. ACM, 2014.

[35] Gergely Neu and Gábor Bartók. An efficient algorithm for learning with semi-bandit feedback. In *International Conference on Algorithmic Learning Theory*, pages 234–248. Springer, 2013.

[36] J Michael Harrison, N Bora Keskin, and Assaf Zeevi. Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, 58(3):570–586, 2012.

[37] N. Bora Keskin and Assaf Zeevi. Chasing demand: Learning and earning in a changing environment. *Mathematics of Operations Research*, 42(2):277–307, 2017. doi: 10.1287/moor.2016.0807. URL `https://doi.org/10.1287/moor.2016.0807`.

[38] Vijay Vazirani. Combinatorial algorithms for market equilibria. In Nisan et al. [90], chapter 7.

[39] Bruno Codenotti and Kasturi Varadarajan. Computation of market equilibria by convex programming. In Nisan et al. [90], chapter 8.

[40] Léon Walras. *Éléments d'économie politique pure, ou théorie de la richesse sociale (Elements of Pure Economics, or the theory of social wealth).* Lausanne, Paris, 1874. (1899, 4th ed.; 1926, rev ed., 1954, Engl. transl.).

[41] Kenneth Arrow, Henry Block, and Leonid Hurwicz. On the stability of the competitive equilibrium: II. *Econometrica*, 27(1):82–109, 1959.

[42] Herbert Scarf. Some examples of global instability of the competitive equilibrium. *Intl. Econom. Rev.*, 1:157–172, 1960.

[43] Christopher Anderson, Charles Plott, Ken-Ichi Shimomura, and Sander Granat. Global instability in experimental general equilibrium: The Scarf example. *J. Econom. Theory*, 115(2):209–249, 2004.

[44] Sean Crockett, Ryan Oprea, and Charles Plott. Extreme Walrasian dynamics: The Gale example in the lab. *Amer. Econ. Rev.*, 101(7):3196–3220, 2011.

[45] Charles Plott. Market stability: Backward bending supply in a laboratory experimental market. *Economic Inquiry*, 38(1):1–18, 2000.

[46] Yun Kuen Cheung, Richard Cole, and Nikhil R. Devanur. Tatonnement beyond gross substitutes? Gradient descent to the rescue. In *Proceedings 45th Annual ACM Symposium on Theory of Computing STOC*, pages 191–200, 2013.

[47] Yun Kuen Cheung, Richard Cole, and Ashish Rastogi. Tatonnement in ongoing markets of complementary goods. In *ACM Conference on Electronic Commerce, EC*, pages 337–354, 2012.

[48] Constantinos Daskalakis, Alan Deckelbaum, and Anthony Kim. Near-optimal no-regret algorithms for zero-sum games. *Games Econom. Behav.*, 92:327–348, 2015.

[49] Alexander Rakhlin and Karthik Sridhanan. Online learning with predictable sequences. In *Proc. 27th Conf. Adv. Neural Information Processing Systems (NIPS)*, pages 3066–3074, 2013.

[50] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert Schapire. Fast convergence of regularized learning in games. In *Proc. 29th Conf. Adv. Neural Information Processing Systems (NIPS)*, pages 2989–2997, 2015.

[51] Bruno Codenotti, Benton McCune, and Kasturi Varadarajan. Market equilibrium via the excess demand function. In *Proc. 37th Symp. Theory of Computing (STOC)*, pages 74–83, 2005.

[52] Richard Cole and Lisa Fleischer. Fast-converging tatonnement algorithms for one-time and ongoing market problems. In *Proc. 40th Symp. Theory of Computing (STOC)*, pages 315–324, 2008.

[53] Richard Cole, Lisa Fleischer, and Ashish Rastogi. Discrete price updates yield fast convergence in ongoing markets with finite warehouses. *CoRR*, abs/1012.2124, 2010. URL `http://arxiv.org/abs/1012.2124`.

[54] Yun Kuen Cheung, Richard Cole, and Ashish Rastogi. Tatonnement in ongoing markets of complementary goods. In *Proc. 13th Conf. Electronic Commerce (EC)*, pages 337–354, 2012.

[55] Yun Kuen Cheung, Richard Cole, and Nikhil Devanur. Tatonnement beyond gross substitutes? Gradient descent to the rescue. In *Proc. 45th Symp. Theory of Computing (STOC)*, pages 191–200, 2013.

[56] Yun Kuen Cheung. *Analyzing Tatonnement Dynamics in Economic Markets*. PhD thesis, Courant Institute of Mathematical Sciences, NYU, 2014.

[57] Guillermo Gallego and Ming Hu. Dynamic Pricing of Perishable Assets Under Competition. *Management Sci.*, 60(5):1241–1259, 2014.

[58] Guillermo Gallego and Ruxian Wang. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Oper. Res.*, 62(2):450–461, 2014.

[59] Ming Chen and Zhi-Long Chen. Recent Developments in Dynamic Pricing Research: Multiple Products, Competition, and Limited Demand Information. *Production and Operations Management*, 24(5):704–731, 2015.

[60] Krishnamurthy Dvijotham, Yuval Rabani, and Leonard Schulman. Convergence of incentive-driven dynamics in fisher markets. In *Proc. 28th Symp. Discrete Algorithms (SODA)*, pages 554–567, 2017.

[61] Jacob Ramskov and Jesper Munksgaard. Elasticities – a theoretical introduction. *Balmorel Project*, 2001.

[62] Carl Meyer. *Matrix analysis and applied linear algebra*. SIAM, 2000.

[63] John Hunter and Bruno Nachtergaele. *Applied analysis*. World Scientific, 2001.

[64] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[65] Bora Keskin and Assaf Zeevi. Dynamic Pricing with an Unknown Demand Model: Asymptotically Optimal Semi-Myopic Policies. *Operations Research*, 62(5):1142–1167, 2014.

[66] Panayotis Mertikopoulos. Learning in concave games with imperfect information. 2016. URL `http://arxiv.org/abs/1608.07310`.

[67] Alexandre X Carvalho and Martin L Puterman. Learning and pricing in an internet environment with binomial demands. *Journal of Revenue and Pricing Management*, 3(4):320–336, 2005.

[68] Arnoud V den Boer and Bert Zwart. Simultaneously learning and optimizing using controlled variance pricing. *Management science*, 60(3):770–783, 2013.

[69] Josef Broder and Paat Rusmevichientong. Dynamic pricing under a general parametric choice model. *Oper. Res.*, 60(4):965–980, July 2012. ISSN 0030-364X. doi: 10.1287/opre.1120.1057. URL `http://dx.doi.org/10.1287/opre.1120.1057`.

[70] Omar Besbes and Assaf Zeevi. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research*, 57(6): 1407–1420, 2009.

[71] Robert Kleinberg and Tom Leighton. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '03, pages 594–. IEEE Computer Society, 2003. ISBN 0-7695-2040-5.

[72] Omar Besbes and Assaf Zeevi. On the minimax complexity of pricing in a changing environment. *Operations research*, 59(1):66–79, 2011.

[73] Guillermo Gallego and Ming Hu. Dynamic Pricing of Perishable Assets Under Competition. *Management Science*, 60(5):1241–1259, 2014. URL `http://pubsonline.informs.orghttp://dx.doi.org/10.1287/mnsc.2013.1821`.

[74] Ali K Parlaktürk. The value of product variety when selling to strategic consumers. *Manufacturing & Service Operations Management*, 14(3):371–385, 2012.

[75] Guillermo Gallego and Ruxian Wang. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Operations Research*, 62(2):450–461, 2014.

[76] Ming Chen and Zhi-Long Chen. Recent Developments in Dynamic Pricing Research: Multiple Products, Competition, and Limited Demand Information. *Production and Operations Management*, 24(5):704–731, 2015. URL `http://doi.wiley.com/10.1111/poms.12295`.

[77] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pages 928–935. AAAI Press, 2003. ISBN 1-57735-189-4. URL `http://dl.acm.org/citation.cfm?id=3041838.3041955`.

[78] Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In Shai Shalev-Shwartz and Ingo Steinwart, editors, *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 of *Proceedings of Machine Learning Research*, pages 993–1019, Princeton, NJ, USA, 12–14 Jun 2013. PMLR.

[79] Alexander Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, NIPS'13, pages 3066–3074, USA, 2013. Curran Associates Inc. URL `http://dl.acm.org/citation.cfm?id=2999792.2999954`.

[80] K. Arrow, H. Block, and L. Hurwicz. On the stability of the competitive equilibrium: II. *Econometrica*, 27(1):82–109, 1959.

[81] Bruno Codenotti, Benton McCune, and Kasturi R. Varadarajan. Market equilibrium via the excess demand function. In *Proc. 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 74–83, 2005.

[82] Richard Cole and Lisa Fleischer. Fast-converging tatonnement algorithms for one-time and ongoing market problems. In *Proc. 40th Annual ACM Symposium on Theory of Computing STOC*, pages 315–324, 2008.

[83] Richard Cole, Lisa Fleischer, and Ashish Rastogi. Discrete price updates yield fast convergence in ongoing markets with finite warehouses. *CoRR*, abs/1012.2124, 2010. URL `http://arxiv.org/abs/1012.2124`.

[84] Yun Kuen Cheung and Richard Cole. Amortized analysis on asynchronous gradient descent. *CoRR*, abs/1412.0159, 2014.

[85] Yun Kuen Cheung and Richard Cole. A unified approach to analyzing asynchronous coordinate descent and tatonnement. *CoRR*, abs/1612.09171, 2016.

[86] Thodoris Lykouris, Vasilis Syrgkanis, and Éva Tardos. Learning and efficiency in games with dynamic population. In *Proc. 27th Symp. Discrete Algorithms (SODA)*, pages 120–129, 2016.

[87] Dylan Foster, Zhiyuan Li, Thodoris Lykouris, Karthik Sridharan, and Éva Tardos. Learning in games: Robustness of fast convergence. In *Proc. 30th Conf. Adv. Neural Information Processing Systems (NIPS)*, pages 4727–4735, 2016.

[88] Yun Kuen Cheung. *Analyzing Tatonnement Dynamics in Economic Markets*. PhD thesis, Courant Institute of Mathematical Sciences, NYU, 2014.

[89] Raghu Subramanian and Isaac D Scherson. An analysis of diffusive load-balancing. In *Proc. 6th annual ACM symposium on Parallel algorithms and architectures*, pages 220–225. ACM, 1994.

[90] Noam Nisan, Éva Tardos, Tim Roughgarden, and Vijay Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.