Online Algorithms for Markets

Philip Lazos

Somerville College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2018

This thesis is dedicated to my parents,

for their unconditional love and support.

Acknowledgements

The dissertation is the culmination of great personal effort, as well as systematic support and assistance from teachers, relatives and friends, who I would like to personally thank. First of all, I would like to thank my supervisor, Professor Elias Koutsoupias, for the unwavering support he provided throughout my research. Even if I wasn't always the easiest student, Elias remained trusting and optimistic. In his unique way, he showed me that research should be a deeply engaging way of life, having vision, integrity and the courage to pursue beautiful, important problems, no matter how far away the payoff may be.

In part, much of my journey into Computer Science, from writing the first line of code to the last sentences in my thesis have been influenced by my childhood friend Alex Kavvos. I owe a lot to my collaborators Yiannis Giannakopoulos, who helped initiate me into the world of academia and develop my writing style in hopes of reaching his standards, and Aris Filos-Ratsikas for developing my sense of the academic community and self confidence in my relationship with research.

A big thank you to all my amazing friends in Greece: Mikelis, Spyros, Christos, Katerina, Vasilis, Thodoris and Aggela as well as my office mates Francisco, Ninad, Christian, Matthias and Alexandros.

Last but not least, I would like to thank my family: Christos, Christina, Orestis and my beloved cat Batis, who sat by my side all those times I worked late through the night.

Finally, I am grateful to the following institutions for their generous financial support during my graduate studies: the Computer Science department of the University of Oxford, the Engineering and Physical Sciences Research Council (EPSRC) and the European Union through the ERC Advanced Grant 321171 (ALGAME).

Abstract

The thesis consists of two parts, both dealing with issues of uncertainty and incentives in markets. In the first part we examine the *online properties* of markets. In most of the Mechanism Design literature, markets are studied under the assumption that all participants are present at the same time and can seamlessly interact with each other. This may not always be the case in practice. We consider markets where buyers and sellers appear in sequence, one after another, without overlapping and it is the duty of an intermediary to coordinate with them. We take the role of that intermediary and our goal is redistribute items among them to maximise certain objectives, namely the profit, social welfare or gain from trade. We focus on *posted price* mechanisms, which are robust and truthful. There are two natural, complementary variants of the order of arrival of the agents. In the first case, an adversary dictates their order, but the intermediary has prior, distributional information about their valuations, similar to the prophet inequality setting. In the second, the adversary selects their valuations but their order is a uniformly random permutation. We obtain asymptotically tight worst-case guarantees for both cases, under a competitive analysis benchmark.

In the second part, we study the strategic implications that arise from adding one extra option to the miners participating in the bitcoin protocol. We propose that when adding a block, miners also have the ability to pay forward an amount to be collected by the first miner who successfully

extends their branch, giving them the power to influence the incentives for mining. We formulate a stochastic game for the study of such incentives and show that with this added option, smaller miners can guarantee that the best response of even substantially more powerful miners is to follow the expected behavior intended by the protocol designer. Moreover, pay-forward can be used to alleviate the predicted instability when block rewards are small compared with respect to transaction fees, by smoothing out the variability of the rewards collected from transaction fees.

Contents

List of Figures

Chapter 1

Introduction

1.1 Games, Markets and Online Algorithms

Consider an intermediary trying to facilitate trade between a buyer and a seller, who has one item available. How could he help them out? Perhaps he could ask each of them independently how much they value the item and ask them to trade at a price which is the average of their two reports. The answer of the seller and buyer (referred to as strategic *agents*), is not immediately obvious. There is a game going on here: each agent knows how much he values the item, but he might get a better deal by not telling the truth. For example, the seller might value the item at £100 and the buyer at £130. This is their *valuation*, which (although measured in pounds) quantifies how happy they are when having the item. If they are both honest the seller will get £115 for his item. *However*, by reporting £129 he would still sell the item and get £129.5 instead. Therefore, the *strategy* of each agent in this game is what valuation to report and their *objective* is to maximise their *utility*, which is their valuation (if they have the item) plus the amount of money they might have made. Even for simplistic games like this one, this strategic behaviour can lead to very complicated, often inefficient results. The intermediary should implement a protocol that is very

easy to follow. Ideally, both agents should always maximise their gain by truthfully reporting their valuation, without having to put any more thought into their reports: honest participation should be a *dominant strategy*. To achieve this, the intermediary could *estimate* the seller's value (without asking him) and propose that both agents trade at that price. Clearly, none of them has any incentive to misreport, since they would only be missing on a beneficial trade. The design of such protocols of strategic interaction (especially with money) is the field of *Mechanism Design* and this specific setting with a single seller-buyer pair is called *Bilateral Trade* [59].

The intermediary of course has an objective of his own. In this case, it usually is either to maximise his revenue or the *social welfare* which is the overall happiness of all parties involved (including his own). To design better mechanisms, he has two extra tools at his disposal. First of all, to make this estimation of the seller's value, we assume that there exist some *prior* information about the agents' valuations, appearing in the form of a probability distribution. Second, using the *Revelation Principle* we can show that even though the range of possible mechanisms is vast, the truthful ones we are interested in always have the following form: both agents secretly and at the same time report their valuations and then the intermediary decides on the outcome, following an agreed upon procedure. To see this, notice how some more complicated *truthful* interactive mechanism could always take that form, with the intermediary pretending to play like the agents after they tell him their valuations. Most of the mechanisms we will consider are *posted price*, like the one previously described and easily satisfy the truthfulness requirement. Finally, the mechanism can be *random*: the same reports could sometimes lead to different outcomes, but truth telling should still be the best strategy *in expectation.*

In a real market however, there is rarely just one buyer and one seller. Usually we expect a large number of both, with many of them entering and leaving the market each day, rather than completing every trade at once. This of course makes the

intermediary's job significantly more complicated. Assuming he has a large warehouse and interacts with each agent independently, on separate occasions, he not only has to decide on which price to set, but also which items to buy and how many to keep in store. An expensive item would be too hard to sell later. Also, he could sell at a low price to an indifferent buyer, or wait until someone who really wants the item appears. To rigorously deal with this *uncertainty*, we analyse the mechanism under two assumptions. The agents will appear *online*: that is, one after another and in an unknown sequence. Also, they will appear in the worst order possible. An omniscient adversary will select the ordering in advance, to make our mechanism perform as badly as possible compared to his. This is known as *Competitive Analysis.*

1.2 Outline of the Thesis

The thesis is split into two parts, with Part I focusing on online markets and Part II on blockchain mining games. The second part is also game theoretic, but due to the different setting it is mostly self contained. In Chapter 2, we rigorously define the ideas highlighted in Section 1.1, along with the main results in Bilateral Trade.

In Chapters 3 and 4 we formulate the two variants of the online markets studied in this work. In Chapter 3 we find the first of the two, where the ordering of agents is modelled after the *prophet inequality*. We assume that there is prior information about the agents and their order is determined by an adversary. Every item is identical, but of course agents can have different valuations, sampled from known distributions F_S for the sellers and F_B for the buyers. We analyse two objectives: maximising the profit and the social welfare. We provide a cascade of increasingly easier market settings: starting from the completely unrestricted (where we could potentially have infinitely more sellers than buyers), to the stock limited, where the intermediary can only hold up to K items at once and finally the balanced, where the ratio of buyers

to sellers is roughly similar throughout the sequence.

In Chapter 4, we model the complement of Chapter 3, based on the *secretary problem*. The adversary selects the valuation of each agent, but their order is a uniformly random permutation. We find mechanisms that are asymptotically optimal for the social welfare or provide a constant approximation to the *gain-from-trade*, which is *only* the *increase* in welfare after the agents interacted with the mechanism. In terms of approximation it is harder to maximise than welfare and easier than revenue.

Finally, in Chapter 5 we study the strategic implications of allowing nodes to directly influence where others should mine by *paying* anyone who mines after them. This alteration increases the stability of the bitcoin protocol in two ways: by ensuring that even substantially more power miners follow the behaviour prescribed by the protocol designer and by limiting the instability that could arise when only transaction rewards remain.

1.3 List of Papers

This thesis is comprised of the following papers:

[1] Yiannis Giannakopoulos, Elias Koutsoupias and Philip Lazos

Online Market Intermediation

In *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, July 2017.

[2] Elias Koutsoupias and Philip Lazos

Online Trading as a Secretary Problem

In *Proceedings of the 11th Symposium of Algorithmic Game Theory (SAGT 2018)*, September 2018.

[3] Elias Koutsoupias, Philip Lazos, Foluso Ogunlana and Paolo Serafino

Blockchain Mining Games with Pay-Forward

Working Paper.

The author also has other related work which is not part of this thesis.

[4] Christian Coester, Elias Koutsoupias and Philip Lazos

The Infinite Server Problem

In *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, July 2017.

[5] Aris Filos-Ratsikas, Yiannis Giannakopoulos and Philip Lazos

The Pareto Frontier of Inefficiency in Mechanism Design

Working Paper.

[6] Matthias Gerstgrasser, Paul Goldberg, Bart de Keijzer, Philip Lazos and Alexander Skopalik

Multi-Unit Bilateral Trade

Working Paper.

Part I

Market Intermediation

Chapter 2

The Bilateral Trade Setting

2.1 Introduction

In this chapter we present an overview of the setting, key techniques and intuition required for Chapters 3 and 4, elaborating on the high level notions presented in Chapter 1. The overarching field is *Mechanism Design*, viewed through the lens of the very specific setting we extend in the following sections, namely that of *Bilateral Trade*. Whenever possible, we avoid being too abstract in our definitions and focus on how the incentives of the players and useful properties of our solution concepts apply here. Even though this chapter is enough to follow the rest of the thesis, it is by no means a thorough (or fair) introduction to Mechanism Design or Game Theory. For this, we point the reader to the standard textbooks [53, 62].

2.2 Bilateral Trade

The bilateral trade setting is a fundamental economic scenario introduced by Myerson and Satterthwaite in [59], with one buyer and one seller. The seller has one item with some intrinsic value to him, but ideally he would want to sell it to the buyer and maximise his revenue. To facilitate trade between them, we take the role of an

intermediary and design an auction mechanism to coordinate with them.

The interaction with the seller and buyer (we refer to them as *agents*) is very simple: the mechanism asks them to report the value they ascribe to the item, and then tells them if they have to trade and how much they need to pay to each other, or to the intermediary. Both agents know *exactly* how the mechanism works. Formally, let $v_S, v_B \in [a, b]$ be the *true* values of the seller and buyer respectively, where $0 \leq a \leq b$. These values are *only* known by their respective agent. The mechanism has to elicit them, by providing a framework of interaction guaranteeing that honest participation will be rewarded. The *deterministic* mechanism $\mathcal{M}$ can be described as a pair of two functions: the *allocation* function $x : [a, b]^2 \to 0, 1$ determine if the *buyer* receives the item and the payment functions $p^S, p^B : [a, b] \to \mathbb{R}$, where p^S is the payment from the intermediary *to* the seller and p^B *from* the buyer to the intermediary. Note that both these payments could be negative, indicating money flowing in the opposite direction.

The agents of course, being strategic, will not necessarily report their real valuations v_S, v_B to $\mathcal{M}$. They will try to maximise their *utility*, which is their overall gain: the value of the item (if they got it) minus the price paid for it. If they report $\hat{v}_S, \hat{v}_B$ to the mechanism, their utility is defined as:

$$u^B(\hat{v}_S, \hat{v}_B) = v_B \cdot x(\hat{v}_S, \hat{v}_B) - p^B(\hat{v}_S, \hat{v}_B) \tag{2.1}$$

for the buyer and

$$u^S(\hat{v}_S, \hat{v}_B) = v_S \cdot (1 - x(\hat{v}_S, \hat{v}_B)) + p^S(\hat{v}_S, \hat{v}_B) \tag{2.2}$$

for the seller. In the rest of the chapter it is always implicit that v_S, v_B are the true valuations and $\hat{v}_S, \hat{v}_B$ the reported valuations. Moreover, functions (such as u^S, p^B) are assumed to take as arguments the valuations reported to the mechanism. That's why we need to design mechanisms that are *incentive compatible (IC)*, where their

utility is be maximised by truthfully reporting their values v_S, v_B.

Definition 2.1. *A mechanism $\mathcal{M}$ is incentive compatible if for all reported valuations $\hat{v}_S, \hat{v}_B \in [a, b]$:*

$$u^S(v_S, \hat{v}_B) \geq u^S(\hat{v_S}, \hat{v}_B) \text{ and } u^B(\hat{v}_S, v_B) \geq u^B(\hat{v_S}, \hat{v}_B). \tag{2.3}$$

A mechanism with this property is called *dominant strategy* incentive compatible, since it is *always* a good strategy to be truthful, no matter what the other agent might report.

Finally, the mechanism needs to entice the players into participating by ensuring that they benefit, a property called *individual rationality (IR)* or *voluntary participation*.

Definition 2.2. *A mechanism $\mathcal{M}$ is individually rational if for all reported valuations $\hat{v}_S, \hat{v}_B \in [a, b]$:*

$$u^S(v_S, \hat{v}_B) \geq 0 \text{ and } u^B(\hat{v}_S, v_B) \geq 0. \tag{2.4}$$

The mechanism only needs to guarantee that the agents will benefit if in turn they participate truthfully.

There is a variety of objectives that the intermediary might be interested in, but in the Bilateral Trade literature the primary is maximising the social welfare, which is the sum of *all* the utilities (including the intermediary's). Since money is worth the same amount to all parties involved, the social welfare only depends on which agents gets the item.

Definition 2.3. *The social welfare of mechanism $\mathcal{M}$ for reported valuations $\hat{v}_S, \hat{v}_B$ is:*

$$SW(\hat{v}_S, \hat{v}_B) = v_S \cdot (1 - x(\hat{v}_S, \hat{v}_B)) + v_B \cdot x(\hat{v}_S, \hat{v}_B). \tag{2.5}$$

Often, we compare this quantity with the *optimal* social welfare $\max\{v_S, v_B\}$. A

mechanism which achieves the optimal social welfare (for every v_S, v_B) is called *efficient*. It is easy to see that a mechanism is efficient if and only if a trade is performed whenever $v_B > v_S$. More generally, a mechanism provides an α approximation for the welfare objective if

$$\frac{\max\{v_S, v_B\}}{SW(v_S, v_B)} \leq \alpha.$$

The natural question therefore is this: does there exist an individually rational, incentive compatible and efficient mechanism? Fortunately, the answer is positive and the mechanism achieving this is quite simple. It is essentially a version of the VCG mechanism, named after Vickrey [75], Clarke [22] and Groves [40] who generalised this method, adapted for this setting. In particular, the mechanism always gives the item to the msst valuable agent.

$$x_{VCG}(\hat{v}_S, \hat{v}_B) = \begin{cases} 1 & \text{if } \hat{v}_B \geq \hat{v}_S \\ 0 & \text{otherwise} \end{cases}.$$

The payments used to achieve this are:

$$p^S_{VCG}(\hat{v}_S, \hat{v}_B) = \begin{cases} \hat{v}_B & \text{if } \hat{v}_B \geq \hat{v}_S \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad p^B_{VCG}(\hat{v}_S, \hat{v}_B) = \begin{cases} \hat{v}_S & \text{if } \hat{v}_B \geq \hat{v}_S \\ 0 & \text{otherwise} \end{cases}.$$

In essence, each player has to pay the decrease in utility he incurs to the other. If his participation leads to increase in utility, the mechanism actually *pays him* instead. It is easy to verify that this mechanism is indeed truthful and individually rational, as is shown in [75, 22, 40] for more general settings.

Proposition 1. *The VCG mechanism is incentive compatible, individually rational and efficient.*

Proof. We will show these properties for the buyer. For the buyer's utility we have:

$$u^B(\hat{v}_S, \hat{v}_B) = v_B \cdot x_{VCG}(\hat{v}_S, \hat{v}_B) - p^B_{VCG}(\hat{v}_S, \hat{v}_B) \tag{2.6}$$

$$= \begin{cases} v_B - \hat{v}_S & \text{if } \hat{v}_B \geq \hat{v}_S \\ 0 & \text{otherwise} \end{cases}. \tag{2.7}$$

Clearly then we have $u^B(\hat{v}_S, v_B) \geq 0$, satisfying individual rationality.

Fixing $\hat{v}_S$, there are two possible outcomes depending on $\hat{v}_B$, leading to utility $v_B - \hat{v}_S$ or 0 for the buyer. Note that reporting v_B always leads to the maximum of those and since the choice does not depend on $\hat{v}_S$ the mechanism is incentive compatible. The analysis from the seller's perspective is identical.

By incentive compatibility, the agents will report their true valuations to the mechanism. By the Definition 2.3 the social welfare will be:

$$\begin{aligned} SW(v_S, v_B) &= v_S \cdot (1 - x(\hat{v}_S, \hat{v}_B)) + v_B \cdot x(\hat{v}_S, \hat{v}_B) \\ &= \begin{cases} v_B & \text{if } \hat{v}_B \geq \hat{v}_S \\ v_S & \text{otherwise} \end{cases} \\ &= \max\{v_S, v_B\}. \end{aligned}$$

□

However, there is an issue. By incentive compatibility, the mechanism would increase the social welfare by $v_B - v_S$ (provided this quantity is positive). But, the net sum of payments *given* to the agents would also be $v_B - v_S$! The trade would occur of course, but the item itself is only tangential: it has been used to guarantee

truthfulness (as it is not really possible to just pay $v_B - v_S$ to the buyer). Also, this mechanism is not individually rational from the intermediary's perspective since he *spends* his own money to help the market.

To this end, we need to define the notion of *budget balance (BB)*, requiring that the mechanism always maintains a positive balance.

Definition 2.4. *A mechanism $\mathcal{M}$ is* weakly *budget balanced if for all reported valuations $\hat{v}_S, \hat{v}_B \in [a, b]$:*

$$p^B(\hat{v}_S, \hat{v}_B) - p^S(\hat{v}_S, \hat{v}_B) \geq 0 \tag{2.8}$$

Note that the mechanism is actually allowed to make a profit. Curiously, this often makes for *more* efficient mechanisms, as redistributing the excess money back to the agents often clashes with the incentive compatibility requirement. Substituting the inequality with equality would give us *strong* budget balance, where all money transfers have to be strictly between the two agents.

To design mechanisms that satisfy incentive compatibility, individual rationality, weak budget balance and are also efficient, it would be very helpful to have some *prior* information about v_S, v_B, so that the mechanism's actions are not based entirely on the reports but can also rely on dependable information. We assume that $v_S \sim F_S$ and $v_B \sim F_B$ and that these distributions are public knowledge. These need not be identical, but for simplicity we will assume that they are independent. Since the agents also know these distributions, we need to highlight a subtle property of the previous definitions of IC, IR and BB. The way we have defined them, they should *always* hold for any pair of v_S, v_B. In practice, a more relaxed version is often employed, where we only need that they hold for every agent in expectation over the other agents' distribution.

Instead of requiring truthful reporting to maximise the agents' utility for *every*

set of reports, we only need to maximise utility *assuming the other agent will also be truthful* (and essentially treat his report as a random sample of his distribution).

Definition 2.5. *A mechanism $\mathcal{M}$ is Bayesian incentive compatible (BIC) if for any $s : [a, b] \to [a, b]$:*

$$\mathop{\mathbb{E}}_{v_B \sim F_B} \left[u^S(v_S, v_B) \right] \geq \mathop{\mathbb{E}}_{v_B \sim F_B} \left[u^S(s(v_S), v_B) \right]$$

and

$$\mathop{\mathbb{E}}_{v_S \sim F_S} \left[u^B(v_S, v_B) \right] \geq \mathop{\mathbb{E}}_{v_S \sim F_S} \left[u^B(v_S, s(v_B)) \right].$$

The function s is the strategy employed, which could depend on the players valuation. Note that any dominant strategy incentive compatible mechanism is also Bayesian incentive compatible, but the opposite is not true.

The mechanisms considered so far have been deterministic. However, *randomised* mechanisms are often easier to design and achieve better performance guarantees. Intuitively, they allow us to *evenly split the item in expectation*, obtaining a fraction of both the agents' valuations, even though the item itself is indivisible. A randomised mechanism obtains the agents reports and then runs a randomly selected deterministic mechanism. The notion of incentive compatibility also applies to randomised mechanisms. If the support of possible mechanisms only contain incentive compatible ones, then the overall mechanism is *universally* incentive compatible. However, if only the game induced over the mechanism's expectation is, then it is incentive compatible *in expectation*[1].

Similarly, we can weaken the budget balance condition to hold only in expectation.

$$\mathop{\mathbb{E}}_{v_S \sim F_S, v_B \sim F_B} \left[\mathop{\mathbb{E}}_{\mathcal{M}} \left[p_B(\hat{v}_S, \hat{v}_B) - p_S(\hat{v}_S, \hat{v}_B) \right] \right] \geq 0,$$

[1]This definition could be made more general, by allowing the mechanism to signal some of its random coin tosses to the agents. Since we're mostly interested in universal dominant strategy incentive compatible, this is sufficient.

where the expectation is over first over v_S, v_B and *then* over the mechanism. This weaker version is referred to as *ex-ante* budget balance and the one from Definition 2.4 as *ex-post*. The social welfare can be similarly defined in expectation.

Unfortunately, one of the key findings of Myerson and Satterthwaite in [59] was that despite relaxing the notions of incentive compatibility, allowing randomisation and giving the mechanism access to Bayesian priors, there still is no efficient mechanism satisfying all the desired properties. To simplify the presentation, we prove a slightly weaker version of the theorem. The full version holds for Bayesian incentive compatibility as well.

Theorem 2.1. *There exists no (even randomised) mechanism that is simultaneously efficient, individually rational, universally dominant strategy incentive compatible and ex-ante budget balanced, assuming f_S and f_B are positive over the support $[a, b]$.*

Proof. Fix any mechanism $\mathcal{M}$ and let $\mathcal{M}'$ be any mechanism selected by $\mathcal{M}$ with positive probability. By universal dominant strategy IC, we only need to show the result for $\mathcal{M}'$, upon which the rest of the proof is based.

By efficiency we need that the trade occurs if and only if $v_B > v_S$. By incentive compatibility 2.3 for the buyer we have that his reported valuation must satisfy $\hat{v}_B \in \operatorname{argmax}_{\hat{v}_B} v_B \cdot x(\hat{v}_S, \hat{v}_B) - p^B(\hat{v}_S, \hat{v}_B)$. Since this holds for any $\hat{v}_S$, it should also hold in expectation. We define $x^B(\hat{v}_B) = \mathbb{E}\left[x(\hat{v}_S, \hat{v}_B)\right] = F_S(\hat{v}_B)$ and $p^B(\hat{v}_B) = \mathbb{E}\left[p^B(\hat{v}_S, \hat{v}_B)\right]$. Under some mild conditions on the mechanism, this maximisation requires that

$$v_B \frac{dx^B(\hat{v}_B)}{d\hat{v}_B} = \frac{dp^B(\hat{v}_B)}{d\hat{v}_B} \Rightarrow \hat{v}_B \frac{dx^B(\hat{v}_B)}{d\hat{v}_B} = \frac{dp^B(\hat{v}_B)}{d\hat{v}_B}$$

By integrating from a to v_B (knowing that $v_B \in [a, b]$), we get that the required

payment is:

$$p^B(v_B) = p^B(a) + \int_a^{v_B} \hat{v_B} \frac{dx^B(\hat{v_B})}{d\hat{v_B}} d\hat{v_B} \tag{2.9}$$

$$\leq \int_a^{v_B} \hat{v_B} \frac{dx^B(\hat{v_B})}{d\hat{v_B}} d\hat{v_B}. \tag{2.10}$$

Because the mechanism is efficient, the buyer would never obtain the item if his valuation is a, as $\hat{v}_S \geq a$. By individual rationality his payment needs to be $p^B(a) \leq 0$, leading to 2.10. Taking into account that $x^B(\hat{v_B}) = F_S(\hat{v}_B)$, we obtain:

$$p^B(v_B) = \int_a^{v_B} \hat{v_B} f_s(\hat{v_B}) d\hat{v_B} \tag{2.11}$$

Similarly for the seller, starting from

$$v_S \frac{d\left(1 - x^S(\hat{v}_S)\right)}{d\hat{v}_S} = -\frac{dp^S(\hat{v}_S)}{d\hat{v}_S},$$

we obtain:

$$p^S(v_S) \geq \int_{v_S}^b \hat{v_S} f^B(\hat{v_S}) d\hat{v_S}. \tag{2.12}$$

For the expected payment by mechanism $\mathcal{M}'$, we subtract 2.12 from 2.11 and take expectations to obtain:

$$\mathbb{E}\left[p^B(v_S, v_B) - p^S(v_S, v_B)\right] = \mathbb{E}\left[p^B(v_B) - p^S(v_S)\right] \tag{2.13}$$

$$= \int_a^b p^B(v_B) f_B(v_B) dv_B - \int_a^b p^S(v_S) f_S(v_S) dv_S \tag{2.14}$$

$$\leq \int_a^b \left(\int_a^{v_B} \hat{v_B} f_S(\hat{v_B}) d\hat{v_B} \right) f_B(v_B) dv_B \tag{2.15}$$

$$- \int_a^b \left(\int_{v_S}^b \hat{v_S} f_B(\hat{v_S}) d\hat{v_S} \right) f_S(v_S) dv_S \tag{2.16}$$

$$\leq \int_{v_B > v_S} (v_S - v_B) f_S(v_S) f_B(v_B) dv_B dv_S, \tag{2.17}$$

where the last inequality follows since the supports of F_S and F_B have a nonempty intersection, by the assumption that f_S, f_B are positive over $[a, b]$.

This holds for every mechanism $\mathcal{M}'$ that could be selected by $\mathcal{M}$. Therefore, by taking expectations over the $\mathcal{M}$ the ex-ante budget balance constraint is contradicted.

□

Note that this result *only* holds if F_S and F_B are independent (consider for example $v_S \sim \mathcal{U}[0, 1]$ and $v_B = \sqrt{v_S}$).

It seems then that if IC, IR and efficiency are needed no mechanism can be budget balanced. Not only that, but by 2.17, it seems that the mechanism pays at least $v_B - v_S$ every time $v_B > v_S$. The VCG mechanism described previously, which we considered quite generous, actually pays the *least* amount needed to satisfy these properties.

Since the IC, IR and BB properties are necessary from an economic point of view, this gives us two options: we can find mechanisms that get as close to the optimal welfare as possible, or slightly change the setting. Both of these alternatives have been thoroughly investigated.

2.2.1 Markets with Many Agents

Although this impossibility holds for the simple setting we considered, in reality we expect that multiple agents will be involved in a market. At first glance this seems to make the issue more complicated. After all, how would we be able to ensure that n buyers and n sellers are truthful? In a seminal paper , McAfee [56] showed that this inefficiency actually *vanishes* as the number of agents goes to infinity.

We introduce some more convenient notation for multiple buyers and sellers. Let $s^1 \leq s^2 \leq \ldots \leq s^n$ and $b^1 \geq b^2 \geq \ldots \geq b^n$ be the valuations of the sellers and buyers in *increasing* and *decreasing* order respectively. Let k be the largest index such that

$b_k \geq s_k$. Even though there are multiple items available, agents are *unit demand* and only are interested in a single copy. Let $\hat{\mathbf{s}}$ be the reported valuations of all agents. As before, the utility of some buyer i is

$$b^i \cdot x_i(\hat{\mathbf{s}}) - p_i^B(\hat{\mathbf{s}}),$$

where $x_i \in \{0, 1\}$ is the number of items he obtained and p_i the price paid, as defined as before. Notice that even though the number of items is n, each agent is unit demand and a sensible mechanism would give him at most one item. Of course, x^i, p^i need not be the same for all agents. Similarly, the utility of seller j is $s^j \cdot \min\{1, (1 - x_j(\hat{\mathbf{s}}))\} + p_j^S(\hat{\mathbf{s}})$.

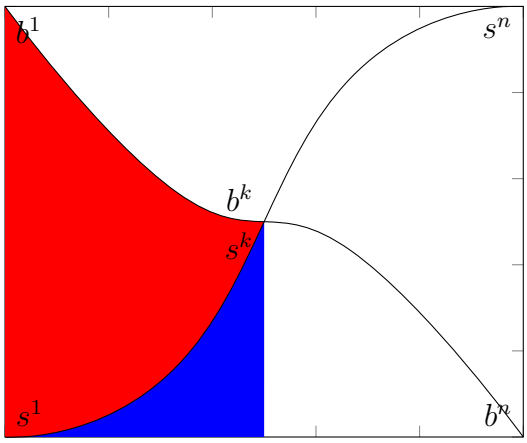

Figure 2.1: The graph represents a visualisation of the ordered agents. The blue area corresponds to the welfare of the k lowest valued sellers and the red area to the increased welfare obtained by trading the item to the k highest valued buyers. The index k is at the intersection.

This index k is very important, as it is the number of trades performed to achieve the optimal welfare. The trade-reduction mechanism (which is a close relative to the mechanism in [56]) works as follows:

- Order the agents by $\hat{b}^1 \leq \ldots \leq \hat{b}^n$ and $\hat{s}^1 \geq \ldots \geq \hat{s}^n$ according to their reported valuations.

- Calculate k.
- Set price s^k and buy from the first $k-1$ sellers at price s^k.
- Sell these items to the $k-1$ first buyers at price b^k.

The IC and IR guarantees are easily satisfied: the first $k-1$ buyers and sellers cannot directly control the price set to them and changing their valuation can only result in them *not* participating in a desirable exchange. The k-th buyer and seller have no incentive to misreport. If they move up the first $k-1$ positions, they would be participating in a trade at a price higher than their value leading to negative utility.

Proposition 2. *The trade reduction mechanism is dominant strategy incentive compatible, individually rational, ex-post weakly budget balanced and achieves a* $1-1/k$ *fraction of the optimal welfare, where* k *is the optimal number of trades.*

The mechanism is weakly budget balanced and always has nonnegative profit equal to $(k-1)\cdot(b^k-s^k)$. The optimal welfare is generated by the top k trades. The trade reduction mechanism always misses one, but only the least valuable. Therefore the generated welfare is at least $1-1/k$ of the optimal. Note that this mechanism did not require any prior knowledge of F_S or F_B. It is however, completely useless for the bilateral trade setting, where $k=1$, and has poor guarantees even if the agents' valuations are assumed to be independent and individually distributed.

McAffee's mechanism is slightly different: the price set is $p=(s^{k+1}+b^{k+1})/2$. If $b_k \geq p \geq s_k$ then all k trades are performed, otherwise it continues as the trade reduction mechanism. The advantage here is that a $1-1/n$ fraction of the optimal welfare is guaranteed, provided that all agent valuations are bounded above zero. Also, assuming that s^i, b^j are identically and independently distributed for all i,j, the approximation ratio is $1-o(1)$ as $n\to\infty$. However, the rate at which it reaches 1 depends on the distributions. The first mechanism to *always* achieve a guaranteed

approximation ratio was by Colini-Baldeschi et al. in [23]. Their mechanism is 16-approximate, strongly budget balanced and works for different, independent valuation distributions as well.

2.2.1.1 Richer Valuation Classes

In markets where multiple agents are concerned, it is natural to assume that they would be interested in more than one item at once. So far, we have only considered *unit demand* agents in single parameter environments, where their valuation could be adequately described by a positive real number v, multiplied by a binary random variable depending on the outcome. We need to extend this to a valuation *function*, assigning value to every possible bundle of items.

Let S be the set of all items available and assume without loss of generality that they are all unique. Then, a valuation is any function $v : 2^S \to \mathbb{R}_{\geq 0}$. For example, the unit demand valuation would be $v(S) = \max_{t \in S} v(\{t\})$, modelling a situation where the agent is only interested in one copy and can dispose the rest for free. A very common assumption in economics, is that of diminishing returns and substitutes. In a nutshell, the more items someone has, the less valuable the rest of the items become.

Definition 2.6. *A valuation function v is called* subadditive *if for any $S_1, S_2 \subset S$:*

$$v(S_1 \cup S_2) \leq v(S_1) + v(S_2).$$

Furthermore, it is called submodular *if for every $t \in S$ and $S_1 \subset S_2 \subset S$:*

$$v(S_1 \cup \{t\}) - v(S_1) \geq v(S_2 \cup \{t\}) - v(S_2).$$

It is easy to see that any submodular valuation is also subadditive, but the opposite is not true. For both cases, Colini-Baldeschi et al. proved a 4-approximate algorithm

in [25], extending their results from [23]. Specifically for the case of Bilateral Trade where the seller holds n identical items and both agents have submodular valuations, we have found a $1 - 1/e$ approximate randomised mechanism in joint work with Matthias Gerstgrasser, Paul Goldberg, Bart de Keijzer, and Alexander Skopalik.

2.2.2 Welfare Guarantees for Bilateral Trade

Even in the Bilateral Setting with only one item, we can guarantee some fraction of the welfare. The following mechanism by Blumrosen and Dobzinksi [14] shares key features with our mechanisms in Chapter 3. The mechanism is very simple: it computes a price μ_S such that $F_S(\mu_S) = 1/2$, in other words the median of the seller's distributions. It then *posts* this price to both agents, asking them if they want to exchange the item with each other for μ_S. This leads to a very robust implementation of the IR, IC and strong BB properties and it's a technique used with great success in many extensions of the bilateral trade setting. It is perhaps useful to think of the mechanism as asking the agents this binary question, but in fact it can be just as easily modelled with x, p as before. The agents would report a valuation and mechanism would make the (obvious) decision for them.

Proposition 3. *The median mechanism is IR, IC, BB and achieves a* $1/2$ *fraction of the optimal welfare.*

Proof. The mechanism is clearly individually rational and incentive compatible. The price μ does not depend on the agents' reports and they can unilaterally cancel the trade if they want, leaving with the same utility the started with. From the seller's perspective, if $v_S > \mu$ no trade will occur. Reporting $\hat{v}_S > \mu$ will not change this, while $\hat{v}_S < \mu$ could potentially lead to selling the item for a low price. Similarly, if $v_S \leq \mu$, reporting any $\hat{v}_S < \mu$ will have no effect and $\hat{v}_S > \mu$ would stop a beneficial trade. The analysis for the buyer is almost identical.

To prove its approximation ratio, it is enough to show that it holds for any fixed value of v_B since the trade price only depends on F_S. The expected optimal welfare is at most $\Pr[v_B \geq v_S]\, v_B + \Pr[v_S > v_B]\, \mathbb{E}[v_S \mid v_S > v_B]$.

There are two cases:

- $v_B \leq \mu_S$. The expected welfare of the mechanism is:

$$\begin{aligned}
&\Pr[v_S \leq \mu_S]\, v_B + \Pr[v_S > \mu_S]\, \mathbb{E}[v_S \mid v_S > \mu_S] \\
&\qquad = \Pr[v_S \leq \mu_S]\, v_B + \Pr[v_S \geq v_B]\, \mathbb{E}[v_S \mid v_S \geq v_B] \\
&\qquad\quad + \Pr[v_B > v_S > \mu_S]\, \mathbb{E}[v_S \mid v_B > v_S > \mu_S] \\
&\qquad \geq \frac{v_B}{2} + \Pr[v_S \geq v_B]\, \mathbb{E}[v_S \mid v_S \geq v_B],
\end{aligned}$$

  which is at least half of the optimal.

- $v_B \geq \mu_S$. In this case no trade happens and the mechanism has welfare $\mathbb{E}[v_S]$. Note that $\Pr[v_S > v_B]\, \mathbb{E}[v_S \mid v_S > v_B] \geq \mu_S/2$. Therefore:

$$\begin{aligned}
&\frac{\mathbb{E}[v_S]}{\Pr[v_B \geq v_S]\, v_B + \Pr[v_S > v_B]\, \mathbb{E}[v_S \mid v_S > v_B]} \\
&\qquad \geq \frac{\Pr[v_S > v_B]\, \mathbb{E}[v_S \mid v_S > v_B]}{\mu_S/2 + \Pr[v_S > v_B]\, \mathbb{E}[v_S \mid v_S > v_B]} \\
&\qquad \geq \frac{\mu_S/2}{\mu_S/2 + \mu_S/2} = \frac{1}{2}.
\end{aligned}$$

□

The authors also have a $1 - 1/e \approx 0.632$ approximate *randomised* mechanism, which can select from a larger support of prices instead of always choosing the median. This is the best known bound for the welfare. Colini-Baldeschi et al. have the tightest upper bound which is 0.749, leaving a relatively small gap.

2.2.3 Gain From Trade

As far as objectives go, the welfare is somewhat permissive for this setting. Since the seller enters the market with an item (and some associated utility) the welfare of any individually rational mechanism is at least that much. Therefore, there exist markets where the initial welfare by itself could be a decent approximation, or at least a guarantee that things cannot go horribly wrong.

Appropriately, a different objective has received significant attention recently, the gain from trade, defined as the *increase* in welfare from the initial to the final allocation of items. Formally:

$$GFT(\hat{v}_S, \hat{v}_B) = v_B \cdot x(\hat{v}_S, \hat{v}_B) + v_S \cdot (1 - (\hat{v}_S, \hat{v}_B)) - v_S \tag{2.18}$$

Of course, any welfare maximising mechanism would also be optimal for the gain from trade. From an approximation perspective however, the gain from trade is considerably more difficult. Any approximation guarantee for the gain from trade clearly carries over to the welfare as well. The objective was mentioned in [59], where it actually had a more prominent role than the welfare and was used for the impossibility theorem.

McAfee presented a simple IC, IR and strongly budget balanced posted price mechanism in [57]. As in [14] the mechanism proposes a trading price q and the agents trade if they both agree. He showed the following crucial property:

Proposition 4. *Given any price q we have:*

$$\frac{GFT(q)}{OPT_{GFT}} \geq \min\left\{F_S(q), 1 - F_B(q)\right\},$$

where $GFT(q)$ is the expected gain from trade from trading at price q and OPT_{GFT} is the optimal grain from trade.

Let μ_B be the median of F_B. Then, trading at any price q satisfying $\mu_B \geq q \geq \mu_S$ would provide at least $1/2$ of the optimal gain from trade. This assumption is not too strong though: in general, we expect the buyers to be more interested in acquiring some resource than the sellers. This result was significantly improved by Colini-Baldeschi et al. in [24] to $1/r$ and $O(\log(1/r))$, where r is the probability that the buyer's valuation for the item is higher than the seller's.

Blumrosen and Mizrahi [16] devised a $1/e$-approximate, Bayesian incentive compatible mechanism for the gain from trade assuming the buyer's valuation is *monotone hazard rate (MHR)*. We get into much more detail about this class of distributions in Section 3.4, but their key property used here is that their *virtual value* $\phi(x)$ is increasing. For the buyer:

$$\phi_B(x) = x - \frac{1 - F_B(x)}{f_B(x)}.$$

In a seminal result [58], Myerson showed that, in a single item auction with one buyer, asking the price $\phi^{-1}(0)$ maximises the expected profit of the auctioneer, provided that the item has no value for him. Given that, the mechanism proposed is simply:

- The seller sets a price $t = \phi_B^{-1}(v_S)$ for his item.
- The buyer can accept to trade at that price, or reject the offer.

Notice that the mechanism is not dominant strategy incentive compatible, as knowledge of v_B would lead the seller to report $t = v_B - \epsilon$. The mechanism is not presented in the usual form, where both agents report their valuations, but it can be easily converted by having the mechanism post this price given the seller's report.

A $1/2$-approximation of the gain from trade was achieved using a Bayesian incentive compatible mechanism by [19], under a relaxed benchmark. The approximation is measured against the best possible BIC, IR, and SBB mechanism, which less efficient than the more commonly used theoretical optimal. A very recent work, [4],

proposes mechanisms that achieve gain from trade guarantees for both optimality benchmarks: against the optimal and best possible BIC, IR and SBB.

2.2.4 Profit

Of course, the objective of profit maximisation plays an important role in bilateral trade as well. The intermediary is only interested in maximising his revenue, by selling the item at a higher price, disregarding social welfare (although increasing it is necessary for his own benefit, by individual rationality).

Definition 2.7. *The profit of mechanism $\mathcal{M}$ for reported valuations $\hat{v}_S, \hat{v}_B$ is:*

$$p^B(\hat{v}_S, \hat{v}_B) - p^S(\hat{v}_S, \hat{v}_B). \tag{2.19}$$

Clearly, such a mechanism cannot be budget balanced.

Myerson and Satterthwaite [59] settled the case for bilateral trade with Bayesian priors using a technique reminiscent of *virtual valuations* from Myerson's seminal paper [58]. This was extended to many buyers and sellers, each trading a single identical item, in [27]. In the same paper, truthful approximation algorithms were devised for variants with different item types. Gerstgrasser et al. showed that for correlated distributions, revenue maximisation is NP-complete in [38].

Chapter 3

Online Market Intermediation

3.1 Introduction

Our first extension to the bilateral trade is inspired by McAfee's work in [56]. As noted previously, he considered markets with many, unit demand buyers and sellers having independent, identical distributions and showed that the approximation to the welfare vanishes as the market size increases. In particular, the proposed *trade reduction* mechanism is $1-1/k$ approximate, where k is the optimal number of trades. In practice however, the assumption that every participant of this large market is available at the same time might not be always the most plausible. Considering that posted price mechanism, which are trivially truthful and easy to implement, have achieved a great deal of success in auction settings, it makes sense to consider such a setting where buyers and sellers arrive sequentially.

Thankfully, the online algorithms and economics literature on sequential decision making and optimal stopping is very well developed. There are two main ways to formulate this setting and have access to the necessary mathematical toolbox: as a secretary problem or as a prophet inequality. Before moving on to our exact setting specification, we provide a short introduction to secretary and prophet inequalities,

highlighting the main results, as extension of these arguments will be very often used in our design and analysis.

3.1.1 Prophets and Secretaries

The 'secretary problem' and its cousin, the 'prophet inequality', have received very significant attention, drawing researchers from fields such as statistics, economics and computer science. Imagine a person trying to sell a house. Every day, a different prospective buyer would come to check it out and make an offer. The seller has to immediately and irrevocably decide whether to accept the offer or not, knowing that if the offer is rejected, the buyer would leave never to come again. The goal is to maximise the profit of the seller. To tackle the inherent uncertainty a measure known as the *competitive ratio* is employed: the ratio between the *offline* optimum, the best offer after all potential buyers have been considered, compared to the expected value of the offer selected by the seller. Since it is impossible to obtain any meaningful fraction of the optimal offer in the worst case, there are two common restrictions to this input. In the secretary problem, all offers are chosen by an adversary and then *uniformly permuted*, while in the prophet inequality they represent *independent* draws of known distributions according to an adversarial order. In both settings, the number of total offers is known from the beginning. Both versions have been studied extensively and optimal results are known: in particular, the competitive ratio of the secretary problem is e [31] and of the prophet inequality is 2 (e.g. [52]).

In this chapter we mostly focus on the prophet inequality. We show the proof of the 2-approximation, as it is quite short and helpful in underlining the intuition necessary to understand our approach. Formally, the prophet inequality is a game with n stages. At each step i, the algorithm is offered a draw π_i from distribution F_i, where all the $F_1, \ldots, F_n$ are known in advance. After observing π_i, we can immediately accept it

or irrevocably reject it and move to step $i+1$. The following result first appeared in [69] and the presentation is adapted from Tim Roughgarden's lecture notes[1].

Theorem 3.1. *For every sequence $F_1, \ldots, F_n$ of independent distributions, the algorithm that uses threshold $t = \Pr[\max_i \pi_i \geq t] = 1/2$ and accepts the first $\pi_i \geq t$ guarantees expected reward. of $\mathbb{E}[\max_i \pi_i]/2$.*

Proof. We use the notation $z^+ = \max\{z, 0\}$. Considering a threshold strategy t, let $q(t) = \Pr[\max_i \pi_i < t]$ be the probability that no π_i is accepted. Clearly, $q(t)$ is an increasing function of t: as the threshold increases, it is less likely to accept any draw, but the reward will be higher.

This threshold strategy generates reward zero with probability $q(t)$ and at least t with probability $1 - q(t)$. We need to improve the second inequality. If any draw is greater that t, we would also get an 'excess' of $\pi_i - t$ where i is the index of the first draw with this property. For simplicity, we assume than only one draw exceeds t. Therefore:

$$\begin{aligned}
&\mathbb{E}[\text{payoff of } t \text{ threshold}] \\
&\geq (1 - q(t)) \cdot t + \sum_{i=1}^{n} \mathbb{E}[\pi_i - t | \pi_i \geq t, \pi_j < t \, \forall j \neq i] \Pr[\pi_i \geq t] \cdot \Pr[\pi_j < t \, \forall j \neq i] \\
&\geq (1 - q(t)) \cdot t + \sum_{i=1}^{n} \mathbb{E}[\pi_i - t | \pi_i \geq t] \Pr[\pi_i \geq t] \cdot \Pr[\pi_j < t \, \forall j] \\
&\geq (1 - q(t)) \cdot t + \sum_{i=1}^{n} \mathbb{E}\left[(\pi_i - t)^+\right] \cdot q(t), \qquad (3.1)
\end{aligned}$$

where we used the independence of the F_i's to disentangle the i-th draw from the

[1] http://theory.stanford.edu/~tim/f13/l/l6.pdf

rest. Turning our attention to the optimal reward:

$$\begin{aligned}
\mathbb{E}\left[\max_i \pi_i\right] &= \mathbb{E}\left[t + \max_i(\pi_i - t)\right] \\
&\leq t + \mathbb{E}\left[\max_i(\pi_i - t)\right] \\
&\leq t + \mathbb{E}\left[\max_i(\pi_i - t)^+\right] \\
&\leq t + \sum_{i=1}^{n} \mathbb{E}\left[(\pi_i - t)^+\right]. \qquad (3.2)
\end{aligned}$$

Comparing 3.1 and 3.2, a choice of $q(t) = 1/2$ seems sensible and gives us a 2-approximation.

This result is also tight. Fix $\epsilon > 0$ and let $\pi_1 = \epsilon$ always and $\pi_2 = 1$ with probability ϵ or $\pi_2 = 0$ with probability $1-\epsilon$. Without loss, any algorithm will accept the second draw. Therefore, no matter if the first draw is accepted or not, the expected reward cannot be greater than ϵ. The optimal reward however is $\epsilon\cdot(1-\epsilon)+1\cdot\epsilon = 2\epsilon-\epsilon^2$ and the ratio tends to $1/2$ as $\epsilon \to 0$. □

The take home point of the proof is that even though samples are chosen sequentially, there is a tight connection between the expected number of accepted choices and the final reward. In a certain light, we extend this result when many choices can be made, for probability distributions following some common assumptions found in the mechanism design literature.

Applications to Mechanism Design The prophet inequality was particularly appealing to economists who spearheaded the initial development, but in recent years it has been adopted by computer scientists as well, as a flexible framework to tackle complicated challenges in mechanism design. Hajiaghayi et al. [41] were the first to realise the connections with online auctions. They also considered a variant with uniform matroid constraints on the selected items. Their results were improved by Alaei [1] and expanded by Chawla et al. [21] to general matroid, where the mechanism

was also allowed to influence the order of arrivals. The more algorithmic matroid prophet inequality was formulated by Kleinberg and Weinberg [48] (who gave a 2-competitive algorithm) and has received plenty of attention in its own right [68].

3.2 Prophet Market Intermediation

Contrary to the assumptions in most of the market literature and armed with our knowledge of prophet inequalities, we consider a market setting with a key difference: the buyers and sellers appear one-by-one, in a dynamic way. It is natural to study this question in the incomplete information setting in which the intermediary, whose objective is to maximise either profit or welfare, does not know the sequence of buyers and sellers in advance. The framework that we employ to study the question is the standard worst-case analysis of online algorithms whose goal is to do as well as possible in the face of a powerful adversary which tries to embarrass them.

We are not the first to apply techniques from online algorithms to quantify uncertainty in markets: the closest work to ours would be by Blum et al. [13] who consider buyers and sellers trading identical items. In their setting, motivated mostly from a financial standpoint, buyers and sellers arrived in an online manner, with their bids appearing to the trader and expiring after some time. The trader would have to match prospective buyers and sellers to facilitate trade. Among a plethora of interesting results, the trader's profit maximisation problem was studied using competitive analysis and techniques from online weighted matching. The key difference in our setting is that buyers and sellers do not overlap: whenever a seller appears, the intermediary has to decide whether or not to attempt to buy the item, without having a buyer ready to go. Instead, the intermediary stores the item to sell it at a later time. We believe this variation is able to capture "slower" markets, like online marketplaces similar to Amazon or AliExpress (or even regular retail stores), where

uncertainty stems from not knowing how large a stock of items to buy, in expectation of the buyers to come.

3.2.1 Our Results

Our aim is to study this dynamic market setting, where an intermediary faces a sequence of potential buyers and sellers in an online fashion. The goal of the intermediary is to maximise his profit, or society's welfare, by buying from the sellers and selling to buyers. We take a Bayesian approach to their utilities but use competitive analysis for their arrivals: the main difficulty stems from the unknown (and adversarially chosen) sequence of agents. Further particulars and notation is discussed in Section 3.3. All the online algorithms we design are posted price, which are simple, robust and strongly truthful.

First, in Section 3.5 we study the case of arbitrary sequences of buyers and sellers and show that the competitive ratio—the ratio of the optimal offline profit over the profit obtained by the online algorithm—is $\Theta(\sqrt{n})$, where n is the total number of buyers and sellers. We also study the social welfare objective, where the goal is to maximise the total utility of all participants, including the sellers, the buyers and the intermediary. The competitive ratio here is $\Theta(\ln n)$. All these results are achieved via common regularity assumptions on the distributions of the agent values (see Section 3.4), which we also prove to be necessary, by providing arbitrarily bad competitive ratios in the case they are dropped (Theorem 3.4).

To overcome the above pessimistic results, we next study in Section 3.6 the setting where both the online and offline algorithms have a limited stock, i.e. at no point in time can they hold more than K items. Under this assumption, the competitive ratio is improved to $\Theta(K \ln n)$, asymptotically matching the guarantee obtained for welfare in the general setting (although in this case we are comparing profit against the restricted offline algorithm). Finally, we also propose a way to restrict the input

sequence, by introducing in Section 3.7 the notion of α-balanced streams, where at every prefix of the stream the ratio of the number of sellers to buyers has to be at least α, $\alpha \geq 1$. Under this condition we are able to bring down the competitive ratios for both objectives to constants. In particular, the online posted-price mechanism that we use for profit maximisation, and which is derived by a fractional relaxation of the optimal offline profit, achieves an asymptotically optimal ratio of $1 + o(\alpha^{3/2})$. A similar mechanism is 4-competitive for the welfare objective.

3.2.2 Prior Work on Sequential Posted Prices

Sequential auctions have produced a collection of interesting results, either extending the ideas of simple approximate mechanisms instead of more complex, theoretically optimal ones or dealing with entirely new settings. Blumrosen and Holenstein were among the first to compare the revenue (or welfare) generated by simple, posted-price sequential auctions to the optimal [15] for single item auctions with multiple, iid buyers. They showed that sequential posted price revenue converged to the optimal one for distributions with bounded support, but not for unbounded. Chawla et al. considered multi-dimensional settings in [21, 76], where the sets of agents acquiring items could also have matroid constraints and constructed posted price mechanisms with good approximation ratios, given the added power of controlling the order of the agents. Feldman et al. considered combinatorial auctions with XOS valuations in [35].

There have been many approaches that apply competitive (worst-case) analysis to mechanism design. The analysis of auctions with unlimited supply is explored in [10, 12] where near optimal algorithms are developed using techniques inspired from no-regret learning. A comprehensive exposition of online mechanism design by Parkes can be found in [63].

There are also positive results in online auctions when the valuation distribution is unknown (but usually known to be restricted in some way, having bounded support or being monotone hazard-rate etc). Babaioff et al. explored the case of selling a single item to multiple i.i.d. buyers in [3]. The case of k items in a similar setting was studied in [6], while the case of unlimited items (digital goods auctions) in [47] and [51]. Budget constraints where also introduced in [9], where a procurement auction was the focus.

3.3 Preliminaries and Notation

The input is a finite string $\sigma \in \{S, B\}^*$ of buyers (B) and sellers (S). The online algorithm has no knowledge of $\sigma(t)$, i.e. whether $\sigma(t) = S$ or $\sigma(t) = B$, before step t. Also, it doesn't know the length $n(\sigma)$ of σ. Denote $n_S(\sigma)$, $n_B(\sigma)$ the number of sellers and buyers, respectively, in σ, and let $N_S(\sigma)$, $N_B(\sigma)$ be the corresponding set of indices, i.e. $N_S(\sigma) = \{t \mid \sigma(t) = S\}$ and $N_B(\sigma) = \{t \mid \sigma(t) = B\}$. Let $N(\sigma) = N_S(\sigma) \cup N_B(\sigma) = \{1, 2, \ldots, n(\sigma)\}$. In the above notation we will often drop the σ if it is clear which input stream we are referring to.

The values of the sellers are drawn i.i.d. from a probability distribution (with cdf) F_S and these of buyers i.i.d. from a distribution F_B, both supported over intervals of nonnegative reals. We denote the random variable of the value of the t-th agent with X_t. We assume that distributions F_S and F_B are continuous, with bounded expectations μ_S and μ_B, and have (well-defined) density functions f_S and f_B, respectively. It will also be useful to denote by X_S a random variable drawn from distribution F_S, and similarly $X_B \sim F_B$, and for any random variable Y and positive integer m use $Y^{(m)}$ to represent the maximum order statistic out of m i.i.d. draws from the same distribution as Y. We will also use the shortcut notation $\mu^{(m)} = \mathbb{E}[Y^{(m)}]$.

We study *posted-price* online algorithms that upon seeing the identity of the t-th

agent (whether she is a seller or a buyer), offer a price p_t. We buy one unit of the item from sellers that accept our price (i.e. if $\sigma(t) = S$ and $X_t \leq p_t$) and pay them that price, and we sell to buyers that accept our price (i.e. if $\sigma(t) = B$ and $X_t \geq p_t$), given stock availability (see below), and collect from them that price. So, a price p_{t+1} can only depend on $\sigma(1), \ldots, \sigma(t+1)$ and the result of the comparison $X_i \leq p_i$ in all previous steps $i = 1, 2, \ldots, t$. Let K_t denote the available stock at the beginning of the t-th step, i.e. $K_1 = 0$ and

$$K_{t+1} = \begin{cases} K_t + 1, & \text{if } \sigma(t) = S \ \wedge \ X_t \leq p_t \\ K_t - 1, & \text{if } \sigma(t) = B \ \wedge \ K_t \neq 0 \ \wedge \ X_t \geq p_t \\ K_t, & \text{otherwise.} \end{cases}$$

Then, the set of sellers from whom we bought items during the algorithm's execution is $I_S = \{t \in N_S \mid X_t \leq p_t\}$ and the set of buyers we sold to is $I_B = \{t \in N_B \mid X_t \geq p_t \wedge K_t \neq 0\}$. Notice that these are random variables, depending on the actual realisations of the agent values X_t.

The total *profit* that the intermediary deploying an algorithm A makes throughout the execution on an input stream σ, is the amount he manages to collect from the buyers via successful sales, minus the amount he spent in order to maintain stock availability from the sellers, that is

$$\mathcal{R}(A, \sigma) = \mathbb{E}\left[\sum_{t \in I_B} p_t - \sum_{t \in I_S} p_t\right].$$

The social *welfare* of algorithm A is the sum of valuations that all participants achieve throughout the entire execution. That is, a seller at position t of the stream has a value

of X_t if she keeps her item, or a value of p_t if she sold the item to the intermediary; a buyer has a value of $X_t - p_t$ if she managed to buy an item, since the item has a value of X_t and he spent p_t to buy it, or 0 otherwise. And the intermediary, has a value of $\mathcal{R}(A)$ plus the value of the items that he didn't manage to sell in the end and which are now left in his stock. Putting everything together and performing the occurring cancellations, this results in the welfare to be expressed simply as the sum of the values of the sellers that kept their items plus the sum of the values of the buyers that bought an item, i.e.

$$\mathcal{W}(A, \sigma) = \mathbb{E}\left[\sum_{t \in N_S \setminus I_S} X_t + \sum_{t \in I_B} X_t\right]. \tag{3.3}$$

We use *competitive analysis*, the standard benchmark for online algorithms (see e.g. [18]), in order to quantify the performance of an online algorithm A: we compare it to that of an unrealistic, offline optimal algorithm OPT that has access to the entire stream σ in advance. Then, we say that A is $\rho(n)$-competitive with respect to welfare, if for any input sequence of agents σ with length n and distributions F_S, F_B for the agent values, it is $\mathcal{W}(\text{OPT}, \sigma) \leq \rho(n) \cdot \mathcal{W}(A, \sigma)$. Notice how we allow the competitive ratio $\rho(n)$ to explicitly depend on the input's length, so that we can perform asymptotic analysis as $\mathcal{W}(\text{OPT}, \sigma)$ and n tend to infinity. It is common in competitive analysis to allow for an additional constant in the right hand side of the above expression, that does not depend in the input, and which intuitively can capture some initial configuration disadvantage of the online algorithm. We do that for the case of the profit objective, as this constant will have a very natural interpretation: you can think of it as the maximum amount of deficit on which an online algorithm can run at any point in time, since an adversary can always stop the execution whenever he

wishes. Given this interpretation, it makes sense to allow for this constant to depend on seller distribution F_S since, even when we face a single seller at the first step, we expect to spend an amount that depends on the realisation of her value. Thus, we will say that an online algorithm is $\rho(n)$-competitive with respect to profit, if for any input sequence of agents σ and any probability priors F_S, F_B,

$$\mathcal{R}(\mathrm{OPT}, \sigma) \leq \rho(n) \cdot \mathcal{R}(A, \sigma) + O(\mu_S). \tag{3.4}$$

3.4 Distributional Assumptions

Throughout most of the chapter we will make some assumptions on the distributions F_B, F_S from which the buyer and seller values are drawn. In particular, we will assume that F_B has *monotone hazard rate (MHR)*, i.e. $\log(1 - F_B(x))$ is concave, and that F_S is *log-concave*, i.e. $\log F_S(x)$ is concave. For convenience, we will collectively refer to both the above constraints as *regularity assumptions*. These conditions are rather standard in the optimal auctions literature, and they encompass a large class of natural of distributions including e.g. exponential, uniform and normal ones. Notice that distributions that satisfy the above conditions also fulfil the regularity requirements introduced in the seminal paper Myerson and Satterthwaite [59] for the single-shot, one buyer and one seller setting of bilateral trade, namely that $x + \frac{F_S(x)}{f_S(x)}$ and $x - \frac{1-F_B(x)}{f_B(x)}$ are both increasing functions. Finally, we must mention that such regularity assumptions are necessary, in the sense that dropping them would result in arbitrarily bad lower bounds for the competitive ratios of our objectives, as it is demonstrated by Theorem 3.4.

Lemmas 3.1 and 3.2 demonstrate some key properties of distributions satisfying our regularity assumptions and which will be very useful in our subsequent analysis:

Theorem 3.2. *For any continuous random variable Y drawn from an MHR distri-*

bution with bounded expectation μ and standard deviation s,

1. $\Pr[Y \geq y] \geq \frac{1}{e}$ *for any* $y \leq \mu$
2. $\Pr[Y \geq y] < \frac{1}{e}$ *for any* $y > 2\mu$
3. $\mathbb{E}[Y^{(m)}] \leq H_m \cdot \mu$*, where* H_m *is the* m*-th harmonic number.*
4. $s \leq \mu$

Proof. A proof of Property 1 can be found in [11, Theorem 3.8], of Property 2 in [11, Corollary 3.10], and of Property 3 in [3, Lemma 13]. For Property 4, from [39, Lemma 2] we know that $\mathbb{E}[Y^2] \leq 2\mu^2$, so $s^2 = \mathbb{E}[Y^2] - \mu^2 \leq \mu^2$. □

Lemma 3.1. *For any distribution over* $[0, \infty)$ *with log-concave cdf* F *and expectation* μ*,*

$$x \leq e\mu F(x) \qquad \text{for any } x \leq \mu.$$

Proof. Fix some $x \leq \mu$ and let $c = \frac{x}{\mu}$. Define the random variable $Y = cX$, where X is drawn from F, and let F_Y be the cdf of Y. Since F is log-concave, $\ln F(t)$ is a concave function, and so from Jensen's inequality

$$\ln F(c\mu) = \ln F(\mathbb{E}[Y]) \geq \int_0^\infty \ln F(t)\, dF_Y(t) = \int_0^\infty \ln F(t) c\, dF(t) = c\int_0^1 \ln u\, du = -c.$$

So, $F(x) \geq e^{-c} = \frac{c\mu}{\mu}\frac{e^{-c}}{c} = \frac{x}{\mu}\frac{e^{-c}}{c}$. The lemma follows from the fact that $\frac{e^{-c}}{c}$ is decreasing for $c \in (0, 1]$. □

Finally, we prove the following property bounding the sum of maximum order statistics of a distribution, that holds for general (not necessarily MHR) distributions and might be of independent interest:

Lemma 3.2. *The expected average of the* k*-th highest out of* m *independent draws from a probability distribution with expectation* μ *and standard deviation* s *can be at most* $\mu + 2\sqrt{\frac{m}{k}}s$*.*

Proof. Let $Y^{(1:m)} \leq Y^{(2:m)} \leq Y^{(m:m)}$ denote the order statistics of m independent draws from a probability distribution with mean μ and standard deviation s. We want to prove that

$$\sum_{i=m-k+1}^{m} \mathbb{E}[Y^{i:m}] \leq k\mu + 2\sqrt{km}s.$$

From [2, Eq. (4)] we know that $\mathbb{E}[Y^{(i:m)}] \leq \mu + s\sqrt{\frac{i-1}{m-i+1}}$, so it is enough to show that $\sum_{i=m-k+1}^{m} \sqrt{\frac{i-1}{m-i+1}} \leq 2\sqrt{km}$. Indeed, by using the transformation $j = m - i + 1$, we get

$$\sum_{i=m-k+1}^{m} \sqrt{\frac{i-1}{m-i+1}} = \sum_{j=1}^{k} \sqrt{\frac{m}{j} - 1} \leq \sqrt{m} \sum_{j=1}^{k} \sqrt{\frac{1}{j}} \leq \sqrt{m} \int_0^k x^{-1/2}\, dx = \sqrt{m} \cdot 2\sqrt{k}.$$

□

3.5 General Setting

We start by studying the general setting where no additional assumptions are enforced on the structure of the input sequence. The adversary is free to arbitrarily choose the identities of the agents.

3.5.1 Welfare

Theorem 3.3. *Under our regularity assumptions*[2]*, the online auction that posts to every seller the median of F_S and to every buyer the median of F_B is $O(\ln n)$-competitive with respect to welfare. This bound is tight.*

Proof. We split the proof of the theorem in two more general lemmas below, corresponding to upper and lower bounds. Then, the upper bound for our case follows easily from Lemma 3.3 by using constants $c_1 = c_2 = 2$, and taking into consideration

[2] As matter of fact, in the proof of Theorem 3.3 just F_B being MHR would suffice.

that, from Property 3 of Theorem 3.2, the ratio of the maximum order statistic for the MHR distribution F_B is upper bounded by $r_B(m) \leq H_m \leq O(\ln m)$. For the lower bound, it is enough to observe that this ratio is attained by an exponential distribution, which is MHR.

Lemma 3.3. *For any choice of constants $c_1, c_2 > 1$, the following fixed-price online auction has a competitive ratio of at most $\max\left\{\frac{c_1}{c_1-1}, c_1 c_2 \cdot r_B(n_B)\right\}$ with respect to welfare, where n_B is the number of buyers, and $r_B(m) = \mu_B^{(m)}/\mu_B$ is the ratio between the m-maximum-order statistic and the expectation of the buyer value distribution.*

- *Post to all sellers price $q = F_S^{-1}\left(\frac{1}{c_1}\right)$.*
- *Post to all buyers price $p = F_B^{-1}\left(\frac{c_2-1}{c_2}\right)$.*

Proof. Let A denote our online algorithm and OPT an offline algorithm with optimal expected welfare. Fix an input stream σ. Looking at (3.3), the maximum welfare that OPT can get from the sellers is at most $\mathbb{E}\left[\sum_{t \in N_S} X_t\right] = n_s \mu_S$, while from the buyers at most $\mathbb{E}\left[|I_B| \cdot X_B^{(n_B)}\right] \leq \kappa\, \mathbb{E}\left[X_B^{(n_B)}\right]$, where κ is the maximum number of sellers that can be matched to *distinct* buyers that arrive after them[3] in σ: clearly, no mechanism can sell more than κ items. Bringing all together we have that

$$\mathcal{W}(\text{OPT}) \leq n_s \mu_S + \kappa \mu_B^{(n_B)} = n_s \mu_S + r_B(n_B) \cdot \kappa \mu_B.$$

For the online algorithm now, from the sellers we get

$$\sum_{i \in N_S} \Pr\left[X_i > q\right] \mathbb{E}[X_i | X_i > q] \geq n_s (1 - F_S(q))\, \mathbb{E}[X_S] = \frac{c_1 - 1}{c_1} \cdot n_S \mu_S$$

[3] You can think of that as the maximum size of a matching in the following undirected graph: the nodes are the sellers and the buyers, and there is an edge between any seller and all the buyers that appear after her in σ.

and from the buyers at least

$$\kappa \Pr[X_S \leq q] \Pr[X_B \geq p] \mathbb{E}[X_i | X_i \geq p] \geq \kappa F_S(q)(1 - F_B(p)) \mathbb{E}[X_B] = \frac{1}{c_1}\frac{1}{c_2} \cdot \kappa \mu_B,$$

just by considering one of the κ-size matchings discussed before: if we manage to buy from one of these κ sellers, then we will definitely have stock availability for the matched buyer. □

The upper bound in Lemma 3.3 cannot be improved:

Lemma 3.4. *For* any *probability distribution F, even if the seller and buyer values are i.i.d. from F, the sequence SB^n forces all posted-price online mechanisms to have a competitive ratio of $\Omega(r(n))$, where $r(n) = \mu^{(n)}/\mu$ is the ratio of the n-maximum-order statistic of distribution F to its expectation.*

Proof. Assume that the seller and buyer values are drawn i.i.d. from a distribution F. Let $Y \sim F$ denote a random variable following this distribution and denote $\mu = \mathbb{E}[Y]$, $\mu^{(n)} = \mathbb{E}[Y^{(n)}]$. Fix an online algorithm A that posts price q to the seller and prices $\mathbf{p} \equiv p_1, p_2, \ldots$ to the buyers. Notice that this sequence of buyer prices $\mathbf{p}$ cannot depend on the actual stream length n, since that is being selected adversarially.

We overestimate A's expected welfare by assuming that it gets maximum welfare from the first seller, i.e. $\mathbb{E}[Y] = \mu$, while at the same time buys for sure the item from her, so that it has stock availability to sell in the following sequence of buyers. Then, from (3.3), its expected welfare is given by

$$\mathcal{W}(\mathbf{p}) = \mu + \sum_{t=1}^{n} \pi(t) \cdot \lambda(p_t), \tag{3.5}$$

where

$$\pi(t) = \pi(\mathbf{p}, t) = \prod_{j=1}^{t-1} \Pr[Y < p_j] = \prod_{j=1}^{t-1} F(p_j)$$

and

$$\lambda(y) = \Pr[Y \geq y] \cdot \mathbb{E}[Y \mid Y \geq y] = (1 - F(y))\,\mathbb{E}[Y \mid Y \geq y] = \int_y^\infty x f(x)\,dx \leq \mu.$$

First we show that we can without loss assume that the buyer prices are non increasing. Indeed, for a contradiction suppose that exists a time step t^* such that $\alpha \equiv p_{t^*} < p_{t^*+1} \equiv \beta$. Consider now the online mechanism that uses prices $\mathbf{p}'$, where $\mathbf{p}'$ results from the original prices $\mathbf{p}$ if we flip the prices at steps t^*, t^*+1, i.e. $p'_{t^*} = \beta$, $p'_{t^*+1} = \alpha$, and $p'_t = p_t$ for all $t \neq t^*, t^*+1$. Then, the difference in the expected welfare between the two mechanisms is

$$\begin{aligned}
\mathcal{W}(\mathbf{p}') - \mathcal{W}(\mathbf{p}) &= \sum_{t=t^*}^{t^*+1} \pi(\mathbf{p}', t) \cdot \lambda(p'_t) - \sum_{t=t^*}^{t^*+1} \pi(t) \cdot \lambda(p_t) \\
&= \pi(t^*)\lambda(\beta) + \pi(t^*)F(\beta)\lambda(\alpha) - \pi(t^*)\lambda(\alpha) - \pi(t^*)F(\alpha)\lambda(\beta) \\
&= \pi(t^*)\left[(1 - F(\alpha))\lambda(\beta) - (1 - F(\beta))\lambda(\alpha)\right] \\
&= \pi(t^*)(1 - F(\alpha))(1 - F(\beta))\left(\mathbb{E}[Y \mid Y \geq \beta] - \mathbb{E}[Y \mid Y \geq \alpha]\right),
\end{aligned} \tag{3.6}$$

which is nonnegative since $\alpha < \beta$.

There are two options for the prices $\mathbf{p}$: either $F(p_t) = 1$ for all t, or $k = \min\{t \mid F(p_t) < 1\}$ is a well-defined positive integer that does not depend on n, in which case define the constant $c \equiv F(p_k) < 1$. From (3.5), in the former case it is easy to see that $W(\mathbf{p}) = \mu$, while in the latter one

$$\mathcal{W}(\mathbf{p}) \leq \mu + \pi(k) \sum_{t=k}^{n} F(p_k)^{t-k} \lambda(p_t) \leq \mu + \pi(k) \sum_{t=k}^{n} c^{t-k} \mu \leq \left(1 + \sum_{j=0}^{\infty} c^j\right) \mu = \frac{2-c}{1-c} \mu$$

On the other hand, it is a well-know fact from the theory of prophet inequalities (see e.g. [49]) that by using a price of $\frac{\mu^{(n)}}{2}$ for all the buyers an offline mechanism can achieve a welfare of at least $\frac{\mu^{(n)}}{2}$ from the buyers, given of course availability of stock.

So, by setting e.g. a price equal to the median of F for the seller, the optimal offline welfare is at least $\frac{1}{2}\mu + \frac{1}{4}\mu^{(n)} = \Omega(\mu^{(n)})$. □

□

As the following theorem demonstrates, the regularity assumption on the agent values is necessary if we want to hope for non-trivial bounds. In particular, the lower bound in Lemma 3.4 can be made arbitrarily high:

Theorem 3.4. *For any constant $\varepsilon \in (0,1)$, there exists a continuous probability distribution F such that any online posted-price mechanism has a competitive ratio of $\Omega(n^{1-\varepsilon})$ for welfare on the input sequence SB^n, even if the values of the sellers and the buyers are i.i.d.*

Proof. Fix some $\varepsilon \in (0,1)$ and choose the Pareto distribution with $F(x) = 1 - x^{-\frac{1}{1-\varepsilon}}$ for $x \in [1,\infty)$. The expected value of this distribution is $\mu = \frac{1}{\varepsilon}$ while the expectation of the maximum order statistic out of n independent draws is

$$\mu^{(n)} = \frac{n\Gamma(n)\Gamma(\varepsilon)}{\Gamma(n+\varepsilon)} \sim \Gamma(\varepsilon)n^{1-\varepsilon},$$

since $\lim_{n\to\infty} \frac{\Gamma(n+\varepsilon)/\Gamma(n)}{n^{\varepsilon}} = 1$, where $\Gamma(x)$ denotes the standard gamma function. So, as n grows large, the ratio in Lemma 3.4 becomes

$$r(n) = \frac{\mu^{(n)}}{\mu} = \varepsilon\Gamma(\varepsilon) \cdot n^{1-\varepsilon} \geq \frac{4}{5}n^{1-\varepsilon} = \Omega(n^{1-\varepsilon}).$$

□

3.5.2 Profit

Now we turn our attention to our other objective of interest, that of maximising the expected profit of the intermediary. As it turns out, this objective has some additional

challenges that we need to address. For example, as the following theorem demonstrates, if the distribution of seller values is bounded away from 0, the competitive ratio can be arbitrarily bad, even for i.i.d. values from a uniform distribution:

Theorem 3.5. *For any $a > 0$ and $\varepsilon \in (0,1)$, if the seller and buyer values are drawn i.i.d. from the uniform distribution over $[a,b]$ where $b > 2a$, then no online posted-price mechanism can have an approximation ratio better than $a\left(1-\frac{1}{k}\right)^4 n^{1-\varepsilon}$ with respect to profit, where $k = \frac{b}{a} - 1$. In particular, for any uniform distribution over an interval $[1,h]$ with $h \geq 3$ the lower bound is $\frac{1}{2^4} n^{1-\varepsilon} = \Omega\left(n^{1-\varepsilon}\right)$.*

Proof. Fix $a, b > 0$ such that $k \equiv \frac{b}{a} - 1 > 1$. Assume that the buyer and seller values are drawn i.i.d. from the uniform distribution $[a,b]$, i.e. the cdf is $F(x) = \frac{x-a}{b-a} = \frac{x-a}{ak}$ for all $x \in [a,(k+1)a]$. Consider the input stream $\sigma = S^{n/2}B^{n/2}$, for n even.

First, it is easy to see that for any $\varepsilon \in (0,1)$ no online algorithm can buy more than $\frac{n^\varepsilon}{128}\frac{1}{a}$ items from the sellers in the first part of the stream, otherwise it will have to spend more than $\frac{n^\varepsilon}{128} = \omega(1)$. This means that the maximum profit that an online algorithm can get, even if it manages to sell to the buyers all the items she bought from the sellers, is at most $\frac{n^\varepsilon}{128}\frac{1}{a}(b-a) = \frac{k}{128}n^\varepsilon$.

Consider an offline algorithm that posts to seller and buyers the prices corresponding to the $\frac{1}{8}\left(1-\frac{1}{k}\right)^2$ and $\frac{1}{2}\left(1-\frac{1}{k}\right)$ percentiles, respectively. That is, buyers get a price of $p = F^{-1}(y) = a(yk+1)$ and sellers $q = F^{-1}\left(\frac{y^2}{2}\right) = \frac{a}{2}(2+y^2k)$, where $y = \frac{1}{2}\left(1-\frac{1}{k}\right)$. Then, the probability that the offline algorithm buys an item from a specific seller is $F(q)$, resulting in the algorithm spending $\frac{n}{2}F(q)q$ in expectation. On the other hand, underestimate its expected income by considering only selling to the i-th buyer the item that you got from the i-th seller. Then, the probability of achieving a successful transaction with a particular buyer is $F(q)(1-F(p))$, resulting in an expected profit of at least

$$\frac{n}{2}F(q)(1-F(p))p - \frac{n}{2}F(q)q = \frac{n}{2}\frac{y^2}{2}\left[(1-y)F^{-1}(y) - F^{-1}\left(\frac{y^2}{2}\right)\right]$$
$$= a\frac{n}{8}y^3\left[(3y-2)k-2\right]$$
$$= \frac{ak}{128}n\left(1-\frac{1}{k}\right)^4.$$

□

If we consider distributions supported over intervals that include 0, under our regularity assumptions we can do a little better than the trivial lower bound of Theorem 3.5:

Theorem 3.6. *Under our regularity assumptions*[4]*, for agent values distributed over intervals that include* 0 *the following online posted-price mechanism achieves a competitive ratio of* $O(n^{\frac{1}{2}+\varepsilon})$ *for any* $\varepsilon > 0$*:*

- *Post to the* i*-th seller price* $q_i = F_S^{-1}\left(\frac{1}{e}\frac{1}{i^{1/2+\varepsilon}}\right)$
- *Post to all buyers price* $p = \mu_B$*.*

Proof. Fix an input stream σ of length n. Let μ_B and s_B be the expectation and standard deviation of the buyer value distribution F_B. As in the proof of Lemma 3.3, let κ denote the maximum number of sellers that can be matched to distinct buyers that arrive after them in σ. If $\mu_B^{(j:m)}$ denotes the expectation of the j-th largest out of m independent draws from F_B, since no algorithm can make more than κ sales over its entire execution, the optimal offline profit is upper bounded by

$$\sum_{j=1}^{\kappa}\mu_B^{(n_B-j+1:n_B)} \leq \sum_{i=n-\kappa+1}^{n}\mu_B^{(i:n)} \leq \kappa\mu_B + 2\sqrt{\kappa n}s_B \leq 3\sqrt{\kappa}\sqrt{n}\mu_B,$$

[4]Unfortunately Theorem 3.4 still applies, leading to a competitive ration of $\Omega(n^{1-\epsilon})$ if these assumptions are dropped.

where for the second inequality we have used Lemma 3.2 and for the last one we have used Property 4 from Theorem 3.2 and the obvious fact that $\kappa \leq n$.

For the analysis of the online mechanism now, the expected number of items that it gets from the first κ sellers is $\sum_{i=1}^{\kappa} F_S(q_i) = \frac{1}{e}\sum_{i=1}^{\kappa} \frac{1}{i^{1/2+\varepsilon}} \geq \frac{1}{e}\kappa^{1/2-\varepsilon}$. So, by considering the FIFO matching between these first κ sellers and their corresponding buyers (see Lemma 3.9), the expected income of our algorithm is at least $\frac{1}{e}\kappa^{1/2-\varepsilon}(1-F(p)) = \frac{1}{e}\kappa^{1/2-\varepsilon}(1-F(\mu_B)) \geq \frac{1}{e^2}\kappa^{1/2-\varepsilon}$, where in the last step we deployed Property 1 of Theorem 3.2. So, it only remains to be shown that the online algorithm does not spend more than a constant amount. Indeed, our expected spending is at most

$$\sum_{i=1}^{\infty} q_i F_S(q_i) \leq \sum_{i=1}^{\infty} e\mu_S F_S(q_i)^2 = \frac{1}{e}\mu_S \sum_{i=1}^{\infty} \frac{1}{i^{1+2\varepsilon}} = O(\mu_S),$$

where for the first inequality we have used Lemma 3.1, taking into consideration that seller prices q_i are decreasing and q_1 is below μ_S. This is true because again from Lemma 3.1 for $x = \mu_S$ we know that $\mu_S \leq e\mu_S F(\mu_S)$, or equivalently $F(\mu_S) \geq \frac{1}{e} = F(q_1)$. □

The algorithm of Theorem 3.6 is asymptotically optimal:

Theorem 3.7. *If the seller and buyer values are drawn i.i.d. from the uniform distribution over* $[0,1]$*, then no online posted-price mechanism can have an approximation ratio better than* $\Omega(\sqrt{n})$.

Proof. As in the lower bound proof of Theorem 3.5 we again deploy an input sequence $\sigma = S^{n/2}B^{n/2}$ with n even. Let $F(x) = x$ be the cdf of the uniform distribution over $[0,1]$. This time we argue that no online algorithm can buy more than $\Omega(\sqrt{n})$ items from the sellers, in expectation. Indeed, let q_i be the price that the online mechanism posts to the i-th seller. Then, the expected number of items m_σ bought from the sellers is $\sum_{i=1}^{n/2} F(q_i) = \sum_{i=1}^{n/2} q_i$, while the expected expenditure c_σ is $\sum_{i=1}^{n/2} F(q_i)q_i = \sum_{i=1}^{n/2} q_i^2$.

By the convexity of the function $t \mapsto t^2$ and Jensen's inequality it must be that

$$m_\sigma = \sum_{i=1}^{n/2} q_i \leq \sqrt{\frac{n}{2}} \left(\sum_{i=1}^{n/2} q_i^2 \right)^{\frac{1}{2}} = O\left(\sqrt{c_\sigma}\sqrt{n}\right),$$

so, given that our deficit must be $c_\sigma = O(\frac{1}{2})$, we get the desired $m_\sigma = O(\sqrt{n})$. As a result, the online profit can be at most $O(\sqrt{n}) \cdot 1 = O(\sqrt{n})$.

For the offline algorithm we use prices $q = \frac{1}{8}$ and $p = \frac{1}{2}$ for the buyers and sellers, respectively, and by an analogous analysis to that of the proof of Theorem 3.5, we get that the expected offline profit is at least

$$\frac{n}{2}F(q)(1 - F(p))p - \frac{n}{2}F(q)q = \frac{n}{2}\frac{1}{8}\left(1 - \frac{1}{2}\right)\frac{1}{2} - \frac{n}{2}\frac{1}{8}\frac{1}{8} = \frac{n}{128} = \Omega(n).$$

□

3.6 Limited Stock

If one looks carefully at the lower bound proof for the profit in Theorem 3.7, it becomes clear that the source of difficulty for any online algorithm is essentially the fact that without knowledge of the future, you cannot afford to spend a super-constant amount of money into accumulating a large stock of items, without the guarantee that there will be enough demand from future buyers. In particular, it may seem that the offline algorithm has an unrealistic advantage of using a stock of infinite size. The natural way to mitigate this would be to introduce an upper bound K on the number of items that both the online and offline algorithms can store at any point in time. As it turns out, this has a dramatic improvement in the competitive ratio for the profit. On the other hand, the welfare guarantee is not improved asymptotically: the same logarithmic lower bound still applies, as described in Remark 1 at the end of this section.

Theorem 3.8. *Assuming stock sizes of at most K items, under our regularity assumptions the following online mechanism is $O(Kr \ln n)$-competitive, where $r = \max\left\{1, \frac{\mu_S}{\mu_B}\right\}$:*

- *If your stock is not currently full, post to sellers price $q = F_S^{-1}\left(\frac{1}{r}\frac{1}{2eK}\right)$*
- *Post to all buyers price $p = \mu_B$.*

Proof. The proof is similar to that of Theorem 3.6, but certain points need some special care. Let κ again be the maximum number of sellers that can be matched to distinct buyers that follow them, but this time under the added restriction of the K-size stock. This corresponds to the maximum matching with no "temporal" cut of size greater than K. We write "temporal" cut to mean any cut in the graph that separates the vertices (buyers and sellers) $1 \dots i$ from vertices $i+1 \dots n$ — that is, precisely the condition that we cannot match more than K sellers from an initial segment to buyers later in the sequence. Lemma 3.9 in the appendix of this chapter demonstrates that such a κ-size matching can be computed not only offline, but also online using a FIFO queue of length K, adding sellers to the queue while it is not full and matching buyers greedily: we post prices to sellers, only if we have free space in our stock, i.e. when the matching queue is not full. The proof is a straightforward application of the potential method. We underestimate the online profit by considering only selling an item to the buyer that is matched to the seller from which we bought the item. Mimicking the analysis in the proof of Theorem 3.6 we can see that the expected number of items bought from the κ matched sellers is $\kappa F_S(q) \geq \kappa \frac{1}{2eK}\frac{1}{r}$.

Now we argue that $q \leq \frac{\mu_B}{2}$. Indeed, since $F_S(q) \leq \frac{1}{e}$ we know for sure that $q \leq \mu_S$, and so from Lemma 3.1 it is $q \leq e\mu_S F(q) \leq e\mu_S \frac{\mu_B}{\mu_S}\frac{1}{2e} = \frac{\mu_B}{2}$. Next, notice that whenever we make a successful sale, the contribution to profit is $p - q \geq \mu_B - \frac{\mu_B}{2} =$

$\frac{1}{2}\mu_B$. Thus, the total expected gain in profit from sales is at least

$$\kappa F_S(q)(1-F_B(p))(p-q) \geq \kappa \frac{1}{2eK} \frac{1}{\frac{\mu_S}{\mu_B}+1}(1-F_B(\mu_B))\frac{1}{2}\mu_B \geq \frac{1}{4e^2}\frac{1}{Kr}\kappa\mu_B,$$

where in the bound for the quantile $1-F_B(\mu_B)$ we used Property 1 of Theorem 3.2. Also, the profit we loose from the cost of unsold items cannot be more than $Kq \leq K\mu_S e^{\frac{1}{2eK}} = O(\mu_S)$. On the other hand, the offline profit is at most κ times the expected maximum order statistic out of n independent draws from F_B, so by Property 3 of Theorem 3.2 it is upper bounded by $\kappa H_n \mu_B$. Putting everything together, the competitive ratio of the online algorithm is at most

$$\frac{\kappa H_n \mu_B}{\frac{1}{4e^2}\frac{1}{Kr}\kappa\mu_B} = O\left(Kr \ln n\right).$$

□

Remark 1. *The upper bound in Theorem 3.8, although a substantial improvement from the $\Theta(\sqrt{n})$ one for the general case in Theorem 3.6, it cannot be improved further: the logarithmic lower bound is unavoidable, since a careful inspection of the welfare lower bound in the proof of Lemma 3.4 reveals that the same analysis carries over to the profit. In particular, the last parenthesis of $\mathbb{E}[Y \mid Y \geq \beta] - \mathbb{E}[Y \mid Y \geq \alpha]$ in (3.6) will be replaced by $\beta - \alpha$ which is still nonnegative, and also the bad instance sequence of SB^n does not use a stock of size more than 1. We try to overcome these obstacles by considering a different model of constrained streams in the following section.*

3.7 Balanced Sequences

As we saw in Section 3.6, introducing a restriction in the size of available stock can improve the performance of our online algorithms with respect to profit. However,

the bound is still super-constant. Thus, it is perhaps more reasonable to assume some knowledge of the ratio α between sellers and buyers in sequences the intermediary might face. This allows us finer control over the trade-off between high volume of trades and the hunt for greater order statistics.

In this section we analyse the competitive ratio for profit and welfare obtained by online algorithms on α-*balanced* sequences.

Definition 3.1. *Let α be a positive integer. A sequence containing m buyers is called α-balanced if it contains exactly αm sellers and the i-th buyer is preceded by at least αi sellers.*

For example, the sequence $SBSSBSBB$ is 1-balanced, but $SBBSSB$ is not. Similarly, $SSSBSB$ is 2-balanced, while $SSBSBSSSB$ isn't. Note that since $n = n_S\frac{\alpha+1}{\alpha} = n_B(\alpha+1)$, we only need to know the number of buyers of a sequence. For convenience, we will denote it by m instead of n_B, as it is will be used quite often from now on.

3.7.1 Profit

We first work on profit, deriving bounds for a variety of online and offline mechanisms. Naturally, there are two types of offline mechanisms: adaptive and non-adaptive. The *non-adaptive* posted-price mechanism calculates all prices in advance based on the sequence of buyers and sellers, while the *adaptive* posted-price mechanism can alter the prices on the fly, depending on the outcomes of previous trades. We define adaptive(σ) as the profit generated by the adaptive offline for given sequence σ. The adaptive offline is more powerful than the non-adaptive, therefore for any σ we have adaptive(σ) $\geq$ non-adaptive(σ), but we can show that in fact their are asymptotically equal, which we use to our advantage.

We show that there is a competitive online mechanism for α-balanced sequences. To do this, we compare the optimal adaptive and non-adaptive profit to the profit of

a class of hypothetical (offline) mechanisms, called *fractional mechanisms*, which are allowed to buy fractional quantities of items: posting the price p would buy exactly $F_S(p)$ items or sell $1 - F_B(p)$ items. The advantage of using fractional mechanisms is that at any point we know the exact quantity of items in the hands of the intermediary instead of the expectation; an immediate consequence of this is that we know in advance whether there is enough quantity to sell, which implies that *the adaptive and non-adaptive versions of the optimal fractional mechanism are identical.*

We can now give an outline of the results in this section: For α-balanced sequences σ with m buyers and αm sellers, we establish the following relations of optimal profits:

$$\text{adaptive}(\sigma) \leq \text{fractional}(\sigma) \leq \text{fractional}(S^{\alpha m}B^m) \approx \text{non-adaptive}(\sigma). \tag{3.7}$$

The last quantity in this chain will be achieved (within a constant-factor approximation) by our online algorithm. We begin by the fractional offline mechanism.

Theorem 3.9. *The profit gained by the optimal fractional mechanism for the sequence* $S^{\alpha m}B^m$ *is*

$$\begin{aligned} \max \quad & m\left(p(1 - F_B(p)) - \alpha \cdot qF_S(q)\right) \\ s.t. \quad & 1 - F_B(p) = \alpha F_S(q) \\ & p, q \in [0, \infty). \end{aligned} \tag{3.8}$$

Proof. The profit and optimal prices can be calculated through the following optimisation:

$$\begin{aligned} \max \quad & \sum_{i=1}^{m} p_i(1 - F_B(p_i)) - \sum_{i=1}^{\alpha m} q_i F_S(q_i) \\ \text{s.t.} \quad & \sum_{i=1}^{m}(1 - F_B(p_i)) \leq \sum_{i=1}^{\alpha m} F_S(q_i) \\ & p_i, q_i \in [0, \infty), \end{aligned}$$

where q_i and p_i are the prices for buying and selling respectively. However, we can assume that the first constraint is tight, as all q_i's can be lowered until equality is

achieved, without hurting the trades happening in the second half of the sequence. Remember, these are *not* in expectation, but rather, fractions.

This constrained optimisation can be reduced to finding stationary points of its Lagrange function

$$\mathcal{L} = \sum_{i=1}^{m} p_i(1 - F_B(p_i)) - \sum_{i=1}^{\alpha m} q_i F_S(q_i) - \lambda(\sum_{i=1}^{m}(1 - F_B(p_i)) - \sum_{i=1}^{\alpha m} F_S(q_i)).$$

Taking its derivative with respect to price p_i we get:

$$(1 - F_B(p_i)) - p_i f_B(p_i) = -\lambda f_B(p_i) \qquad \Leftrightarrow \qquad p_i - \frac{1 - F_B(p_i)}{f_B(p_i)} = \lambda,$$

which has at most one solution for any given λ due to the distribution being regular. The treatment of q_i's is similar, leading to a unique solution as well. Thus, since $p_i = p$ and $q_i = q$ for all i we obtain the stated result. □

For other sequences containing αm sellers and m buyers in a different order, we can use the following lemma to establish the middle part of inequality (3.7).

Lemma 3.5. *For any α-balanced σ with m buyers, fractional$(\sigma) \leq$ fractional$(S^{\alpha m}B^m)$*

Proof. Let q_i, p_i be the prices set by the optimal fractional mechanism for sequence σ. These prices have to satisfy $\sum_1^m(1 - F_B(p_i)) \leq \sum_1^{\alpha m} F_S(q_i)$, to ensure that the total quantity of items sold does not exceed the amount bought. Thus, the prices p_i, q_i represent a feasible solution to the optimisation problem for the sequence $S^{\alpha m}B^m$ and by definition, their profit is at most as much as the optimal. □

We now compare the adaptive and fractional algorithms. The intuition behind the proof of the theorem is that the expected value optimal adaptive profit is exactly the quantity that the fractional algorithm tries to directly maximise.

Theorem 3.10. *For any sequence σ we have adaptive$(\sigma) \leq$ fractional(σ).*

Proof. Fix an adaptive mechanism and let Q_i be the price posted to seller i and $\tilde{Q}_i$ be the probability of sale at price Q_i. Since in an adaptive mechanism the price depends on the history, Q_i and $\tilde{Q}_i$ are random variables. Similarly define P_j and $\tilde{P}_j$ to be the price and probability of buying from buyer j. For the payments to sellers and from buyers we have:

$$\mathbb{E}\left[Q_i F_S(Q_i)\right] = \mathbb{E}\left[\tilde{Q}_i F_S^{-1}(\tilde{Q}_i)\right]$$
$$\mathbb{E}\left[P_j(1 - F_B(P_j))\right] = \mathbb{E}\left[\tilde{P}_j F_B^{-1}(1 - \tilde{P}_j)\right].$$

Summing over all agents we get the expected profit:

$$\begin{aligned}
&\sum_{j \in N_B} \mathbb{E}\left[P_j(1 - F_B(P_j))\right] - \sum_{i \in N_S} \mathbb{E}\left[Q_i F_S(Q_i)\right] \\
&= \sum_{j \in N_B} \mathbb{E}\left[\tilde{P}_j F_B^{-1}(1 - \tilde{P}_j)\right] - \sum_{i \in N_S} \mathbb{E}\left[\tilde{Q}_i F_S^{-1}(\tilde{Q}_i)\right] \\
&\leq \sum_{j \in N_B} \mathbb{E}[\tilde{P}_j] F_B^{-1}(1 - \mathbb{E}[\tilde{P}_j]) - \sum_{i \in N_S} \mathbb{E}[\tilde{Q}_i] F_S^{-1}(\mathbb{E}[\tilde{Q}_i]), \qquad (3.9)
\end{aligned}$$

where the last inequality follows from our regularity assumptions. Note that in the last inequality $F_B^{-1}(1 - \mathbb{E}[\tilde{P}_j])$ and $F_S^{-1}(\mathbb{E}[\tilde{Q}_i])$ can be interpreted as prices set by the fractional mechanism, with $\mathbb{E}[\tilde{P}_j]$ and $\mathbb{E}[\tilde{Q}_i]$ the fractions of items bought and sold.

We have obtained the objective function of the optimisation and it is left to a set of inequalities concerning the prices, to serve as the constraints. Observe that $\mathbb{E}[\tilde{Q}_i]$ is the expected number of items bought from seller i, while $\mathbb{E}[\tilde{P}_j]$ sold to buyer j. Let $\mathcal{S}_t$ and $\mathcal{B}_t$ be the sets of indices of sellers and buyers contained in the first t agents of the sequence.

Let Z_t be the number of items exchanged with the agent encountered at step t. The number of items currently held by the intermediary at time t is $\sum_1^t Z_i \geq 0$ by

the no short selling assumption. Thus for all t:

$$\begin{aligned}\mathbb{E}\left[\sum_{i=1}^{t} Z_i\right] &= \sum_{i\in\mathcal{S}_t}\mathbb{E}\left[Z_i\right] - \sum_{j\in\mathcal{B}_t}\mathbb{E}\left[Z_j\right] \\ &= \sum_{i\in\mathcal{S}_t}\mathbb{E}\left[\mathbb{E}\left[Z_i|\tilde{Q}_i\right]\right] - \sum_{j\in\mathcal{B}_t}\mathbb{E}\left[\mathbb{E}\left[Z_j|\tilde{P}_j\right]\right] \\ &= \sum_{i\in\mathcal{S}_t}\mathbb{E}[\tilde{Q}_i] - \sum_{j\in\mathcal{B}_t}\mathbb{E}[\tilde{P}_j] \geq 0 \end{aligned} \tag{3.10}$$

Combining 3.9 and (3.10) gives us exactly the same optimization problem the optimal fractional mechanism would face for that sequence. □

At this point, we have a clear model of the adversary's power: the fractional mechanism's revenue for sequence $S^{\alpha m}B^m$, setting only two prices p, q for sellers and buyers. Could we do the same online? It seems likely. After all, long sequences of buyers and sellers seem to lead to a similar amount of trading on average by a mechanism setting the same prices.

Based on the previous discussion we propose the following online posted price algorithm:

- Use prices p, q given by the optimal fractional solution for $S^{\alpha m}B^m$(see Theorem 3.9).

This algorithm works without knowing the length of the sequence chosen by the adversary.

Lemma 3.6. *Let A be the online algorithm defined by the optimal fractional offline prices of (3.8). Consider two α-balanced sequences σ_1 and σ_2 of equal length. We write $\sigma_1 \succ \sigma_2$ whenever every prefix of σ_1 contains more sellers than the prefix of σ_2 having equal length. Then, $\sigma_1 \succ \sigma_2 \Rightarrow \mathcal{R}(A, \sigma_1) \geq \mathcal{R}(A, \sigma_2)$*

Proof. Assume the draws of σ_1 and σ_2 come from the same probability space, so that the i-th agent gets the same draw in both sequences. We will show that all trades

(or at least as many) that happened in σ_2 will occur in σ_1. Let i be the index of an arbitrary buyer that was matched to a seller in σ_2 and k the number of items in stock when he arrives in σ_1. If $k > 0$, then we trade with him as we would do in σ_2. If $k = 0$, we have already traded at least as many items as σ_2 at this point. To see this, note that since $\sigma_1 \succ \sigma_2$, at least as many items have been bought from the first $i-1$ agents of σ_1 than from σ_2 and because $k = 0$, at least as many have been traded. □

Although not all sequences are comparable (e.g. $SSBBSB$ and $SBSSBB$), the sequence $(S^\alpha B)^m$ is the bottom element among all α-balanced sequences of length $(\alpha+1)m$. This is trivial, as any balanced sequence must have at least $\lceil \frac{i}{(\alpha+1)/(\alpha)} \rceil$ sellers for any prefix of length i and $(S^\alpha B)^m$ is tight for this bound.

To formalise our intuition of making the same number of trades in the long run, we reformulate our algorithm in the more familiar setting of random walks. Instead of considering agents separately, each "timestep" would be one sub-sequence $S^\alpha B$, giving m steps in total. Thus, we are interested in the random variables Z_i, denoting the items in stock at the end of each step, starting with $Z_0 = 0$. Knowing the algorithm buys $\alpha m F_S(q)$ items in expectation, the expected profit can be given by

$$\mathcal{R}((S^\alpha B)^m) = (\alpha m F_S(q) - \mathbb{E}[Z_m])(p-q) - \mathbb{E}[Z_m]\, q, \tag{3.11}$$

which is the revenue of the expected number of trades minus the cost of the unsold items.

Lemma 3.7. $\mathbb{E}[Z_m] \leq \sqrt{2m\alpha^2 \ln m}\left(1 - \frac{2}{m}\right) + 2$

Proof. The process Z_i is almost a martingale but not quite: clearly $\mathbb{E}[Z_i] \leq \alpha m$ for all i and we do have $\mathbb{E}[Z_{i+1} | Z_i \geq 1] = Z_i$ since the expected change in items after that step is $\alpha F_S(q) - (1 - F_B(p)) = 0$ by Theorem 3.9. However, $\mathbb{E}[Z_{i+1} | Z_i = 0] > Z_i$, by the no short selling assumption.

We can define Y_i in the same probability space, where $Y_0 = 0$, and

$$Y_{i+1} = \begin{cases} Z_{i+1} & \text{if } Y_i > 0 \\ -Z_{i+1} & \text{if } Y_i < 0 \\ \begin{cases} Z_{i+1} & \text{with probability } \frac{1}{2} \\ -Z_{i+1} & \text{with probability } \frac{1}{2} \end{cases} & \text{if } Y_i = 0 \end{cases}. \tag{3.12}$$

The crucial observation is that Y_i behaves similar to Z_i but has no barrier at 0. Notice, that $|Y_i| \geq Z_i$ for all i and Y_i is a martingale.

Moreover, we have that $|Y_{i+1} - Y_i| \leq \alpha$ thus by the Azuma-Hoeffding inequality we can bound the expected value $\mathbb{E}[Z_m]$:

$$\Pr[Z_m \geq x] \leq \Pr[|Y_m| \geq x] = \Pr[|Y_m - Y_0| \geq x] \leq 2e^{\frac{-x^2}{2m\alpha^2}} \Rightarrow \tag{3.13}$$

$$\mathbb{E}[Z_m] \leq x\left(1 - 2e^{\frac{-x^2}{2m\alpha^2}}\right) + 2\alpha m e^{\frac{-x^2}{2m\alpha^2}}, \tag{3.14}$$

where we can set $x = \sqrt{2m\alpha^2 \ln m}$ to obtain the simpler form:

$$\mathbb{E}[Z_m] \leq \sqrt{2m\alpha^2 \ln m}\left(1 - \frac{2}{m}\right) + 2\alpha. \tag{3.15}$$

□

Lemma 3.8. *Let $r = \max\left\{2, \frac{\mu_S}{\mu_B}\right\}$. The optimal value of Programme (3.8) is at least $m\frac{\mu_B}{2er}$. Furthermore, at any optimal solution the buyer price has to be at most $p \leq 4\ln(4er)\mu_B$.*

Proof. Consider the value of Programme (3.8) that corresponds to the solution determined by the seller price q such that $F_S(q) = \frac{1}{e\alpha r}$. In a similar way to the proof of Theorem 3.6, it is again easy to see that $q \leq \mu_S$ since $F_S(q) \leq \frac{1}{e}$, and so by Lemma 3.1 and the regularity of F_S we get that $q \leq e\mu_S\frac{1}{e\alpha r} \leq \frac{\mu_S}{r}\frac{\mu_B}{2}$. Furthermore,

for the corresponding buyer price p we have $1 - F_B(p) = \alpha F_S(q) = \frac{1}{er} < \frac{1}{e}$ and so from Property 1 of Theorem 3.2 we get that $p \geq \mu_B$. Thus, the objective value of the particular solution is at least $m\alpha F_S(q)(p-q) \geq m\frac{1}{er}(\mu_B - \frac{\mu_B}{2}) = m\frac{\mu_B}{2er}$.

Next, for the upper bound on the buyer price, consider a solution that has buyer price $\hat{p} = cp^*$ for $c \geq 1$, where $F_B(p^*) = 1 - \frac{1}{e}$. Then, since F_B is an MHR distribution, $(1-F_B(x))^{\frac{1}{x}}$ is decreasing with respect to x, as can be verified using that $\log(1-F_B(x))$ is concave (see e.g. [11]), so $1 - F_B(\hat{p}) \leq (1 - F_B(p^*))^{\frac{\hat{p}}{p^*}} = e^{-c}$. Furthermore, since $F_B(p^*) = 1 - \frac{1}{e}$, from Property 2 of Theorem 3.2 it must be that $p^* \leq 2\mu_B$, and thus $\hat{p} \leq 2c\mu_B$, resulting in

$$1 - F_B(2c\mu_B) \leq 1 - F_B(\hat{p}) \leq e^{-c}.$$

This means that if we use a solution with $p = 2c\mu_B$, for some $c \geq 1$, the objective value of the Programme cannot exceed $m(1 - F_B(p))(p-q) \leq me^{-c}2c\mu_B$. So, unless this value is at least $m\frac{\mu_B}{2er}$, the particular choice of p cannot be part of an optimal solution. Thus, it must be $ce^{-c} \geq \frac{1}{4er}$. It is not difficult to check that this requires $c \leq 2\ln(4er)$, since $2\ln x e^{-2\ln x} = \frac{2\ln x}{x^2} < \frac{1}{x}$ for any $x > 0$ and ce^{-c} is a decreasing function for $c \geq 1$. As a result of the above analysis we can conclude that the buyer price p of any optimal solution in Programme (3.8) must be such that $p < 2\mu_B$, or otherwise satisfy $p \leq 2 \cdot 2\ln(4er) \cdot \mu_B = 4\ln(4er)\mu_B$. In any case, the desired upper bound for p in the theorem's statement holds. □

Theorem 3.11. *Under our regularity assumptions, the proposed non-adaptive online mechanism is* $(1 + o(\alpha^{3/2} r \ln r))$*-competitive for any balanced sequence, where* $r = \max\left\{2, \frac{\mu_S}{\mu_B}\right\}$.

Proof. Plugging (3.15) into (3.11), we get:

$$\begin{aligned}\mathcal{R}((S^\alpha B)^m) &\geq \alpha m F_S(q)(p-q) - \mathbb{E}\left[Z_m\right](p-q) - \mathbb{E}\left[Z_m\right] q \\ &\geq \alpha m F_S(q)(p-q) - \left(\sqrt{2m\alpha^2 \ln m}\left(1-\frac{2}{m}\right) + 2\alpha\right) p \\ &\geq \alpha m F_S(q)(p-q) - O(\alpha\sqrt{m \ln mp}). \end{aligned} \tag{3.16}$$

Using Lemma 3.5, Theorem 3.10 and Theorem 3.9 we know that for every α-balanced sequence, the profit of our non-adaptive online algorithm is at least $\mathcal{R}((S^\alpha B)^m)$ and the optimal offline is at most that of the fractional on sequence $S^{\alpha m}B^m$, i.e. $\alpha m F_S(q)(p-q)$. Thus, the second term in (3.16) bounds the additive difference of the online and optimal offline profit, and its ratio with respect to the offline profit is upper bounded by

$$O\left(\frac{\alpha\sqrt{m \ln mp}}{\alpha m F_S(q)(p-q)}\right) = O\left(\frac{\alpha\sqrt{m \ln m}\mu_B \ln(4er)}{m\frac{\mu_B}{2er}}\right) = O\left(\alpha^{3/2}\sqrt{\frac{\ln n}{n}} r \ln r\right) = o(\alpha^{3/2} r \ln r),$$

using $m = n/(\alpha+1)$.

□

Remark 2. *Among all 1-balanced sequences, the sequence that gives the maximum profit is not the sequence S^mB^m; intuitively, by moving some buyers earlier in the sequence, we obtain an improved profit by adapting the remaining buying prices to the outcome of these potential trades. For example, it should be intuitively clear that the sequence $S^{m/2}BS^{m/2}B^{m-1}$ has (slightly) better adaptive profit than the sequence S^mB^m for large m. Our work above shows that the difference is asymptotically insignificant, but it remains an intriguing question to determine the balanced sequence with the maximum profit.*

3.7.2 Welfare

Welfare on balanced sequences also improves the competitive ratio of Theorem 3.3 to a constant. Intuitively, the reason is that the high volume of possible trades dampens the advantage the adversary has in obtaining higher order statistics from buyers. As before, the fact that all sellers start with some contribution to the welfare is also helpful.

Theorem 3.12. *The online auction that posts to any seller and buyer the median of their distribution is* 4*-competitive.*

Proof. The algorithm buys from half the sellers in expectation, so in the end the welfare obtained just from sellers is at least:

$$\mathbb{E}\left[\sum_{t\in N_S\setminus I_S} X_t\right] = \sum_{t\in N_S} \mathbb{E}\left[X_t | X_t \geq q\right](1 - F_S(q)) \geq \frac{1}{2} n_S \mu_S.$$

Following the proof of Lemma 3.3, let κ denote the maximum size of a matching between sellers and buyers. Since the input is α-balanced, we are guaranteed that every buyer is preceded by some *distinct* seller, meaning that κ is exactly N_B. The welfare obtained from buyers is

$$\kappa \Pr\left[X_S \leq q\right] \Pr\left[X_B \geq p\right] \mathbb{E}\left[X_B | X_B \geq p\right] \geq \frac{1}{4} n_B \mu_B,$$

Adding everything together, the online algorithm gets at least $\frac{1}{4}(n_B\mu_B + n_S\mu_S)$. On the other hand the optimal welfare is at most:

$$\mathbb{E}\left[\sum_{t\in N_S\setminus I_S} X_t + \sum_{t\in I_B} X_t\right] \leq \mathbb{E}\left[\sum_{t\in N_S} X_t + \sum_{t\in N_B} X_t\right] = n_S\mu_s + n_B\mu_B.$$

□

Notice that the above theorem holds without any regularity assumption on the

agent value distributions.

3.8 Conclusion

In this chapter we have studied an online version of the bilateral trade problem, where unit demand buyers and sellers with Bayesian valuations are adversarially selected and presented in online fashion to an intermediary. We considered different types of markets, relative to the power of the intermediary and adversary, ranging from fully unconstrained, to having a limit on the number of items held in stock and finally to guaranteeing that the ratio of sellers and buyers arriving will be fixed.

Appendix

Lemma 3.9. *The matching computed using an online FIFO queue of size K, adding sellers while it's not full and popping them when a buyer is encountered, in the proof of Theorem 3.8 is a maximum one.*

Proof. The proof is for a finite K. For the general setting (where $K = \infty$) the proof is almost identical, with one fewer case. $K = \infty$. We use a potential argument. Let $\phi(t) = |I_{off}(t) - I_{on}(t)|^+$, where $|x|^+ = \max\{x, 0\}$. The potential indicates how many more items the offline algorithm has in stock. The intuition is that the offline cannot get an advantage without decreasing the potential, as for any successful trade he would lose an item. Let $R_{on}(t)$ be equal to 1 if the online sold an item to the t-th agent and 0 otherwise. We also define $R_{off}(t)$ for the offline algorithm. We analyse a number of cases:

- If the t-th agent is a seller:

$$R_{on}(t) - \phi(t) + \phi(t-1) = -|I_{off}(t) - I_{on}(t)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+.$$

If the online algorithm did not buy an item, then $I_{on}(t) = I_{on}(t-1) = K \geq I_{off}(t)$. Otherwise:

$$\begin{aligned}
&-|I_{off}(t) - I_{on}(t)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&\qquad = -|I_{off}(t) - (I_{on}(t-1) + 1)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&\qquad \geq -|I_{off}(t-1) + 1 - (I_{on}(t-1) + 1)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&\qquad = 0 = R_{off}(t).
\end{aligned}$$

- If the t-th agent is a buyer and the online algorithm sold an item:

$$\begin{aligned}
R_{on}(t) - \phi(t) + \phi(t-1) &= R_{on}(t) - |I_{off}(t) - I_{on}(t)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&= 1 - |(I_{off}(t-1) - R_{off}(t)) - (I_{on}(t-1) - 1)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&= 1 - |I_{off}(t-1) - I_{on}(t-1) + 1 - R_{off}(t)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&\geq 1 - |1 - R_{off}(t)|^+ \\
&\geq R_{off}(t).
\end{aligned}$$

- If the t-th agent is a buyer and $I_{on}(t-1) = 0$, so the online algorithm did not sell an item:

$$\begin{aligned}
R_{on}(t) - \phi(t) + \phi(t-1) &= R_{on}(t) - |I_{off}(t) - I_{on}(t)|^+ + |I_{off}(t-1) - I_{on}(t-1)|^+ \\
&= 0 - |(I_{off}(t-1) - R_{off}(t))|^+ + |I_{off}(t-1)|^+ \\
&= -|I_{off}(t-1) - R_{of}(t)|^+ + |I_{off}(t-1)|^+ \\
&\geq R_{off}(t).
\end{aligned}$$

In every case therefore, we have that $R_{on}(t) - \phi(t) + \phi(t-1) \geq R_{off}(t)$. Summing over the entire sequence we get

$$\sum_{t=1}^{n} R_{on}(t) \geq \sum_{t=1}^{n} R_{off}(t) + \phi(n) - \phi(0) \geq \sum_{t=1}^{n} R_{off}(t),$$

since $\phi(n) \geq \phi(0) = 0$. Clearly $\sum_{t=1}^{n} R_{on}(t) \leq \sum_{t=1}^{n} R_{off}(t)$, the online algorithm achieves a maximum sized matching. □

Chapter 4

Market Intermediation as a Secretary Problem

We study the complement of Chapter 3, where the adversary can chose the exact valuation of each agent, but their order of appearance is a uniformly random permutation. On a technical level, our work is more closely related to the secretary problem and online matching literature, but borrows the setting and the aspects of truthfulness and rationality from mechanism design. Similar to Chapter 3, we completely drop the budget balance requirement that is usually found in the bilateral trade literature.

4.1 Introduction

We study the problem of facilitating trade between n buyers and n sellers that arrive online. As before, every agent is unit-demand and all items are identical. We consider online algorithms that offer prices based on the sequence of past values and we assume that the online algorithm knows only the number of buyers and sellers, but not their values. The values of the sellers and buyers are selected *adversarially and are randomly permuted*. In that respect, the problem is a generalisation of the well-known secretary problem. The secretary problem corresponds to the special case in which

there are only buyers, the algorithm starts with a single item, and the objective is to maximise the total welfare, which is to give the value to a buyer with as high value as possible.

Extending this to both sellers and buyers, creates a substantially richer setting. One of the most important differences between the two settings is that besides the objective of maximising the total welfare, we now have the objective of maximising the gain-from-trade. For both objectives, the algorithm must buy from sellers with low values and sell to buyers with high values. The welfare of a solution is defined as the value of the buyers and sellers that have an item. The gain-from-trade of a solution is the difference between the welfare at the end of the process minus the welfare at the beginning. At optimality the two objectives are interchangeable: an algorithm achieves the maximum welfare if and only if it achieves the maximum gain-from-trade. But for approximate solutions, the two objectives are entirely different, with the gain-from-trade being the most demanding one.

A generalisation of our model is when the items are not identical and each buyer has different value for each one of them, i.e., each seller has a value for its item and each buyer has a vector of values, one for every pair buyer-seller. This is also a generalisation of the well-studied online maximum-matching problem [50, 43]. One can cast the online maximum-matching problem as the version in which the sellers arrive first and have zero value for their item. The optimal online algorithm for this problem has competitive ratio $1/e$, when the objective is the welfare (which in the absence of seller values is identical to the gain-from-trade). Our model is incomparable to the online maximum-matching problem: it is simpler in the sense that the items are identical (a single value for each buyer instead of a vector of buyer-item values), and at the same time more complicated in that the items are not present throughout the process, but they are brought to the market by sellers that have their own utility. The fact that in our model the buyer-item values are related, allows for a

much better competitive ratio regarding the welfare, (almost) 1 instead of $1/e$. More importantly, our algorithm is truthful, while in contrast, no good truthful algorithm is known for the online maximum-matching problem, which remains one of the main open problems of the area. On the other hand, the introduction of sellers poses new challenges, especially with respect to the objective of the gain-from-trade.

There are also similarities between our model and the extension of the classical secretary problem to k secretaries. From an influential result by Kleinberg [46] we know that this problem has competitive ratio $1 - 1/\sqrt{k}$ which is asymptotically tight, and can be transformed into a truthful algorithm. This result depends strongly on the knowledge of k. In our case the equivalent measure, the *number of trades* is not known from the beginning and has to be learned, with a degree of precision that is crucial, especially for the gain-from-trade objective. The fact that the gain-from-trade is not monotone as a function of time highlights the qualitative difference between the two models; the gain-from-trade temporarily *decreases* when the algorithm buys an item, with the risk of having a negative gain at the end. More generally, with the mix of buyers and sellers, wrong decisions are penalised more harshly and the monotone structure of the problem is disrupted.

4.1.1 Our results

We consider the case when both the number of buyers and the number of sellers is n. For the welfare objective we show a competitive ratio of $1 - \tilde{O}(n^{-1/3})$, where $\tilde{O}$ hides logarithmic factors.

Actually we can compare an online algorithm with two offline benchmarks: the *optimal* benchmark, in which all trades between buyers and sellers are possible, independently of their order of appearance, and the expected *sequential optimal* in which an item can be transferred from a seller to a buyer only if the seller precedes the buyer in the order.

Our online algorithm achieves a competitive ratio of $1 - \tilde{O}(n^{-1/3})$ against the optimal benchmark. To achieve this, it has a small sampling phase of length $\tilde{O}(n^{2/3})$ to estimate the *median* of the values of all traders, and then uses it as a price for the remaining traders. But if the optimal number of trades is small, such a scheme will fail to achieve competitive ratio almost one, because with constant probability there will not have enough items to sell to buyers with high value. To deal with this risk, the algorithm not only samples values at the beginning but it additionally buys sufficiently many items, $\tilde{O}(n^{2/3})$, from the first sellers[1]. The number $\tilde{O}(n^{2/3})$ of bought items balances the potential loss of the welfare that results from removing items from sellers to the expected loss from not having enough items for buyers of high values.

The term $O(n^{-1/3})$ in the competitive ratio seems to be optimal for a scheme that fixes the price after the sampling phase and relates to the number of items needed to approximate the median to a good degree. It may be possible to improve this term to $O(n^{-1/2})$ by a more adaptive scheme, as in the case of the k-secretary problem [46]. Finally, it may be possible to remove the logarithmic factors from the competitive ratio, but we have opted for simplicity and completeness.

For the objective of gain-from-trade, we give a truthful algorithm that has a constant competitive ratio, assuming that the algorithm starts with an item. The competitive ratio is high, approximately 10^3, but it drops to a small constant when the optimal number of trades is sufficiently high. The additional assumption of starting with an item is necessary, because without it, no online algorithm can achieve a bounded competitive ratio.

The main difficulty of designing an online algorithm for gain-from-trade is that even a single item that is left unsold at the end has dramatic effects on the gain-

[1]Buying from the first sellers cannot be done truthfully unless the algorithm knows an upper bound on their value. But this is not necessary since there is an alternative that has minor effects on the competitive ratio: the algorithm offers each seller the maximum value of the sellers so far. This is a truthful scheme that buys from all but a logarithmic number of sellers, in expectation.

from-trade. The online algorithm must deal with the case of many traders, plenty of welfare, but few optimal trades and small gain-from-trade.

To address this problem, our algorithm has a large sampling phase proportional to the input size, contrary to what's necessary for welfare. It uses this phase to estimate the number of optimal trades and two prices for trading with buyers and sellers. If the expected number of optimal trades is high, the algorithm uses the two prices for trading with the remaining traders. But if the number is small, it runs the secretary algorithm with the item that it starts with.

The analysis needs high concentration bounds on the expected number of trades to minimise the risk of having items left unsold. Our algorithm is ordinal, in the sense that it uses only the order statistics of the values instead the actual values themselves. This leaves little space for errors and it may be possible that cardinal algorithms that use the actual values can do substantially better.

4.1.2 The Secretary Problem and Related Work

The random order model we are using has its origins in the well-known secretary problem, where n items arrive in online fashion and our goal is to maximise the probability of selecting the most valuable, without knowing their values in advance. An adversary secretly selects the value of each item and then we observe them one by one, in a uniformly random permutation. At each step, we can select the item and end the game or irrevocably reject it and move to the next.

Proposition 5. *Let v be the highest value among the first $1/e \cdot n$ items. The algorithm that rejects these items and then selects the first which has value higher than v has probability $1/e$ of choosing the most valuable item.*

This algorithm is the best we can hope for. Actually, for the more 'relaxed' objective of maximising the expected value of the item obtained, the approximation

guarantee is still $1/e$, against an offline algorithm that always chooses the best item. There has been a spectacular amount of interest in variants of this setting, stemming chiefly from the matroid secretary problem was introduced by Babaioff et al. [7]. In this setting, we are allowed to select more than item, provided our final selection satisfies matroid constraints. A variety of different matroids have been studied, with many recent results presented by Dinitz in [29].

Of particular interest to our problem are secretary problems on bipartite graphs. Here, the left hand side vertices of the graph are fixed and the right hand side vertices (along with their incident) edges appear online. The selected edges must form a (incomplete) matching and the goal is to maximise the sum of their weights. Babaioff et al. in [7] provided a $4d$-competitive algorithm for the transversal matroid with bounded left degree d, which is a special case of the online bipartite matching where all edges connected to the same left hand side vertex have equal value. This was later improved to 16 by Dimitrov and Plaxton [28]. The case where all edges have unrelated weights was first considered by Korula and Pal in [50] who designed a 8-competitive algorithm, which was later improved to the optimal $1/e$ by Kesselheim et al. [43]. Combining the previous setting, [37] considers optimising the gains from trade in a two-sided market setting tailored to online advertising platforms, and the authors extend this idea further in [36] by considering two-sided markets in an online algorithm setting.

Another secretary variant which is close to our work is when the online selects k items instead of one, where Kleinberg [46] showed an asymptotically tight algorithm with competitive ratio $1 - O(\sqrt{1/k})$.

4.2 Model

The setting of the *random intermediation* problem consists of sets $B = \{b_1, \ldots, b_n\}$ and $S = \{s_1, \ldots, s_n\}$ containing the valuations of the buyers and sellers. For convenience, we assume that they are all distinct. The intermediary interacts with a uniformly random permutation σ of $B \cup S$ which is presented to him one agent at a time, over $2n$ steps. The intermediary has no knowledge of $\sigma(t)$ before step t. We use b^i and s^j to denote the i-th *highest* valued seller and j-th *lowest* valued seller respectively.

We study *posted price* mechanisms that upon seeing the identity of agent t offer price p_t. This price can not depend on the entire valuation function; only the values within $\sigma(1) \ldots \sigma(t-1)$ which are revealed at this point. The algorithm interacts with the agents in exactly the same way as Chapter 3 and the notation is the same. The gain from trade (or GFT) produced by algorithm A throughout the run is the difference between the final and starting welfare:

$$GFT_A(S, B) = \mathbb{E}\left[\sum_{b \in T_B} b - \sum_{s \in T_S} s\right]$$

.

We are interested in the *competitive ratio* of our online algorithm A compared to the offline algorithm OPT. In this setting there are two different offline algorithms to compare against: optimal offline and sequential offline. They both know S, B, but the first can always achieve the maximum welfare, whereas the second operates under the same constrains as we, namely he can only perform trades permitted by σ, which is unknown. We say that algorithm A is ρ-competitive for welfare (or gain from trade) if for any S, B we have:

$$\mathcal{W}_A(S, B) \geq \rho \cdot \mathcal{W}_{\mathrm{OPT}}(S, B) - \alpha, \tag{4.1}$$

for some fixed $\alpha \geq 0$.

Often we will refer to the *matching* between a set of buyers and a set of sellers. Let $M(S, B) = \{\{S_1\} \cup \{B_1\}\}$, where $S_1 \subseteq S, B_1 \subseteq B$ is the set of sellers and buyers with whom we trade (or are matched, in the sense that the items move from sellers to buyers) in a welfare maximising allocation and $m(S, B)$ the optimal gain from trade. Note that this does *not* contain pairs: only the set of each side of the matching. Similarly, let $M(S, B, q, p)$ be the matching generated by only trading with sellers valued below q and buyers above p. In a slight abuse of notation, we will use $|M(S, B)| = |S_1|$ for the size of the matching and $M(S, B) \star M(S', B') = \{\{S_1 \star S_1'\} \cup \{B_1 \star B_1'\}\}$, where $\star$ is any set operation. For convenience, we refer to $M(\sigma) = M(\{s \mid s \in \sigma\}, \{b \mid b \in \sigma\})$ where σ is a sequence of agents.

4.3 Welfare

In order to approximate the welfare, the online algorithm uses a sampling phase to find the median price, in an attempt to transfer items from agents below the median to more valuable ones above it. The two main challenges, in terms of its performance, are estimating the median with a small sample and not missing too many trades due to the online nature of the input. Before we delve into the actual algorithm, it is useful to state two probability concentration results, similar to the familiar Azuma-Hoeffding inequality, but for the setting where sampling happens *without* replacement as is our case.

Lemma 4.1. *Let multiset $\mathcal{X} = \{x_1, \ldots, x_N\}$ where $x_i \in \{0, 1\}$, $x_1 = x_2 = \ldots = x_m = 1$ and $x_{m+1} = \ldots = x_N = 0$ for some integer $m \geq 0$. Consider sampling n values of $\mathcal{X}$ uniformly at random* ***without*** *replacement and let X_i be the value of the*

$i-th$ draw. For $Y = \sum_{i=1}^{n} X_i$, we have that for any $\epsilon > 0$:

$$\Pr[Y \geq (1+\epsilon)\,\mathbb{E}[Y]] \leq e^{-2\epsilon^2 \max\{m,n\}\frac{mn}{N^2}} \tag{4.2}$$

and

$$\Pr[Y \leq (1-\epsilon)\,\mathbb{E}[Y]] \leq e^{-2\epsilon^2 \max\{m,n\}\frac{mn}{N^2}}. \tag{4.3}$$

Proof. Let $Y_i = \mathbb{E}[Y|X_1, \ldots, X_i]$ be the Doob martingale of Y, exposing the choices of the first i draws. Clearly we have that $|Y_{i+1} - Y_i| \leq 1$, since the knowledge of one draw cannot change the expectation by more than 1. Applying Azuma's inequality, we obtain:

$$\Pr[Y_n - Y_0 \geq t] \leq e^{\frac{-t^2}{n}}. \tag{4.4}$$

Let Z_j for $1 \leq j \leq m$ indicate if x_j was chosen. Since only these x_j contribute to Y, we have that $Y = \sum_{i=1}^{m} Z_i$. Repeating the previous martingale construction, we get:

$$\Pr[Y_m - Y_0 \geq t] \leq e^{\frac{-t^2}{m}}. \tag{4.5}$$

But, we know that $Y_0 = \mathbb{E}[Y] = \mathbb{E}[\sum_{i=1}^{m} Z_i] = m\frac{n}{N}$. Setting $t = \epsilon m \frac{n}{N}$ in both (4.4) and (4.5) and using $Y_n = Y_m = Y$ we obtain:

$$\Pr[Y \geq (1+\epsilon)\,\mathbb{E}[Y]] \leq e^{-2\epsilon^2 \max\{m,n\}\frac{mn}{N^2}}. \tag{4.6}$$

Concentration in the opposite direction is found by repeating the same analysis, using the complementary form of Azuma's inequality. □

Note that this result is not superfluous: by immediately applying Hoeffding's inequality for sampling with replacement, we would obtain:

$$\Pr[Y \geq (1+\epsilon)\,\mathbb{E}[Y]] \leq e^{-2\epsilon^2 \frac{m^2 n}{N^2}},$$

which is only tight if m is large *compared to* N. The concentration should intuitively work if n is a large fraction of N as well: imagine $n = N$.

Similarly, we often encounter a situation where we are interested in the number of trades between n sellers and n buyers, arriving in a uniformly random permutation. Assuming we buy from all sellers, occasionally we would encounter a buyer without having any items at hand. This result shows that even though this is the case, few trades are lost.

Lemma 4.2. *The number of trades $M(\sigma)$, where σ is a uniformly random sequence containing n buyers and n sellers, is:*

$$\mathbb{E}\left[M(\sigma)\right] \geq \frac{n-1}{n}\left(n - \sqrt{2n\log n}\right), \tag{4.7}$$

assuming all sellers are valued below all buyers.

Proof of Lemma 2. Since we buy from all sellers and attempt to sell to all buyers, let X_t be 1 if at step t a seller is encountered and -1 if it is a buyer. We define the following martingale, with $Y_0 = 0$:

$$Y_{t+1} = \begin{cases} Y_t + X_t & \text{if } Y_t > 0 \\ Y_t - X_t & \text{if } Y_t < 0 \\ \begin{cases} 1 & \text{with probability } \frac{1}{2} \\ -1 & \text{with probability } \frac{1}{2} \end{cases} & \text{if } X_t = 1 \text{ and } Y_t = 0 \\ Y_t & \text{otherwise} \end{cases}.$$

Basically, Y_t keeps track of the unsold items: sellers pull away from 0 and buyers

towards 0. The actual number of unsold items at time t is $\sum_{i=1}^{t} X_i$. We need to define Y_t since $\sum_{i=1}^{t} X_i$ is not a martingale: $\mathbb{E}\left[\sum_{i=1}^{t+1} X_i \mid \sum_{i=1}^{t} X_i\right] > 0$, as we cannot sell items when the stock is empty. However, Y_t could also be negative, circumventing this difficulty. Inductively, it is easy to show that $|Y_t| = \sum_{i=1}^{t} X_i$. Therefore, the number, of unsold items at time t is at most $|Y_t|$. By a simple case analysis we have that $|Y_{t+1} - Y_t| < 1$. Thus, by Azuma's inequality we have:

$$\Pr[|Y_{2n} - Y_0| \geq \sqrt{2n \log n}] \leq e^{-2\frac{4n \log n}{2 \cdot 2n}} = \frac{1}{n}. \tag{4.8}$$

Since n items are bought, we have:

$$\mathbb{E}\left[M(\sigma)\right] \geq \frac{n-1}{n}\left(n - \sqrt{2n \log n}\right) \tag{4.9}$$

□

All the machinery is now in place analyse sequential algorithms in this setting. We first show a key property of the offline algorithm.

Proposition 6. *The optimal offline algorithm sets a price p, equal to the median of all the agents' valuations and trades items from sellers valued below p to buyers valued above p.*

Proof. Since there are only n items available, if we could freely redistribute the items we would choose the top n agents with highest valuations. Let p be the value of the n-th most valuable agent. If there are k buyers valued above p we have $n - k$ buyers and k sellers valued below it. Thus, buying from all sellers below p and selling to all buyers above it is an optimal algorithm. □

However, the optimal sequential offline algorithm would not just trade at this price. For instance, if there is 1 buyer and $n-1$ sellers above p and 1 seller and $n-1$ buyers below, trading at this price would give a $1/2$ probability of transferring the

item, since only one transfer increases the welfare and the agents have to appear in the right order. Therefore, if that buyer has a much larger valuation than anyone else, this algorithm would only be 1/2-competitive. However, we can modify this approach with a bias towards buying more items than needed, in order to maximise the probability of finding high valued buyers.

Lemma 4.3. *The optimal sequential online algorithm is* $\left(1 - O\left(\frac{\log n}{n^{1/3}}\right)\right)$*-competitive against the optimal offline for welfare.*

Proof. The optimal online algorithm adjusts the price p according to the following two cases, where $M = |M(S, B)|$.

1. $M \geq n^{2/3}$. In this case the same price p is used. At the end, the online algorithm will still keep the highest valued $n - M$ sellers and by Lemma 4.2 will match

$$\frac{M-1}{M}\left(M - \sqrt{2M \log M}\right)$$

buyers in expectation. The offline optimum will of course keep the highest $n-M$ sellers and M buyers, leading to a competitive ratio of at most:

$$\frac{M-1}{M}\left(1 - \frac{\sqrt{2 \log M}}{\sqrt{M}}\right) = 1 - O\left(\frac{\log n}{n^{1/3}}\right).$$

2. $M < n^{2/3}$.

   In this case, suppose two prices are used: p_S to buy from the lowest $n^{2/3}$ sellers and p_B to sell to the highest $n^{2/3}$ buyers. For the buyers, the online does at least as well as the previous case. In particular, it obtains a uniformly random sample of size at least $n^{2/3} - \sqrt{2n^{2/3} \log n^{2/3}}$ by Lemma 4.2, amongst the top $n^{2/3}$ buyers with probability at least $(n^{2/3} - 1)/n^{2/3}$. Since the M buyers matched by the optimal offline are contained within the highest $n^{2/3}$ buyers, the ratio just from buyers remains the same as before.

From the sellers side, the online keeps the highest $n - n^{2/3}$ sellers, while the offline keeps at most n, for a ratio at most $1 - 1/n^{1/3}$.

Combining both cases, the ratio is asymptotically at most:

$$1 - O\left(\frac{\log n}{n^{1/3}}\right). \tag{4.10}$$

Note that the choice of $n^{2/3}$ to separate the two cases is optimal. □

The next step is to design an online algorithm without knowing p or $|M(S, B)|$ beforehand. The algorithm is as follows:

1. Record the first $8n^{2/3} \log n$ agents and calculate their median p'. Buy from all sellers during this sampling phase.

2. After the sampling starts the trading phase:

   (a) Buy from seller s if $s \leq p'$.

   (b) Sell to buyer b is an item is available and $b \geq p'$.

For the analysis of this algorithm, we first need a concentration result on the sample median p'.

Lemma 4.4. *Let $X = \{1, \ldots, 2n\}$ and select $8n^{2/3} \log n$ elements from X* without *replacement. Then, their sample median M satisfies:*

$$\Pr[|M - n| \geq n^{2/3}] \leq O\left(\frac{1}{n}\right). \tag{4.11}$$

Proof. We have that:

$$\Pr[M \geq n + n^{2/3}] = \Pr[\text{more than } 4n^{2/3} \log n \text{ elements picked no less than } n + n^{2/3}]$$

Since we are sampling without replacement, this is equivalent to selecting $8n^{2/3}\log n$ elements uniformly at random from X' containing $n+n^{2/3}$ 0's and $n-n^{2/3}$ 1's and having their sum be greater than $4n^{2/3}\log n$. Using X', $\epsilon = n^{2/3}/(n-n^{2/3})$ and taking $8n^{2/3}\log n$ samples in Lemma 4.1, we have:

$$\begin{aligned}\Pr[M \geq n+n^{2/3}] &= \Pr[Y \geq (1+\epsilon)\,\mathbb{E}\,[Y]] = \Pr[Y \geq n^{2/3}] \\ &\leq \exp\left(-2\left(\frac{n^{2/3}}{(n-n^{2/3})}\right)^2 (n-n^{2/3})\frac{8n^{2/3}\log n(n-n^{2/3})}{4n^2}\right) \\ &\leq O\left(\frac{1}{n}\right).\end{aligned}$$

By symmetry, the same holds for $\Pr[M \leq n-n^{2/3}]$: just reverse the ordering of the agents. □

This shows that our sample median p' might have at most $n^{2/3}$ agents more on one side compared to the true median p. However, this loss is negligible asymptotically, as these agents are a uniformly random subset of the $S \cup B$. We now show that buying from sellers during the sampling phase, before considering any buyers, can only increase the number of trades in the next phase.

Lemma 4.5. *Let σ be a sequence containing n buyers and n sellers. Move an arbitrary seller the beginning of the sequence to obtain $s\sigma'$. Then we have:*

$$|M(s\sigma')| \geq |M(\sigma)|.$$

Proof. Let b' be the first buyer not to receive an item in σ. Clearly, if b' doesn't exist then the number of items sold in both cases is n. Assume we sell the item bough from s only if it is the last item left. Then, it is sold to b': otherwise b' would not be the first buyer not to be sold an item is σ. There are two cases:

1. If s appears in σ before b': both sequences continue identically as we have no

items in stock after b'.

2. If s appears after b' in σ: there is one fewer seller in $s\sigma'$ after b', since s was moved to the front. However, this can result in at one lost sale.

□

Actually, we have shown that moving sellers to the beginning can only increase trades, which is slightly more powerful. We are now ready to state one of the main results of this chapter.

Theorem 4.1. *This algorithm is* $\left(1 - \tilde{O}\left(\frac{1}{n^{1/3}}\right)\right)$*-competitive for welfare.*

Proof. As before, let $M = |M(S, B)|$ be the size of the optimal offline matching. The following analysis assumes that the event of Lemma 4.4 did not occur and p and p' split the agents in two sets, differing by at most $n^{2/3}$. Given this, we analyse the algorithm in three steps. First show that we never buy too many items from highly valued sellers, therefore we keep most of the sellers' contribution to the final welfare. Then we show that we always match a high proportion of the valuable buyers by considering two cases: if there are few such buyers then they are matched to the sellers we obtained during the sampling phase, otherwise we have enough sellers below p' to match them to.

We introduce some notation useful to the analysis: let W be the set containing the top $n - n^{2/3}$ highest valued agents. Then let S_W, B_W be the number of sellers and buyers respectively in W and S'_W, B'_W be how many of them appeared after the sampling phase. valued agents. To show the competitiveness of our algorithm, it suffices to find the fraction of W that is achieved at the end of the sequence: being $(1 - \tilde{O}(1/n^{1/3}))$-competitive against the top $n - n^{2/3}$ agents implies a ratio of

$$\left(1 - \tilde{O}(1/n^{1/3})\right) \cdot \frac{n - n^{2/3}}{n} = 1 - \tilde{O}(1/n^{2/3})$$

against all n agents above the median and therefore the optimal offline.

We first show that we never lose too much welfare by buying from sellers, both in the sampling and trading phase. Given p', the only occasion on which a seller in W is bought is if he is amongst the first $8n^{2/3}\log n$ sellers. This event is clearly independent from the condition on p', meaning in expectation we keep

$$\mathbb{E}\left[S'_W\right] = S_W\left(1 - \frac{8n^{2/3}\log n}{n}\right) = S_W\left(1 - \frac{8\log n}{n^{1/3}}\right) \tag{4.12}$$

highly valued sellers. Therefore, enough of the sellers' original value is kept. The rest of the analysis will only focus *only* the proportion of buyers in W who get an item. For the number of items I_S bought during the sampling phase, the following holds by Lemma 4.1:

$$\Pr\left[I_S \leq (1-\frac{1}{2})4n^{2/3}\log n\right] \leq e^{-2\frac{1}{4}8n^{2/3}\log n\frac{n^2}{4n^2}} \leq e^{-n^{2/3}}, \tag{4.13}$$

as there are n out of $2n$ agents are sellers and we sample $8n^{2/3}\log n$ of them. Therefore, we enter the trading phase with an excess of at least $2n^{2/3}\log n$ items with high probability.

To analyse the number of buyers in W matched, we consider two cases.

$B_W \leq n^{2/3}\log n$: In this case there are few valuable buyers and all we need to show is that the excess of items bought during sampling is enough to trade with most of them. We first need to find $\mathbb{E}\left[B'_W\right]$, which is slightly more complicated, since we have conditioned on p' approximating the median. Given p', at least $4n^{2/3}\log n$ agents were above the median value during the sampling phase. Note that all of the agents in W are above the median. Therefore, any of the agents in the upper $4n^{2/3}\log n$ half

of the sampling phase could be replaced by a buyer in W. At worst, $4n^{2/3}\log n$ agents from W are in the sampling phase, which means that in a random permutation, we have:

$$n^{2/3}\log n \geq \mathbb{E}\left[B'_W\right] \geq B_W\left(1 - \frac{4n^{2/3}\log n}{n - n^{2/3}}\right).$$

We might also consider up to $n^{2/3}$ extra buyers, if p' underestimated p. However, given that $I_S \geq 2n^{2/3}\log n$ with high probability, every buyer in B'_W will be matched with an item, giving the claimed competitive ratio for this case.

$\boldsymbol{B_W > n^{2/3}\log n}$**:** Let $k \geq B_W$ be the number of trades the optimal offline algorithm would perform. Since the median might be underestimated, the number of sellers we consider is at least $k - n^{2/3}$ and buyers at most $k + n^{2/3}$. We show that, with the help of the extra items we bought during sampling, we have more items than buyers in total, with high probability. Let $S_{p'}, B_{p'}$ the number of sellers and buyers below and above p' *after* the sampling phase. By by Lemma 4.1 we expect to find

$$\Pr[S_{p'} \leq (1 - \sqrt{\frac{\log k}{k}})(k - n^{2/3})\frac{2n - 8n^{2/3}\log n}{2n}] \tag{4.14}$$

$$\leq \exp\left(-2\frac{\log k}{k}(k - n^{2/3})\frac{(2n - 8n^{2/3}\log n)^2}{4n^2}\right) \leq O\left(\frac{1}{k}\right), \tag{4.15}$$

by sampling $2n - 8n^{2/3}\log n$ out of $2n$ with $k - n^2/3$ important elements. Similarly we have

$$\Pr[B_{p'} \geq (1 + \sqrt{\frac{\log k}{k}})(k + n^{2/3})\frac{2n - 8n^{2/3}\log n}{2n}] \leq O\left(\frac{1}{k}\right). \tag{4.16}$$

It is important to note that these quantities are almost equal, other than a $n^{2/3}$ factor which is insignificant compared to k. Then, with high probability:

$$B_{p'} - S_{p'} \leq (1 + \sqrt{\frac{\log k}{k}})(k + n^{2/3}) - (1 - \sqrt{\frac{\log k}{k}})(k - n^{2/3}) \tag{4.17}$$

$$\leq 2(\sqrt{k \log k} + n^{2/3}) \tag{4.18}$$

$$\leq 3(\sqrt{n \log n}), \tag{4.19}$$

given than $k \leq n$. Since we bought at least $2n^{2/3} \log n$ items during the sampling phase, the *total* number of items bought is higher than the total number of buyers considered for n large enough. Also, by Lemma 4.5 having these items ready before encounter buyers is beneficial.

Therefore, we get a lower bound on the number of buyers in $B_{p'}$ that actually acquire an item using Lemma 4.2. The number of items sold in expectation is at least:

$$\mathcal{M} \geq \frac{B_{p'} - 1}{B_{p'}} \left(B_{p'} - \sqrt{2B_{p'} \log B_{p'}} \right). \tag{4.20}$$

However, we are interested only in the fraction of buyers in B_W who acquired an item. The algorithm does not differentiate between any buyer above p', the sequence is uniformly random and all buyers in B_W are contained within the top $k + n^{2/3}$ buyers. By lower bounding $B_{p'}$ with Lemma 4.1:

$$\Pr[B_{p'} \leq (1 - \sqrt{\frac{\log k}{k}})(k + n^{2/3}) \frac{2n - 2n^{2/3} \log n}{2n}] \leq O\left(\frac{1}{k}\right), \tag{4.21}$$

and using (4.20), the fraction of buyers in B_W matched is at least:

$$\frac{\mathcal{M}}{k + n^{2/3}} \geq 1 - \tilde{O}\left(\frac{1}{n^{2/3}}\right), \tag{4.22}$$

with probability $1 - O(1/k)$, which is asymptotically high as $k \geq n^{2/3} \log n$. □

4.4 Gain from Trade

Compared to the welfare, the gain from trade is a more challenging objective. The main reason is that even for large n, the actual trades that maximise the GFT can be very few and quite well hidden. Moreover, buying from a single seller and being unable to sell could completely shatter the GFT, while it could have very little effect on the welfare.

First of all, the setting has to be slightly changed. We give the online algorithm one extra, free item at the beginning to ensure that at least one buyer can acquire an item, even when the initial sampling has been inconclusive. For fairness, the offline algorithm is also provided with this starting item. We show that this modification is absolutely necessary to study this setting under competitive analysis.

Theorem 4.2. *Starting with no items, there exist S, B such that the competitive ratio for the GFT is arbitrarily high.*

Proof. Consider two different valuations. The first has $s_1 = c > 0$ and $b_1 = c + \epsilon$. In the second has $\hat{s}_1 = c, \hat{b}_1 = c - \epsilon, \hat{s}_2 < \hat{b}_1 - \epsilon'$. We tweak the value of the buyer so that the trade from instance one no longer increases welfare, but add one extra seller to keep the optimal GFT positive.

Let $p = \Pr[s_1 \in T_S \mid \sigma^{-1}(b_1) > \sigma^{-1}(s_1)]$ be the probability of the online algorithm buying from s_1, conditioned on b_1 arriving later. This must be $p > 0$, otherwise his expected gain from trade will be 0, compared to the $\epsilon/2$ generated by the offline.

However, in the second instance the algorithm should buy from $\hat{s}_2$ instead of $\hat{s}_1$. But, if $\hat{s}_1$ appears first, the first algorithm should buy from him too, as the information

received so far is the same:

$$\begin{aligned}\hat{p} &= \Pr[\hat{s}_1 \in T_S \mid \sigma^{-1}(\hat{b}_1) > \sigma^{-1}(\hat{s}_1) \wedge \sigma^{-1}(\hat{s}_2) > \sigma^{-1}(\hat{s}_1)] \\ &\geq p \Pr[\sigma^{-1}(\hat{s}_2) > \sigma^{-1}(\hat{s}_1)] \\ &= \frac{p}{2} > 0.\end{aligned}$$

So the online algorithm has a positive chance of buying the item from the wrong seller. Assuming in all other cases maximum gain from trade is extracted, we have:

$$GFT_A(S, B) \leq \hat{p}(-\epsilon) + (1 - \hat{p})(\epsilon' - \epsilon). \tag{4.23}$$

Since $\hat{p}$ is independent of ϵ, ϵ', we can set

$$\epsilon' = \frac{\epsilon}{1 - \hat{p}}$$

which leads to $GFT_A(S, B) = 0$ whereas the offline has $\epsilon/2$.

In any case, no online algorithm can perform well in both instances. □

To avoid the previous pitfall, we assume the intermediary starts with one item. Roughly, the algorithms starts by estimating the total volume of trades in an optimal matching by observing the first segment of the sequence. Using this information, two prices $\hat{p} \geq \hat{q}$ are computed, to be offered to agents in the second part.

Remark 3 (Ordinal Mechanism)**.** *Even though this algorithm posts prices, it in fact only cares about the relative order of the agent valuations, but not their exact values. A* cardinal *mechanism could improve on the competitive ratio, by taking into account the welfare generated by the matching as well as its size.*

This being an ordinal mechanism, the goal is to maximise the number of trades *and leave no item unsold*. During the trading phase we are also much more conserva-

tive: at most one item is kept in stock and we stop buying items well before the end of the sequence, to make sure that there are enough buyers left to sell everything. The online algorithm CAUTIOUS(c, ϵ, N) contains parameters whose values will be specified later. Intuitively, $c \in [0, 1]$ determines the fraction of the input used for sampling, $\epsilon \in [0, 1]$ is the tolerance of the estimated parameters and N is a safeguard to ensure that the sampling phase produced a matching large enough to convey meaningful information.

Input: A sequence σ of length $2n$, appearing online.
Output: A matching between buyers and sellers.
With probability $\frac{1}{2}$ ignore sellers and sell the item as in the normal secretary, otherwise continue ;
Split the sequence into two segments such that $\sigma = \sigma_1\sigma_2$, with $|\sigma_1| = c \cdot 2n$;
Let S_1, B_1 denote the sellers and buyers of σ_1;
Calculate the welfare maximising matching $M(S_1, B_1)$;
if $|M(S_1, B_1)| \leq N$ **then**
    Sell the item to the highest remaining buyer (increasing the sample phase if necessary) as in the normal secretary problem and stop;
end
Set $\hat{p}, \hat{q}$ which only keep $(1 - \epsilon) \cdot c \cdot |M(S_1, B_1)|$ many matched pairs;
$i \leftarrow c \cdot 2n$;
$k \leftarrow \emptyset$;
$M \leftarrow \emptyset$;
`/* For the first half of` σ_2`, buy and sell items, keeping at most one in stock */`
while $i \leq c \cdot 2n + (1 - c) \cdot 2n/2$ **do**
    if $\sigma(i)$ *is a seller,* $k = \emptyset$ *and* $\sigma(i) \leq \hat{q}$ **then**
        $k \leftarrow \sigma(i)$;
    end
    if $\sigma(i)$ *is a buyer,* $k \neq \emptyset$ *and* $\sigma(i) \geq \hat{p}$ **then**
        Sell to $\sigma(i)$;
        $k \leftarrow \emptyset$;
    end
    $i \leftarrow i + 1$;
end
For the second half of σ_2, just try to sell the last remaining item, if any;

The idea is to use the first part of the sequence to estimate the matching $M(S, B)$.

If a large (in terms of pairs) GFT maximising matching is observed, it is likely that a proportionate fraction of it will be contained in the second half. In that case, sellers and buyers are matched in non overlapping pairs to avoid buying too many items. However, if the observed matching is too small, then the algorithm defaults to selling only the starting item, as it is very likely that σ_2 will not contain enough buyers for anything more.

Before moving on to the analysis of the CAUTIOUS(c, ϵ, N), we need a simple lemma on the structure of GFT maximising matchings, to explain the prices set.

Lemma 4.6. *For any S, B and $S_1 \subseteq S, B_1 \subseteq B$:*

1. *$m(S, B)$ can be obtained by setting two threshold prices p, q and trading with buyers above and sellers below them.*

2. *Choosing $\hat{p} > p$ and $\hat{q} < q$ such that $|M(S, B, \hat{q}, \hat{p})| \geq \alpha |M(S, B)|$ for $\alpha < 1$ yields $m(S, B, \hat{q}, \hat{p}) \geq \alpha m(S, B)$.*

3. *$|M(S, B)| \geq |M(S_1, B_1)|$ and $m(S, B) \geq m(S_1, B_1)$.*

Proof. For Property 1, assume $s < b < \hat{s} < \hat{b}$, such that $s, b, \hat{s}, \hat{b} \in M(S, B)$. But, instead of two matches we can just match $\hat{b}$ to s instead: $\hat{b} - s > b - s + \hat{b} - \hat{s}$, thus any such pair of matched agents cannot be part of $M(S, B)$. Setting $q = \max\{s \in M(S, B)\}$ and $p = \min\{b \in M(S, B)\}$ we have $q < p$ and the result follows. This is essentially the same observation as using the median price to trade, but using two different prices for robustness, as we will see later.

Property 2 follows because $M(S, B, \hat{q}, \hat{p})$ contains the α highest value pairs for $M(S, B)$. Property 3 is straightforward. □

Theorem 4.3. CAUTIOUS$(c = 0.3, \epsilon = 0.2758, N = 114)$ *is $O(1)$-competitive for the gain from trade.*

Proof. Let $z = |M(S, B)|$. We bound the gain from trade for the case where σ_1, σ_2 contain their analogous proportion of $M(S, B)$ and show that the losses are insignificant otherwise. In particular, let

$$f(c, \epsilon, z)$$
$$= \Pr\left[\frac{M(S,B) \cap M(S_1, B_1)}{M(S,B)} \geq c(1-\epsilon) \wedge \frac{M(S,B) \cap M(S_2, B_2)}{M(S,B)} \geq (1-c)(1-\epsilon)\right]$$

be the *well mixed* probability, where an ϵ-approximate chunk of the matching appears in both parts. The two events are not independent. To bound $f(c, \epsilon, z)$, it suffices to study the distribution of $S_M = \{s \in M(S, B)\}$ and $B_M = \{b \in M(S, B)\}$, the sets of agents comprising the optimal matching. By Lemma 4.6, we know that any seller in S_M can be matched to any buyer in B_M. Since we only care about the *size* of the matching in σ_1 and σ_2, not its actual value, we can rewrite $f(c, \epsilon, z)$ as:

$$f(c, \epsilon, z) = \Pr\left[\frac{|S_M \cap S_1|}{|S_M|} \geq c(1-\epsilon) \wedge \frac{|B_M \cap B_1|}{|B_M|} \geq c(1-\epsilon) \wedge \right. \tag{4.24}$$
$$\left. \frac{|S_M \cap S_2|}{|S_M|} \geq (1-c)(1-\epsilon) \wedge \frac{|B_M \cap B_2|}{|B_M|} \geq (1-c)(1-\epsilon)\right], \tag{4.25}$$

which is easier to handle.

It is useful to think the input as being created in two steps: first the *volume* of agents in S_1, B_1, S_2, B_2 is chosen and *afterwards* their exact values are randomly assigned. As such, a lower bound on the fraction of the size of the online to the offline matching provides the same bound on the gain from trade. We begin by bounding $f(c, \epsilon, z)$.

Lemma 4.7. *The probability the matching is well-mixed is*

$$f(c, \epsilon, z) \geq 1 - 2(e^{-2\epsilon^2 z c^2} + e^{-2\epsilon^2 z (1-c)^2})$$

Proof. Continuing from (4.24) we have:

$$
\begin{aligned}
f(c,\epsilon,z) = \Pr\Bigg[& \frac{|S_M \cap S_1|}{|S_M|} \geq c(1-\epsilon) \wedge \frac{|B_M \cap B_1|}{|B_M|} \geq c(1-\epsilon) \wedge \\
& \frac{|S_M \cap S_2|}{|S_M|} \geq (1-c)(1-\epsilon) \wedge \frac{|B_M \cap B_2|}{|B_M|} \geq (1-c)(1-\epsilon) \Bigg] \\
& \geq 1 - \Pr\left[\frac{|S_M \cap S_1|}{|S_M|} \leq c(1-\epsilon)\right] - \Pr\left[\frac{|B_M \cap B_1|}{|B_M|} \leq c(1-\epsilon)\right] - \\
& \Pr\left[\frac{|S_M \cap S_2|}{|S_M|} \leq (1-c)(1-\epsilon)\right] - \Pr\left[\frac{|B_M \cap B_2|}{|B_M|} \leq (1-c)(1-\epsilon)\right] \quad (4.26) \\
& \geq 1 - 2(e^{-2\epsilon^2 z c^2} + e^{-2\epsilon^2 z (1-c)^2}), \quad (4.27)
\end{aligned}
$$

where (4.26) follows by taking the complement and a union bound and (4.27) by applying Lemma 4.1 individually for each event. □

Let p and q be the prices achieving the matching $M(S,B)$, by Lemma 4.6. We need to show that the prices $\hat{p}, \hat{q}$ computed achieve a constant approximation of $m(S_2, B_2)$. Since $M(S,B)$ is well mixed and by using Lemma 4.6 we have that:

$$
|M(S,B)| \geq |M(S_1,B_1)| \geq |M(S_1,B_1,q,p)| \geq (1-\epsilon)\cdot c \cdot |M(S,B)|, \quad (4.28)
$$

where the second inequality holds since $M(S_1,B_1)$ is a gain from trade maximising matching and the third because at least a $(1-\epsilon)\cdot c$ fraction of $M(S,B)$ appeared in σ_1. In particular, we have that $M(S_1,B_1,q,p) \subseteq M(S_1,B_1)$ is the highest value part of $M(S_1,B_1)$ and $M(S_1,B_1,\hat{q},\hat{p}) \subseteq M(S_1,B_1,q,p)$, thus $\hat{q} \leq q$ and $\hat{p} \geq p$ leading to:

$$
|M(S_1,B_1,\hat{q},\hat{p})| \geq (1-\epsilon)^2 c^2 |M(S,B)| \quad (4.29)
$$

by eq. (4.28). Therefore, the prices $\hat{p}, \hat{q}$ computed find a relatively large *subset* of $M(S,B)$. We now need to find just how many of the trades in $M(S_2,B_2,\hat{p},\hat{q})$ are achieved by our algorithm. Let $\hat{S}_2 = \{s \mid s \in S_2 \wedge s < \hat{q}\}$ and $\hat{B}_2 = \{b \mid b \in B_2 \wedge b > \hat{p}\}$.

We need a high probability guarantee on the size of $\hat{S}_2$ and $\hat{B}_2$.

Lemma 4.8. *Assuming the matching is well mixed:*

$$\Pr\left[|\hat{S}_2| \geq ((1-c)(1-\epsilon) - \frac{1}{2})|S_M|\right] \geq 1 - 2^{-c^2(1-\epsilon)^2|S_M|}.$$

Proof. In the well-mixed case, we have that

$$|S_2 \cap S_M| \geq (1-c)(1-\epsilon)|S_M| \text{ and } |S_1 \cap S_M| \geq c(1-\epsilon)|S_M| \tag{4.30}$$

which leads to

$$|S_1 \cap S_M| \leq (1-(1-c)(1-\epsilon))|S_M|. \tag{4.31}$$

To get a lower bound on the size, we have:

$$\begin{aligned}
\Pr\left[|\hat{S}_2| \geq \frac{|S_M|}{2} - |S_1 \cap S_M|\right] &\geq \Pr\left[|\hat{S}_2| \geq ((1-c)(1-\epsilon) - \frac{1}{2})|S_M|\right] && (4.32)\\
&\geq \Pr\left[\hat{q} \geq \text{median}(S_M)\right] && (4.33)\\
&\geq 1 - 2^{-c^2(1-\epsilon)^2|S_M|} && (4.34)
\end{aligned}$$

where eq. (4.32) follows from eq. (4.30). eq. (4.33) follows since if $\hat{q}$ is greater than the median, then at worst case all elements from $S_1 \cap S_M$ are less than $\hat{q}$, which still leaves plenty of sellers in S_2. eq. (4.34) follows since draws are not actually independent, but this works in the inequality's favour. From eq. (4.29) we know $\hat{q}$ is greater than at least a $c^2(1-\epsilon)^2$ fraction of sellers. Since the 'bad case' is choosing all sellers below the median, this happens with higher probability if each draw is *with* rather than *without* replacement, leading to the result. □

Clearly, Lemma 4.8 holds for buyers as well. The proof is almost identical, keeping in mind that buyers are ordered the opposite way.

At this point we have a clear indication of how many sellers and buyers the prices $\hat{p}, \hat{q}$ cover in the second part of the sequence. Since this is an ordinal mechanism, we want to maximise the number of trades *provided no item is left unsold*. There are no a priori guarantees on the welfare increase of each trade, even a single unsold item ruins our gain from trade guarantees, in the worst case.

Lemma 4.9. *Let $A = |M(S_2, B_2, \hat{q}, \hat{p})|$ and $B = |S_2| + |B_2| - A$. Then, the probability that no item is left unsold is at least $1 - 2^{-A}$. Moreover, the expected number of trades in this case is at least*[2]*:*

$$\frac{\frac{A+B/2-1}{2A+B-1} \cdot \frac{A}{2} - \frac{A}{2^A}}{1 - 2^{-A}} \approx \frac{|M(S_2, B_2, \hat{q}, \hat{p})|}{4}. \tag{4.35}$$

Proof. We begin by calculating the probability of having an unsold item, which is easy: it is at most as much as the probability of not encountering a buyer within the last $(1 - c) \cdot 2n/2$ agents. Using a similar argument as Lemma 4.8, this probability is at most 2^{-A}.

We now need to calculate the expected number of trades. Let X_i be a random variable indicating that an item was *sold* to the i-th agent. We have:

$$\begin{aligned}\Pr[X_i = 1] &= \Pr[\text{ previous transaction was buying } \wedge X_i \text{ is a buyer }] \\ &= \frac{A}{2A} \cdot \frac{A}{2A + B - 1} = \frac{1}{2} \cdot \frac{A}{2A + B - 1},\end{aligned}$$

since the previous transaction being buying from a seller occurs with probability $\frac{A}{2A}$ as there are A sellers in $M(S_2, B_2, \hat{p}, \hat{q})$ for $2A$ total agents and the sequence is shuffled. The second fraction has $2A + B - 1$ for the denominator, taking into account that one seller has already been used.

[2]The approximate equality is used since in the case where this lemma applies in the analysis of CAUTIOUS(c, ϵ, N), we would have $A \geq 100$ by virtue of the final choice of parameters. We have therefore chosen to sacrifice some precision and ignore the exponential terms.

By linearity of expectation, the total number of trades X is (note that we only consider the first half of σ_2, where we both buy *and* sell):

$$\begin{aligned}\mathbb{E}[X] &\geq \mathbb{E}\left[\sum_{i=2}^{(2A+B)/2} X_i\right] = (A + B/2 - 1)\,\mathbb{E}[X_i] \\ &= (A + B/2 - 1)\frac{1}{2} \cdot \frac{A}{2A + B - 1} \\ &= \frac{A + B/2 - 1}{2A + B - 1} \cdot \frac{A}{2}\end{aligned}$$

We can use this to calculate the expected number of trades in the case where nothing is left unsold:

$$\begin{aligned}\mathbb{E}[X|\text{No items unsold}] &= \frac{\mathbb{E}[X] - \mathbb{E}[X|\text{ Unsold items}]\Pr[\text{Unsold Items}]}{\Pr[\text{No unsold items}]} \\ &\geq \frac{\mathbb{E}[X] - \frac{A}{2^{-A}}}{1 - 2^{-A}}\end{aligned}$$

□

Everything is now in place to provide a lower bound on the gain from trade of the matching calculated by the CAUTIOUS(c, ϵ, N). Assuming $z = |M(S, B)|$, we can compose Lemma 4.7, Lemma 4.8 and Lemma 4.9 to show that with probability at least

$$J(c, \epsilon, z) = f(c, \epsilon, z) \cdot (1 - 2^{-c^2(1-\epsilon)^2 z}) \cdot (1 - 2^{-((1-c)(1-\epsilon) - \frac{1}{2})z}), \tag{4.36}$$

the matching has size at least

$$\frac{((1 - c)(1 - \epsilon) - \frac{1}{2})z}{4}. \tag{4.37}$$

The matching is not a uniformly random subset of $M(S, B)$, but it is skewed to contain higher value trades since $\hat{p} > p$ and $\hat{q} < q$. Taking into account that we run a simple secretary algorithm with probability $1/2$ and assuming we lose the highest valued seller $s^\star$ in our matching when the agents are not well mixed (we can only have one unsold item) the GFT is at least:

$$\frac{1}{2e}b^1 + \frac{J(c,\epsilon,z)}{2} \cdot \frac{((1-c)(1-\epsilon) - \frac{1}{2})m(S,B)}{4} - \frac{1 - J(c,\epsilon,z)}{2}s^\star \tag{4.38}$$

whereas the offline GFT is at most

$$m(S,B) + b^1. \tag{4.39}$$

To upper bound the competitive ratio ρ we analyse three different cases:

1. If $z < N$:

   In this case the algorithm would never detect a sufficiently sized matching and would always run a simple secretary algorithm. Note this is possible, as $c = 0.3 \leq 1/e$ required for the secretary.

   $$\rho \geq \frac{\frac{1}{e}b^1}{m(S,B) + b^1} \geq \frac{1}{e} \cdot \frac{b^1}{(z+1)b^1} = \frac{1}{e(z+1)} \tag{4.40}$$

2. If $N \leq z < N\frac{1}{c(1-\epsilon)}$. In the well-mixed case, the online algorithm will not detect a matching and fall back to secretary. Therefore, the competitive ratio is:

   $$\rho \geq \frac{\frac{1}{2e}b^1 + \frac{1}{2}(f(c,\epsilon,z)\frac{1}{e}b^1 - (1 - f(c,\epsilon,z))s^\star}{m(S,B) + b^1} \geq \frac{1 - e + (1+e)f(c,\epsilon,z)}{2e(z+1)}. \tag{4.41}$$

   given that $c < 1/e$ and the sampling phase for the secretary continues.

3. $z \geq N\frac{1}{c(1-\epsilon)}$. Now in the well mixed case a large enough matching is found. We

have:

$$\frac{\frac{1}{2e}b^1 - \frac{1-J(c,\epsilon,z)}{2}s^\star}{b^1} \geq \frac{1}{2e} - (1 - J(c,\epsilon,z)), \tag{4.42}$$

and

$$\frac{\frac{J(c,\epsilon,z)}{2} \cdot \frac{((1-c)(1-\epsilon)-\frac{1}{2})m(S,B)}{4}}{m(S,B)} \geq \frac{J(c,\epsilon,z) \cdot ((1-c)(1-\epsilon)-\frac{1}{2})m(S,B)}{8}. \tag{4.43}$$

Therefore, the competitive ratio is:

$$\begin{aligned}\rho &\geq \frac{\frac{1}{2e}b^1 + \frac{J(c,\epsilon,z)}{2} \cdot \frac{((1-c)(1-\epsilon)-\frac{1}{2})m(S,B)}{4} - \frac{1-J(c,\epsilon,z)}{2}s^\star}{m(S,B) + b^1} \\ &\geq \min\left\{\frac{1}{2e} - (1 - J(c,\epsilon,z)), \frac{J(c,\epsilon,z) \cdot ((1-c)(1-\epsilon)-\frac{1}{2})m(S,B)}{8}\right\}.\end{aligned}$$

Therefore, c, ϵ and N are selected to maximise the minimum amongst all cases of z, which is picked by the adversary. Computationally, we find that setting $c = 0.3, \epsilon = 0.2758$ and $N = 114$ yields $\rho \geq 1/1434$. □

If we are given that $|M(S,B)|$ will be large, then CAUTIOUS(c,ϵ,N) can be adapted to have greatly improved competitive ratio. In particular, setting $c = \epsilon = 0.01$ achieves $\rho \geq 1/17$ as $|M(S,B)| \to \infty$.

4.4.1 A Note on Revenue

It would be natural to consider if this algorithm is competitive against the revenue objective as well. Unfortunately, this is not the case if truthfulness is required. To see this, consider buying an item for price 5 from a seller valued at 3. Despite the price difference, the full gain from trade is achieved. However, since by truthfulness the agents reveal their valuation only after being offered a posted price, there is no guarantee on revenue, even though the algorithm still trades with the correct subset of agents.

4.5 Conclusion

In this chapter we have studied an online version of the bilateral trade problem, where unit demand buyers and sellers with adversarially selected valuations are randomly permuted and are presented in online fashion to an intermediary. As with the usual double auction with multiple buyers and sellers ([56]) the welfare approximation converges to 1 for large markets, but not the gain from trade. The analysis of this setting has drawn mostly from online algorithms, with very mild influences from mechanism design, just in the way posted prices are set.

The main downside of this chapter is that only ordinal mechanisms were considered. This is not really an issue for the welfare, but the gain-from-trade mechanisms suffers from having to make extremely conservative trades. There should exist a mechanism which takes into account the value of each trade (instead of just the size of the matching) and has an adaptive sampling phase, adjusting prices on the fly to maintain an equilibrium of items.

Part II

Incentives in Blockchain Mining Games

Chapter 5

Blockchain Mining Games with pay forward

5.1 Introduction

Taking a more liberal view to the title "Online Markets", this chapter represents somewhat of a departure from the rest of the thesis, both in the setting studied and the techniques used. Previously, we focused our attention on the crisp, standardised model of Bilateral Trade and took an online spin on it. He we study a literal *online market* which has exploded in popularity and value in recent years, Bitcoin.

Game theoretically modelling the Bitcoin protocol is very challenging. As with most cryptographic settings there are many interconnected layers at play: from latencies and (perhaps purposely) inconsistent information spread across the network, number theoretic weaknesses, human engineering attacks, energy considerations and price fluctuations to competing currencies, just to name a few. To study the incentives of players (more accurately called miners in this setting), we will have to strip the protocol down to it's bare bones to end up with a workable environment with legitimate theoretical merit. As a result, the model will not be as clean and intuitive

as Bilateral Trade, but we will attempt to explain our assumptions and how they still lend validity to the worst case nature of our results.

Interestingly, this setting has no connection to online algorithms, but much more pronounced game theoretic elements than the previous chapters. This chapter is based on the working paper "Blockchain Mining Games with Pay-Forward", which is joint work with Elias Koutsoupias, Foluso Ogunlana and Paolo Serafino, which in turn is an extension of [44] by Kiayias et al.

5.2 Incentives in Blockchain Mining

Bitcoin, currently the most-widely used cryptocurrency, was introduced in a 2009 white paper [60] by the mysterious Satoshi Nakamoto as a form of decentralised, distributed, peer-to-peer digital currency. Although in subsection 5.2.1 we include a high level description of the protocol that is enough to motivate and form a foundation for the theoretical understanding of our results, we kindly suggest the reader to go through [60] for a more accurate (but still accessible, it is a white paper after all) presentation. The backbone of the bitcoin protocol is the *blockchain*, a distributed ledger that (ideally) takes the form of a chain where bitcoin *transactions* are stored into *blocks*. Blocks are created by special nodes of the bitcoin network called miners that: (i) collect and validate the set of transactions to be included in a block and (ii) solve a crypto-puzzle (the so-called *proof of work*) that cryptographically links the newly created (*mined*) block to the tail of the existing blockchain. The main purpose of the blockchain is to solve the problem of *distributed consensus*, where the consensus to be obtained is on the history (and relative order) of the bitcoin transactions.

Since mining new blocks is a computationally intensive (and expensive) task, miners need a reward scheme to keep mining blocks. Bitcoin has two main reward schemes for miners: (i) *transaction fees*, that are left by bitcoin users on a voluntary

basis (and hence vary in frequency and size) and *coinbase* (or block) rewards, which are constituted by a fixed set amount (currently 12.5Ƀ) of newly minted bitcoins. Coinbase rewards are the only way by which bitcoins can ever be created, and the protocol specifies that (to avoid inflation) coinbase rewards be decreasing over time, until the limit of 21 million bitcoins are created: at that point the only incentive scheme left for miners will be transaction fees.

In a distributed setting populated by selfish miners, however, the blockchain will hardly be a chain at all, but rather a *tree*. For instance, network delays may lead miners to add newly mined blocks to different blocks that they believed the tail of the chain. For this reason, newly mined blocks are not part of the consensus until sufficient time has passed since their creation (i.e., d levels of other blocks are added to the blockchain – currently $d = 100$), and block rewards become available to miners only then. Worse still, even though the bitcoin protocol prescribes that miners should mine from the last known block in the chain (the so-called FRONTIER strategy), malicious miners may try to create a *fork* by intentionally adding a sequence of blocks (a *branch*) constituting a parallel history of the transactions, in an attempt to reap more block rewards or to double-spend bitcoins (once per each branch of the fork).

It should be clear from now that the bitcoin protocol is rife with game-theoretic issues, which Nakamoto [60] analysed in a simple model, providing rough estimates showing that if a large majority of miners follow FRONTIER, then their chain will be the longest and contain the agreed upon history.

Since Nakamoto's paper, there has been significant work about the strategies of miners, mainly under the assumption that the reward per block is fixed (see for example [32, 70, 44]). In particular, these *mining games* have been systematically evaluated through the lens of game-theory by Kiayias et al. [44]. They considered mining games in which miners can influence the blockchain in two ways: by strategically choosing to mine at different branches in an attempt to overtake the longest branch and by

withholding their mined blocks and releasing them at the right time, wasting everyone else's computational power. In [70, 44], it was shown by both formal proofs and simulations that the Nakamoto protocol is stable when no miner has computation power more than 0.33. Furthermore, it was also shown that when the miners do not withhold blocks, the protocol is stable if no miner has computation power more than 0.42.

Considerably less effort has been devoted to the case – that will eventually happen since coinbase rewards will drop to zero in the future – of rewards coming solely from transaction fees. Carlsten et al. [20] study mining games in this setting and conclude that the protocol will become unstable and FRONTIER will not be an equilibrium, even for miners with small computational power. This appears to be the single most important vulnerability of the bitcoin at the moment.

In this work, we propose a potential fix to this problem, by making a slight modification to the protocol whereby miners can entice other miners to mine at their block by adding a *pay-forward* amount. This pay-forward amount is collected by the miner of the next block, thus providing incentives to other miners to try to extend the tree from this particular branch. When we extend the available strategies for the miners by adding the ability for pay-forward, we can reclaim the stability of the protocol by simulating fixed block rewards even when all rewards come from transaction fees.

The main technical contribution of this chapter is the study of mining games in which the strategies are extended by pay-forward. We show that with this extension, the stability of the protocol increases significantly: the computational power needed by a dishonest miner to disrupt the blockchain is substantially higher. This shows an interesting trade-off for small miners: on one hand they lose the pay-forward amount but on the other hand, they provide the right incentives to large miners to play FRONTIER and thus secure that their block will end up in the longest chain.

5.2.1 Overview of the Bitcoin Protocol

Based on the white paper by Nakamoto [60], we provide a short description of the Bitcoin protocol. The objective of bitcoin is to solve the problem of double spending in a decentralised currency. Each user (also called *node* or *miner*) maintains a ledger of transactions he has observed which have the form {X pays Y the amount ฿Z}, along with an alleged timestamp. Just like this, it is impossible to guarantee the *order* of transactions and the final state of the market, as the system is decentralised and any miner could claim anything. Bitcoin's ingenuity is to arrange the ledger of every miner into a *blockchain*. To add a block of transactions, the miner would have to solve an extremely hard computational puzzle through a process called mining. Given two blockchains, we consider the longest one to represent the consensus, as it required more computational power to assemble. This only works assuming that a large fraction of the miners are honest.

If the blockchain was just a chain of blocks, the precedence between transaction would be easy to verify. This would only happen if miners always mined at the end of the chain and the network was responsive enough so that immediately after the new block is mined it is propagated to every miner. Due to delays in the network and selfish behaviour of the miners, the blockchain ends up becoming a rooted tree, where every root to leaf path contains some subset of transactions in sequence. To minimise this, the protocol itself suggests miners work at the end of the *longest* branch, which is the one containing the most computational effort so far. We call this strategy FRONTIER and the miners who follow it *honest*.

To make mining a block computationally expensive, a technique known as *proof of work*[30, 42, 65, 8] is used. Whenever a user adds a block to this local blockchain, he need to decide which transaction to include (amongst the globally broadcast transactions that are not part of his blockchain yet) and the predecessor of his block. He then needs to compute a *nonce* value, added to the end of his block, which also contains

transaction data and the hash of the predecessor. The nonce must be such so that the entire block's hash (SHA-256) is less than a specified value T. Verifying that a block is valid is easy, but it is assumed that there is no faster way to compute the nonce other than trying values at random. Therefore, the probability that a certain miner computes the block is proportional to his computational power amongst all miners. The difficulty is adjusted by increasing T, so that the average time it takes for the *whole* network to mine a block is 10 minutes. These 10 minutes also serve as timestamps: whenever a miner announces his transaction, he also appends some time data close to the latest mined block in his blockchain. Therefore, if most miners are honest the longest chain will represent the true order of transactions. The last step is to provide incentives for the miners to spend the computational power to mine, by giving rewards to miners who *successfully* add blocks to the blockchain.

5.2.2 Our Results

We consider two types of *stochastic games*, whose states are rooted trees. The game is played in discrete time-steps. At the beginning of each time-step, every miner chooses a block and tries to mine from it. Each player i has probability p_i to mine a new block, proportional to his computational power. In the end, the new block may be added to the blockchain or kept hidden and released later. After many rounds of playing, the utility of each player is the fraction of bitcoins he owns (from block and pay-forward rewards) over the total value that has been mined in the longest chain.

Let p be the computational power of the strongest player, called Miner 1. If p is large enough, his best response, given that everyone else is playing FRONTIER, may be a different strategy. We find thresholds on p and w, the pay forward amount of every other player, to guarantee that mining at the end of the longest chain is a best response for Miner 1. As in [44], we consider two variants. In the immediate release case, any mined block has to be added to the blockchain immediately for other players to use.

In this case, without pay forward rewards, it was proven in [44] that FRONTIER is a best response for Miner 1 for $p < h$, where $0.361 \leq h \leq 0.455$. Experimentally, they also showed that there is always a deviating strategy for $h \approx 0.42$. In the strategic release case, the newly mined blocks are public knowledge, but can only be used by other players when their creators decide to *release them*. As before, it was shown in [44] that there is a threshold $\hat{h} \geq 0.308$ (experimentally shown to be approximately 0.33 [44, 70]) such that $p < \hat{h}$ leads to FRONTIER as a best response for Miner 1.

We improve these thresholds to $h = 0.5$ and $\hat{h} \geq 0.344$ by showing that there exists some w (as a function of p) that guarantees Miner 1's best response is to play FRONTIER when all remaining miners play FRONTIER and pay forward w. For the strategic release case, we find experimentally that $\hat{h} \approx 0.38$. We also devise a linear program to calculate the minimum value of w for any given p.

For the case where the payment comes mainly from transaction fees, we propose an amendment to the FRONTIER strategy so that the total payment for each block is a constant $f^\star$. Let's define as "legal" any block with total miner reward at most $f^\star$.

- Given the pay-forward from previous blocks, the coinbase, and the transaction reward of the block, always pay forward enough so that the block becomes "legal".
- Mine only from the deepest node of chains containing "legal" blocks.

We also propose a change in the bitcoin protocol, to facilitate the amended FRONTIER strategy. Given a large enough initial 'buffer' that gets passed around from honest miners, each of them receives exactly the same amount they would if block rewards were still present. This buffer does not cause inflation, as new bitcoins are never added. Periodically $f^\star$ is recalculated given the previous volume of trades to keep the buffer stable. In this case, the bitcoin protocol can remain as stable as it currently

is, with fixed block rewards.

5.2.3 Related Work

Bitcoin, originally introduced by Nakamoto in [60] and followed by several cryptocurrencies, such as Litecoin, Ethereum and Ripple, initiated the research on blockchains. Nakamoto's original paper analyses double spending attacks while Rosenfeld provides a more detailed analysis [67]. Bonneau et al. [17] and Tschorsch and Scheuermann [74] provided extensive surveys of the research and challenges in cryptocurrencies.

Kroll et al. [54] was one the first papers to consider the economics of Bitcoin mining, assuming that participants behave according to their incentives. Eyal and Sirer [34] showed that the security of bitcoin is not guaranteed by a majority of honest miners as was previously assumed. They gave a specific mining strategy and argued that a pool of miners, with at least a 1/3 of the total processing power, can get extra profit regardless of the block propagation characteristics of the network. With sufficiently favourable block propagation, this threshold of processing power falls to 0. Extending this, Sapirshtein et al. [70] provided systematic analysis of the space of selfish mining strategies based on computational results.

A similar approach was taken by Kiayias et al. [44] who provided a framework for studying the strategic considerations made by Bitcoin miners. They formulated two abstract stochastic games and proved rigorous bounds on the threshold of computational power below which the honest strategy is a Nash Equilibrium. Carlsten et al. [20] extended the previous approaches by investigating mining games when the reward for miners varies and comes mainly from transaction fees. They observed that the random block arrival times lead to high variance in rewards and they considered strategies that lead to instability. They showed via simulation that an equilibrium exists, but it has the counter intuitive and undesirable effect of a growing backlog of unprocessed transactions.

Several other studies examine possible attacks on the protocol and suggest adaptations to ensure its security. Rosenfeld [66] and Courtois and Bahack [26] discuss pool mining attacks. Eyal [32] introduces an attack in which mining pools infiltrate one another resulting in a pool game. Lewenberg et al. [55] also provide a game theoretic analysis of pool mining. Babaioff et al. [5] consider Sybil attacks on the network and propose a reward scheme to prevent miners from hiding transactions in competition with other miners.

There is also considerable work on the performance and scalability of blockchain inspired networks. Sompolinsky and Zohar's [72] Greedy-Heaviest-Observed-Sub-Tree rule is an alternative consensus mechanism. Eyal [33] proposes Bitcoin-NG which increases the throughput of Bitcoin. Poon and Dryja's Lightning Network [64] scales via off-chain transactions and hashed commitments. Sompolinsky and Zohar's PHANTOM [73] achieves near unlimited transaction throughput. SPECTRE [71], by Sompolinsky et al, fully orders the transactions in blocks using recursive elections and can used in combination with PHANTOM. Kiayias et al. developed Ouroboros[45], a proof of stake network with provable security, utilising a secure multi-party coin flipping protocol.

5.3 Model and Notation

The model contains subtleties in the abstractions that at first glance might be overlooked by readers who are too familiar with the bitcoin protocol. We do our best to point out any of those instances, but be especially vigilant when reading about the way payoffs are calculated.

The bitcoin mining game with pay forward is an abstraction of the actual protocol, simplified in a way that can only accentuate its game theoretic issues (i.e. negative results hold a fortiori for the actual protocol) while being open to rigorous analysis.

The parameters of the game are:

1. the number n of miners (or players).

2. the probabilities $p_1, \ldots, p_n$, representing the *hash power* of each miner, which is the probability that they solve the crypto-puzzle. They are proportional to the computational power of each miner and such that: $p_i \geq 0$ for each i and $\sum_{i=1}^{n} p_i = 1$.

3. the depth d after which mined blocks are 'paid', i.e. their coinbase becomes usable by the respective miner. To enhance the stability of the bitcoin protocol, a block is considered *safe* if there exists a tree of depth at least d rooted at that block. In that case, the block is generally accepted as being a permanent part of the longest path and its miner can finally spend his reward for mining it. The attacker will not try to undermine this block, as any transaction involving this coinbase reward will be invalidated, leading to a decrease in trust in the bitcoin protocol. Without loss of generality we mostly consider $d = \infty$ which gives the attacker extra power, as he can essentially mine at any block he wants.

We mostly assume that each block has a *fixed reward*, which by appropriate scaling is equal to 1. However, the main motivation of this work is the work of Carlsten et al. [20] which shows that when the reward per block is not fixed but comes from transaction fees, the Nakamoto [60] strategy is unstable, in the sense that it is not an equilibrium, even for players of small computational power. We address this issue in Section 5.6. Our suggestion is that by extending the strategies of the miners with pay forward, the protocol can regain its stability.

During the execution of the protocol, miners add their blocks to the blockchain one at a time[1]. Their goal is to maximise the fraction of their blocks on the longest

[1]Due to the decentralised nature of bitcoin, it is possible for more than one miners to mine a block simultaneously. This event does occur by chance, but it is very rare and it is not something a miner can plan for. For this reason, we assume that exactly one block is mined at every step,

branch of the blockchain, as the longest branch represents the consensus blockchain and all other branches are pruned and the computational for their creation is wasted. This does not mean that all the miners try to extend the longest branch at every time step, because some times they might benefit by either adding a block to a shorter branch or withholding it for later.

Definition 5.1. *A public state is a rooted tree. Every node is labelled by one of the players and the amount of money he pays-forward. The nodes represent mined blocks and the label indicates the player who mined the block. The pay-forward amount is collected by the next miner in the longest branch. Every level of the tree has at most one node labelled i because there is no reason for a player to mine twice the same level.*

A private state of player i is similar to the public state except it may contain more nodes called private nodes and labelled by i. The public tree is a subtree of the private tree and has the same root.

The incomplete information case (where the p_i's or private states are unknown) is significantly more complicated. In this work we only consider the full information case where even the private states are common knowledge (but only become part of the public state when their respective miners add them). We consider two variants:

Immediate-release model Every mined block is added to the public tree immediately.

Strategic-release model Mined blocks can be withheld: every miner is aware of their existence, but they can only mine from them once they are added to the public tree.

The second model has no counterpart in practice, but it serves as an intermediate step between the full and incomplete information models and allows us to study issues

but most of our analysis could be easily extended to the case in which each miner succeeds with probability p_i independently.

around strategic release of blocks. If a strategy is not dominant in this model it cannot be dominant in the incomplete information setting.

Definition 5.2 (Strategy)**.** *The strategy of miner i can be fully characterised by three functions $\mu_i, r_i, \mathrm{PF}_i$:*

- *the mining function μ_i selects a block of the public state to mine form*
- *the release function r_i which is the rooted tree he releases to the public. It is a subtree of his private state that contains the public state.*
- *the pay-forward function PF_i which is the amount of money to be left for the next miner.*

Both functions depend on the public and private states of every player.

The original suggested strategy in [60] is Frontier.

Definition 5.3 (Frontier)**.** *A miner follows strategy* Frontier *if he releases mined blocks immediately, always mines at the deepest node in the blockchain and pays forward 0 Ƀ.*

In the following, we refer to Frontier(w) as the strategy that mines and releases like Frontier but always pays forward w.

The game is played in *phases*. At each phase exactly one miner will mine a block. Then he will choose if he wants to release it, which may trigger a cascade of releases from other players. Eventually, once no one else wants to release anything the public knowledge is updated and the next phase begins. Even if no one releases a block, miners know when the phase ends since everything is public knowledge.

To incentivise miners to mine new blocks, payments are necessary. When a block is added, it is unclear if it is going to remain on the longest branch permanently. To remedy this, blocks are paid for only after their branch is increased by d blocks,

after which time it is safe to assume it will stay in the longest branch. Blocks are paid through coinbase, transaction fees and pay-forward. Currently, coinbase rewards (which can be claimed at $d = 100$) far outweigh transaction fees. However, by design Bitcoin will eventually do away with coinbase rewards to limit inflation, at which point only transaction fees will remain. Pay-forward rewards are always considered, while in Sections 5.4 and 5.5 we also have coinbase rewards and in Section 5.6 transaction fees. For game theoretic analysis we rigorously define payments:

Definition 5.4 (Payments)**.** *For some nodes of the tree, the miners who discovered them will get a payment (coinbase rewards are normalised to 1). The payments comply with the following rules:*

- *the blocks that receive a payment must form a path from the root. This immediately adds the restriction that at every level of the tree exactly one node receives payment.*
- *among the blocks of a single level that satisfy the above condition, the first one which succeeds in having a descendant d generations later receives payment.*

Since only one block per level is paid for and they form the longest branch, the utility of a miner in the long run is defined as the fraction of pay-forward and coinbase rewards he obtained minus the pay-forward he paid in that branch, over it's length.

Remark 4. *Even though payments are described for a finite time horizon, we are primarily interested in the expected payment per level in the steady state of the system, for an infinite time horizon.*

When a block is paid for, strategic miners will ignore any branch that starts at an earlier node. This limits the shape of the public state to a long path (the *trunk*) with ignored *stale* branches dangling from it. Only at the end of the trunk we may find multiple active branches, competing to reach length d and render the others

stale. We will also consider the case where $d = \infty$. Here there is the possibility that two competing branches will go on forever, but since one of them will have less computational power, a case of gamblers ruin makes this impossible.

5.4 Immediate Release

In this section we show that if all miners but one follow FRONTIER(w) (for some value of w) the remaining miner's best response is FRONTIER, provided his hash power is $p < 0.5$. Since only one miner might deviate, we can view this situation as Miner 1 being the potentially deviant large miner with hash power $p < 0.5$ and Miner 2 all small miners combined into one, which we call the honest miners, since they follow the same strategy.

In essence, every honest player mines at the deepest node and always pays forward w. This amount propagates across the small miners forming Miner 2 but Miner 1 can claim it for himself, leading to an interesting stochastic game where Miner 1 is incentivise to mine the longest chain, since the rewards don't accumulate.

The blockchain itself is a rooted tree, but after pruning abandoned branches we are left with one long, undisputed, main path followed by two branches of length a and b, mined by Miner 1 and Miner 2 respectively. Therefore, the possible states of the game have the form (a, b, c), where $a \leq b + 1$ since Miner 2 always mines the longest branch (except briefly, right when Miner 1's branch overtakes the longest) and c is either 0 or 1, to indicate whether the *last block before the fork* contained a pay forward reward.

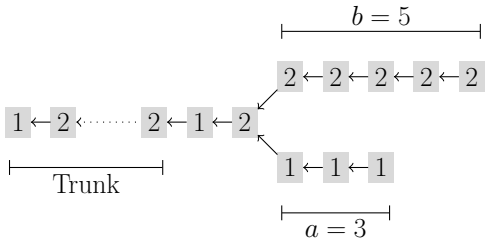

Figure 5.1: A typical state tree. The trunk represents the blocks whose rewards have already been collected. The current state $(3, 5, 1)$ of the game is represented by the blocks mined by Miner 1 and Miner 2 and $c = 1$ because Miner 2 controls the block before the fork.

The set of states (a, b, c) can be partitioned as follows:

1. **Winning states**: the set W of states where Miner 2 capitulates and starts mining from the tip of Miner 1's branch. This happens exactly when $a = b + 1$, therefore $W = \{(b + 1, b, c) \mid b \geq 0 \text{ and } c \in \{0, 1\}\}$. After Miner 1 overtakes, the new state of the game is $(0, 0, 0)$ since they both mine at the same point.

2. **Capitulation states**: the set C of states where Miner 1 capitulates, abandons his branch and mines from a block of Miner 2's branch, thus moving the game to state $(0, s, 1)$ for some s s.t. $0 \leq s < b$. We say that Miner 1 capitulates *at* (a, b, c) and *to* $(0, s, 1)$. Clearly, after capitulating there would be a pay forward reward available.

3. **Mining states**: the set M of states where Miner 1 and Miner 2 mine their respective branches.

Miner 1 can capitulate to any state $(0, s, 1)$ and will always choose the one that maximises his payoff. Since he is rational, when capitulating from state (a, b, c) he would only go to states $(0, s, 1)$ with $s < b$, otherwise he would be undercutting his own tentative branch. The strategy Frontier has $M = \{(0, 0, 0), (0, 0, 1)\}$, $C = \{(0, 1, 0), (0, 1, 1)\}$ and always capitulates to $(0, 0, 1)$.

Let $g_k(a,b,c)$ denote the optimal expected gain of Miner 1 starting from state (a,b,c) when the longest chain is extended by k levels. Knowing Miner 1 will never pay-forward, we can recursively define:

$$g_k(a,b,c) = \begin{cases} 0 & \text{if } k = 0 \\ g_{k-1}(0,0,0) + a + w \cdot c & \text{if } a = b+1 \\ \max \begin{cases} \max_{s=0,\ldots,b-1} g_k(0,s,1) \\ p g_k(a+1,b,c) + (1-p) g_{k-1}(a,b+1,c) \end{cases} & \text{otherwise} \end{cases}. \tag{5.1}$$

Clearly, Miner 1 will eventually add some block to the chain and the game will restart at state $(0,0,0)$. Therefore, we expect that the initial state (a,b,c) has no effect asymptotically, as the total gain over the infinite time horizon will consist of some transient gains from the transitions $(a,b,c) \to \ldots \to (0,0,0)$ and then the gains from the cycle $(0,0,0) \to \ldots \to (0,0,0)$ over and over, therefore this will asymptotically dominate the gain per level. We define the expected gain per level $g^\star$ as:

$$g^\star = \lim_{k \to \infty} \frac{g_k(a,b,c)}{k}. \tag{5.2}$$

We can decompose $g^\star$ into two separate quantities:

$$g^\star = q_M + q_{\text{PF}} \cdot w, \tag{5.3}$$

where q_M is the expected fraction of blocks he mined and q_{PF} the expected fraction of blocks mined and containing pay forward rewards.

Lemma 5.1. *If Miner 1 follows* Frontier*, we have $q_M = p$ and $q_{\text{PF}} = p(1-p)$.*

Proof. If Miner 1 is honest then the probability that he mined any arbitrary block is p. That block also contains pay forward rewards if the one that immediately preceded

it was mined by Miner 2, which occurs with probability $p(1-p)$. □

We are ready to state the main theorem of this section.

Theorem 5.1. *For every $p < 0.5$, there exists $w \geq 0$ large enough so that if every miner but one follows* Frontier(w)*, the best response of the remaining miner with hash power p is* Frontier*.*

Proof. We will show that q_{PF} attained by Frontier is at least as large as that of any optimal strategy, and in particular it is larger for $0 < p < 0.5$.

Lemma 5.2. *If* Frontier *is not the optimal strategy and $0 < p < 0.5$, we have that $p(1-p) > q_{\text{PF}}$. For $p = 0$ or $p = 0.5$ we have equality.*

Proof. Assuming Frontier is not an optimal strategy, we consider different cases, depending on the actions of the optimal strategy starting from state $(0, 1, c)$.

Clearly, we have that $g_k(0,1,1) \geq g_k(0,1,0)$ for all k. To see this, consider what happens if we apply exactly the same sequence of actions starting from $(0,1,0)$ and from $(0,1,1)$. As the only difference between the two states is the initial pay forward amount, if we start from $(0,1,1)$ the reward will be the higher (in expectation) by some fraction of w. Therefore, if the optimal strategy capitulates at $(0,1,1)$ it must also capitulate at $(0,1,0)$. This is exemplified in Figure 5.2, where blocks with a red outline represent those mined by the optimal strategy, and red arrows show how the action of the optimal strategy in one case implies the same action in the other.

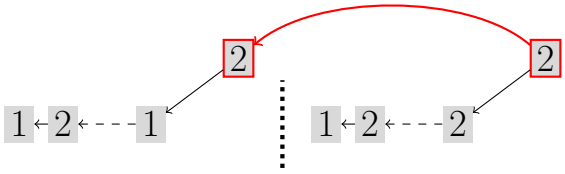

Figure 5.2: Mining like Frontier

But in this case, the optimal strategy capitulates at exactly the same states as Frontier, as shown in Figure 5.2. Since the blockchain starts empty, at $(0,0,0)$ this

strategy would behave exactly as FRONTIER and would therefore not be optimal.

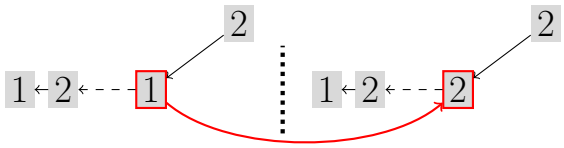

Figure 5.3: Case 1

In Case 1 (shown in Figure 5.3), we assume that the optimal strategy mines at $(0, 1, 0)$. As before, since $(0, 1, 1)$ is a more beneficial state it would mine from there as well.We can therefore assume that the optimal strategy always capitulates to states $(0, s, 1)$ with $s \geq 1$. Blocks are permanently added to the blockchain only after a capitulation. Given that, in order to compute q_{PF} for Miner 1, we construct the Markov chain shown in Figure 5.4, with states $(0, 0, 0)$ and $(0, s, 1)$ indicating that Miner 2 or Miner 1 capitulated respectively. Each transition is labelled with a pair where the first element is the transition probability, whereas the second element is the minimum number of blocks added to the chain when the transition occurs.

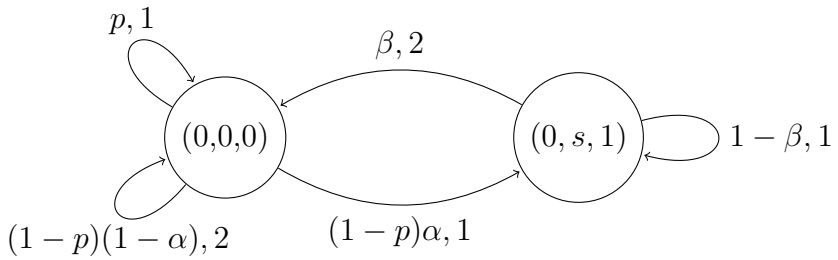

Figure 5.4: Case 1 Markov chain

Starting from $(0, 0, 0)$, with probability p Miner 1 adds a block and the game restarts. With probability $1 - p$ we move to state $(0, 1, 0)$ and the optimal strategy has probability α of losing the race and capitulating to some state $(0, s, 1)$. From there, the optimal strategy has probability β of winning and returning to $(0, 0, 0)$.

As q_{PF} is an asymptotic quantity, we are interested in the fraction of pay forward rewards per block mined at the stationary distribution. Let π be the stationary

probability of state $(0, s, 1)$. Then:

$$\pi = \frac{\alpha - \alpha \cdot p}{\alpha - \alpha \cdot p + \beta} \tag{5.4}$$

In the actual game, multiple blocks are potentially mined in each edge transition, and only the first block might claim a pay forward reward. On every transition at least one block is permanently added, *except from* $(0, s, 1)$ *to* $(0, 0, 0)$ *and* $(0, 0, 0)$ *to itself through transition* $(1 - p)(1 - \alpha)$. In these cases at least two blocks are added, since Miner 1 must mine at least two blocks to be ahead of Miner 2. From $(0, s, 1)$ to $(0, 0, 0)$ is the only transition where Miner 1 wins a block containing pay forward. Dividing the blocks with pay-forward over the total amount mined at the stationary distribution:

$$q_{\text{PF}} \leq \frac{\pi\beta}{(1-\pi)(2(1-\alpha)(1-p) + \alpha(1-p) + p) + \pi(2\beta + 1 - \beta)} \tag{5.5}$$

We now need to set the parameters α and β optimally to maximise the upper bound on q_{PF}. We take the derivative of the upper bound of q_{PF} with respect to α and β to get:

$$\frac{dq_{\text{PF}}}{d\alpha} = \frac{\beta^2(2-p)(1-p)}{(\alpha(p-1) + \beta(p-2))^2} \geq 0 \tag{5.6}$$

and

$$\frac{dq_{\text{PF}}}{d\beta} = \frac{\alpha^2(1-p)^2}{(\alpha(p-1) + \beta(p-2))^2} \geq 0. \tag{5.7}$$

From [44, Lemma 1], we know that the probability that Miner 1 reaches a winning state starting from (a, b) is at most $(p/(1-p))^{(b-a+1)}$. Therefore, the optimal strategy must have

$$\alpha \leq 1 \text{ and } \beta \leq \left(\frac{p}{1-p}\right)^{s+1} \leq \left(\frac{p}{1-p}\right)^{2} \tag{5.8}$$

Since q_{PF} is an increasing function of α, β, the upper bound is obtained by setting

them to their highest possible value, leading to:

$$q_{\text{PF}} \leq \frac{(1-p)p^2}{1-(1-p)p(3-2p)}. \tag{5.9}$$

Compared to the honest outcome $p(1-p)$, we have:

$$p(1-p) - \frac{(1-p)p^2}{1-(1-p)p(3-2p)} = \frac{(1-p)p(1-2p)}{1-(1-p)p(3-2p)} \geq 0, \tag{5.10}$$

with equality obtained only for $p = 0$ and $p = 0.5$. We also plot this result at the end of the proof.

In the last case, the optimal strategy capitulates at $(0, 1, 0)$ but mines at $(0, 1, 1)$, as seen in Figure 5.5.

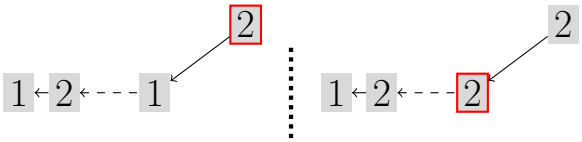

Figure 5.5: Case 2

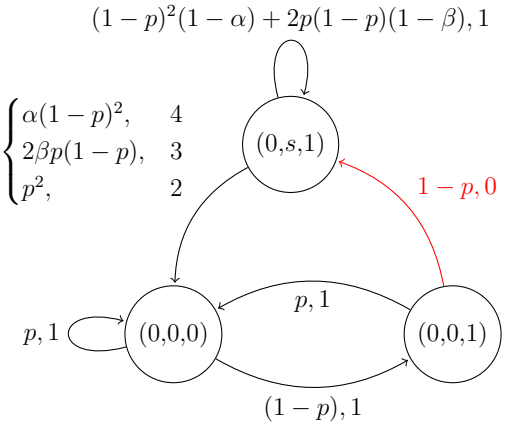

Figure 5.6: Case 2 Markov chain

The markov chain in this case (Figure 5.6) is trickier. As before each state represents a capitulation: $(0, 0, 0)$ for Miner 2 and $(0, 0, 1)$, $(0, s, 1)$ with $s \geq 1$ for Miner 1. However, the transition indicated with the red arrow is *not* a capitulation: it is merely a move to a state that Miner 1 could also capitulate *to*. To more accurately model the transitions of state $(0, s, 1)$, we consider multiple moves ahead. Specifically, three outcomes could happen before the optimal strategy considers capitulating. With probability p^2, Miner 1 could add two blocks and move to $(0, 0, 0)$. Miner 2 could

also add two blocks and move to $(0, s+2, 1)$, with probability $(1-p)^2$. Then, Miner 1 wins with probability α. Finally, Miner 1 and Miner 2 could both add one block leading to $(1, s+1, 1)$ with probability $2p(1-p)$. From there, Miner 1 wins with probability β.

Carefully adjusting for the more complicated transitions, the same analysis as before holds. For both cases, we plot the difference $p(1-p) - q_{\text{PF}}$ in Figure 5.7.

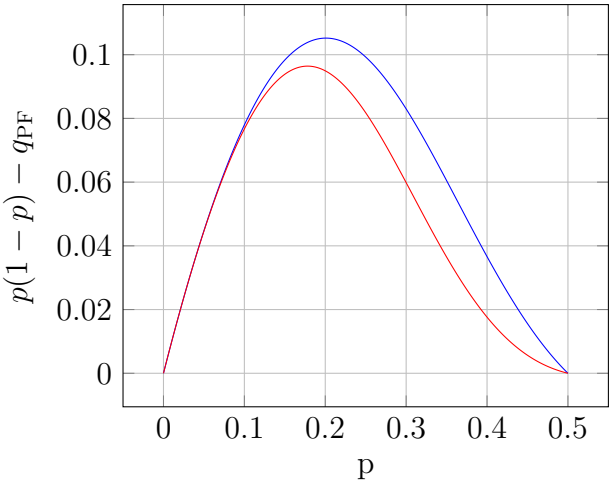

Figure 5.7: Advantage of FRONTIER in collecting q_{PF} for Case 1 (blue) and Case 2 (red)

□

Therefore, by (5.3) and Lemma 5.1 we have that

$$g^\star = q_M + q_{\text{PF}} \cdot w \leq p + (1-p) \cdot w \Rightarrow w \geq \frac{q_M - p}{(1-p) - q_{PF}}$$

and since $(1-p) > q_{PF}$ for $0 < p < 0.5$ there must be some value of w large enough to make FRONTIER the best response. □

Unfortunately, this result is not enough to calculate w. Giving crude upper bounds on q_M or q_{PF} independently is not too hard, we can use [44, Lemma 1] from example. However, because the trade-off between the two is hard to establish and the actual

blockchain does not use $d = \infty$, it is more useful to develop an algorithm that finds the minimum w required for specific a d, to any degree of accuracy.

5.4.1 Calculating the optimal w for finite d

In reality, the values of w needed are not too large. We could attempt to find the minimum w necessary by computing $g_k(0,0,0)/k$ for large values of k and comparing with $p + p(1-p)w$, but a naive implementation would take $O(k^3)$ time. To simplify this search we can define a potential as in [44]

$$\phi(a,b,c) = \lim_{k\to\infty} g_k(a,b,c) - k \cdot g^\star, \tag{5.11}$$

that captures the advantage of Miner 1 at different states. Following recurrence 5.1, we have:

$$\phi(a,b,c) = \begin{cases} \phi(0,0,0) + a + c \cdot w - g^\star & \text{if } a = b+1 \\ \max \begin{cases} \max_{s=0,\dots,b-1} \phi(0,s,1) \\ p \cdot \phi(a+1,b,c) + (1-p) \cdot (\phi(a,b+1,c) - g^\star) \end{cases} & \text{otherwise} \end{cases}, \tag{5.12}$$

where we set $\phi(0,0,0) = 0$. Finding a ϕ that satisfies these constraints is even harder. However, if we truncate the game at d, it becomes more feasible through linear programming. Notice that (5.12) (which holds for $d = \infty$) does not explicitly define the value of $\phi(a,b,c)$: the recursion is unbounded.

By truncating the game at d we limit the available states since $a, b \leq d$. Specifically, Miner 1 *has* to capitulate when $b = d$. Relaxing the two maxes by inequalities,

we get the following LP:

$$\begin{aligned}
&\text{minimise} \quad g + \tfrac{1}{D}\sum_{a=0}^{d}\sum_{b=0}^{d}\sum_{c=0}^{1}\phi(a,b,c) \\
&\text{subject to} \\
&\qquad \phi(b+1,b,c) \geq \phi(0,0,0) + b + 1 + c\cdot w - g \qquad b < d, c \leq 1 \\
&\qquad \phi(a,b,c) \geq \phi(0,s,1) \qquad a \leq b < d, s < b, c \leq 1 \\
&\qquad \phi(a,b,c) \geq p\cdot\phi(a+1,b,c) + (1-p)(\phi(a,b+1,c) - g) \qquad a \leq b < d, c \leq 1
\end{aligned} \tag{5.13}$$

where $\phi(a,b,c) \geq 0$ and $D \gg d^2$ is a normalising factor to keep the sum of states insignificant compared to g. The constraints are straightforward enough, but the objective is a minimisation, which might appear odd at first. The second term of the sum ensures that feasible solutions only contain potentials that tightly satisfy (5.12). We minimise g to use the following lemma:

Lemma 5.3. *For any g, ϕ pair satisfying (5.12) we have: $g_k(a,b,c) \leq \phi(a,b,c) + k\cdot g$.*

Proof. We use induction on k. Clearly, for $k = 0$ we have $g_0(a,b,c) = 0 \leq \phi(a,b,c)$. For fixed k we do strong induction on b and backwards induction on a. Starting with $b = 0$, we begin the induction on $a = b + 1$:

$$\begin{aligned}
g_k(1,0,c) &= g_{k-1}(0,0,0) + 1 + c\cdot w \\
&\leq \phi(0,0,0) + (k-1)g + 1 + c\cdot w \\
&= \phi(0,0,0) + 1 + c\cdot w - g + k\cdot g \\
&= \phi(1,0,c) + k\cdot g
\end{aligned}$$

and

$$\begin{aligned} g_k(0,0,c) &= p \cdot g_k(1,0,c) + (1-p) g_{k-1}(0,1,c) \\ &\leq p(\phi(1,0,c) + k \cdot g) + (1-p)(\phi(0,1,c) + (k-1)g) \\ &= p \cdot \phi(1,0,c) + (1-p)(\phi(0,1,c) - g) + k \cdot g \\ &= \phi(0,0,c) + k \cdot g. \end{aligned}$$

For $b > 0$ the proof works the same, starting from $g_k(b+1, b, c)$ as the base case for a. For $a < b + 1$:

$$\begin{aligned} g_k(a,b,c) &= \max \begin{cases} \max_{s=0,\ldots,b-1} g_k(0,s,1) \\ p g_k(a+1,b,c) + (1-p) g_{k-1}(a,b+1,c) \end{cases} \\ &\leq \begin{cases} \max_{s=0,\ldots,b-1} \phi(0,s,1) + k \cdot g \\ p \cdot (\phi(a+1,b,c) + k \cdot g) + (1-p)(\phi(a,b+1,c) + (k-1) \cdot g) \end{cases} \\ &= k \cdot g \begin{cases} \max_{s=0,\ldots,b-1} \phi(0,s,1) \\ p \cdot \phi(a+1,b,c) + (1-p)(\phi(a,b+1,c) - g) \end{cases} \\ &= \phi(a,b,c) + k \cdot g, \end{aligned}$$

where strong induction was used for the capitulation case. □

Using this lemma, we can find the minimum value of w that leads to Frontier being a best response for different values of p by using binary search on w, checking that the ϕ produced from the LP satisfies (5.12) and that $g = p + p(1-p)w$. By the definition of $g^\star$:

$$g^\star = \lim_{k \to \infty} \frac{g_k(a,b,c)}{k} \leq \lim_{k \to \infty} \frac{\phi(a,b,c) + k \cdot g}{k} = p + p(1-p)w,$$

For $d = 8$, we obtain the following graph:

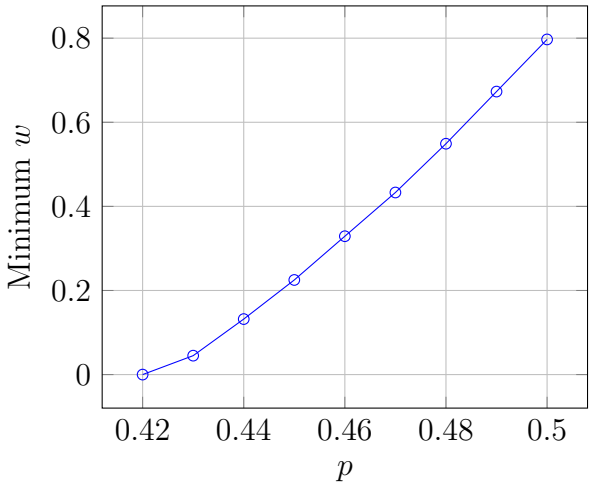

Figure 5.8: Minimum values of w for the immediate release case.

It is not a mistake that $w < 1$ for $p = 0.5$. The reason is that d is finite. For $d = \infty$, the strategic miner with $p \geq 0.5$ never has a reason to capitulate, no matter w: the probability that his branch eventually overtakes the honest one is always quite high. However, the probability that he will win the race *within d steps* is much smaller, leading to this result.

Remark 5. *It is difficult to establish how useful these values of w are. In essence, there exists a trade-off between protocol compliance and fairness. As p grows, if the small miners don't pay forward then Miner 1 will get disproportionately higher payments, as well as destabilise the protocol. By paying forward, the society of small miners can make the payments even more unfair, as they will collectively pay Miner 1 more to keep him on the honest branch. However, this has the effect of limiting variance in payments: if a small miner manages to add a block, he is guaranteed to keep his coinbase reward. An experimental analysis shows that for* $p \geq 0.44$, *w is small enough to be a best response from the small miners for this exact reason, even though the society overall loses more money. For* $p > 0.44$ *paying forward the appropriate w might not be a Nash equilibrium under the current incentives studied*

in this work. A more detailed study would require proper quantification of the small miners' valuation of the stability of bitcoin.

5.5 Strategic Release

Similarly to the immediate release case, we identify the maximum hash power p such that if all miners but one follow FRONTIER(w), the remaining miner's best response is FRONTIER if his hash power is less than p. As before, it is enough to consider a two player game where Miner 1 is the deviant miner with hash power $p < 0.5$ and Miner 2 represents all small miners, as they follow the same strategy.

The blockchain retains its structure: it still contains a long trunk of blocks followed by two branches, corresponding to the released blocks of Miner 1 and Miner 2. Since Miner 1 could also have more unreleased blocks in his branch the state is now a quadruple (a_r, a, b, c) where a is the number of blocks in Miner 1's branch, of which a_r have been released, and b is the number of blocks in Miner 2's branch. Also, contrary to the immediate release case we can have $a > b + 1$.

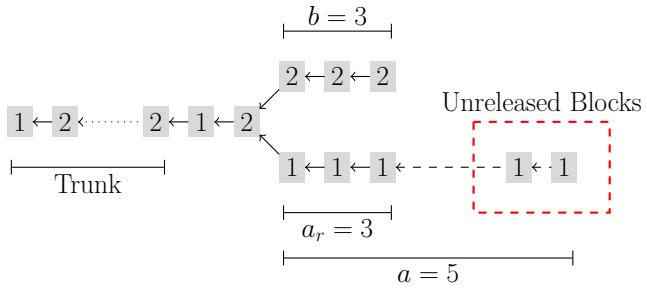

Figure 5.9: This tree represents state $(3, 5, 3, 1)$. Miner 1 has already mined 5 blocks ahead of the fork, but has only released 3 of them. Miner 2 knows this because the game is complete information. However, he cannot mine there until Miner 1 releases his blocks. Miner 1 can strategically release blocks right when Miner 2's branch is about to become the main one, thus wasting Miner 1's computational power.

Since Miner 2 follows the FRONTIER(w) strategy, he will capitulate if $a_r \geq b + 1$ and continue mining his branch for $a_r < b + 1$. Therefore, without loss of generality

we can assume that at state (a_r, a, b, c) if $a < b + 1$ then $a_r = a$ and if $a \geq b + 1$ then $a_r = b$, otherwise Miner 2 would immediately capitulate and the game would continue at state $(0, a - a_r, 0, 0)$. Therefore, we can encode the states of the mining game with strategic release by the triplet (a, b, c), where $a_r = \min(a, b)$.

As before, we need to recursively define the gain of Miner 1 after the longest branch is extended by k levels. When $a < b + 1$, the actions of Miner 1 are exactly the same as in the immediate release case: he can either capitulate to $(0, s, 1)$ or mine. However, for $a \geq b + 1$ he can release one more block to force Miner 2 to capitulate and continue at state $(a - b - 1, 0, 0)$.

$$\hat{g}_k(a,b,c) = \max \begin{cases} k + b + c \cdot w & \text{if } a \geq k + b \\ \max \begin{cases} \max_{s=0,\dots,b-1} \hat{g}_k(0, s, 1) \\ p\hat{g}_k(a+1, b, c) + (1-p)\hat{g}_{k-1}(a, b+1, c) \\ \hat{g}_{k-1}(a - b - 1, 0, 0) + b + 1 + c \cdot w \end{cases} & \text{otherwise} \end{cases}, \tag{5.14}$$

where the last term applies for $a \geq b+1$. Equivalently, we can define $g_k(a, 0, c) = -\infty$ for all k and $a < 0$. The first term is necessary, as without it the recurrence is ill-defined and Miner 1 can keep mining forever. Once he reached the horizon k he can safely release all his blocks. We define the expected gain per level $\hat{g}^\star$ and potential $\hat{\phi}$ as we did in (5.2) for the immediate release. Note that the case where $a = k + b$ does not appear, since ϕ is the asymptotic advantage as $k \to \infty$ and a, b are bounded.

$$\hat{\phi}(a,b,c) = \max \begin{cases} \max_{s=0,\dots,b-1} \hat{\phi}(0, s, 1) \\ p\hat{\phi}(a+1, b, c) + (1-p)(\hat{\phi}(a, b+1, c) - \hat{g}^\star) \cdot \\ \hat{\phi}(a - b - 1, 0, 0) + b + 1 + c \cdot w - \hat{g}^\star \end{cases} \tag{5.15}$$

Releasing blocks only causes Miner 2 to capitulate if $a \geq b + 1$, so we set $\phi(a, 0, c) =$

$-\infty$ for $a < 0$. Before moving on, we need a useful inequality of the *immediate release* potential ϕ, to use for bounding the advantage of some states relative to others.

Lemma 5.4. *For nonnegative integers a, b and $\ell \in \{0, 1\}$ we have that:*

$$\phi(a + \ell, b + \ell, 1) \leq \phi(a, b, 0) + (\ell + w) \cdot \left(\frac{p}{1-p}\right)^{b-a+1}. \tag{5.16}$$

Proof. We first need to establish that for $c \in \{0, 1\}$

$$g_k(a + \ell, b + \ell, c) \leq g_k(a, b, c) + \ell \cdot \left(\frac{p}{1-p}\right)^{b-a+1} \tag{5.17}$$

and

$$g_k(a + \ell, b + \ell, 1) \leq g_k(a, b, 0) + (\ell + w) \cdot \left(\frac{p}{1-p}\right)^{b-a+1}. \tag{5.18}$$

Suppose that from state (a, b) Miner 1 follows the same strategy as state $(a+\ell, b+\ell)$. This is possible, as winning or capitulating depends only on the difference $b - a$. Let $\bar{g}_k(a, b, c)$ be the gain from playing this suboptimal strategy and $\bar{r}(a, b)$ the probability of winning from state (a, b). Clearly, we have:

$$g_k(a, b, c) \geq \bar{g}_k(a, b, c) = g_k(a + \ell, b + \ell, c) - \ell \cdot \bar{r}(a, b). \tag{5.19}$$

Given that $\bar{r}(a, b) \leq (p/(1-p))^{b-a+1}$ from [44, Lemma 1] we get the first inequality.

For the second, let $r(a, b)$ be the probability of winning from (a, b) with the optimal strategy. As before:

$$g_k(a, b, 1) = g_k(a, b, 0) + w \cdot r(a, b) \leq g_k(a, b, 0) + w \cdot \left(\frac{p}{1-p}\right)^{b-a+1}, \tag{5.20}$$

which we substitute in 5.17.

To complete the proof, we take limits in (5.17) and use the definition of ϕ. □

In [44] it was shown that for $p \leq 0.361$ FRONTIER is a best response, therefore

(for $w = 0$) there exists a ϕ such that for $g^\star = p$ (5.12) is satisfied, meaning that honest mining maximises the fraction of blocks added by Miner 1. By Lemma 5.2, increasing w can only strengthen this result, therefore for $p \leq 0.361$ and any $w \leq 1$ there exists ϕ that satisfies (5.12) for $g^\star = p + p(1-p)w$. Using this, we extend the potential to states (a, b, c) with $a > b + 1$:

$$\bar{\phi}(a,b,c) = \begin{cases} \phi(a,b,c) & \text{if } a \leq b+1 \\ a\lambda + b\mu + \kappa + c \cdot w & \text{otherwise} \end{cases}, \tag{5.21}$$

where $\lambda = \frac{(p-1)^2(1-pw)}{1-2p}$, $\mu = \frac{p\left(p-(p-1)^2 w\right)}{2p-1}$ and $\kappa = \frac{(p-1)p(pw-1)}{2p-1}$. The constants have been selected so that $\bar{\phi}(a,b,c) = p\bar{\phi}(a+1,b,c) + (1-p)(\bar{\phi}(a,b+1,c) - \hat{g}^\star)$ and $\bar{\phi}(b+1,b,c) = b+1+c\cdot w - \hat{g}^\star = \phi(b+1,b,c)$, for $\hat{g}^\star = p + p(1-p)w$ which is the honest gain. Doing this, we can use known results about ϕ for smaller states while having a tight enough closed form of $\hat{\phi}$ on states which are unlikely to be reached.

Following the notation of [44] we define $\bar{\phi}_M$ for when Miner 1 continues to mine, $\bar{\phi}_R$ when Miner 1 releases some blocks and $\bar{\phi}_C$ when he capitulates.

$$\begin{aligned} \bar{\phi}_M(a,b,c) &= p \cdot \bar{\phi}(a+1,b,c) + (1-p)(\bar{\phi}(a,b+1,c) - \hat{g}^\star) \\ \bar{\phi}_R(a,b,c) &= \bar{\phi}(a-b-1,0,0) + b + 1 + c \cdot w - \hat{g}^\star \\ \bar{\phi}_C(a,b,c) &= \max_{s=0,\ldots,b-1} \bar{\phi}(0,s,1). \end{aligned}$$

Theorem 5.2. *For every $p < 0.344$, there exists $w \geq 0$ large enough so that if every miner but one follows* Frontier(w)*, the best response of the remaining miner with hash power p is* Frontier*.*

Proof. We need to show that $\bar{\phi}$ is a valid potential and satisfies (5.15) for $\hat{g}^\star =$

$p + p(1-p)w$, or equivalently:

$$\bar{\phi}(a,b,c) = \max\{\bar{\phi}_M(a,b,c), \bar{\phi}_R(a,b,c), \bar{\phi}_C(a,b,c)\}$$

Lemma 5.5. *The potential $\bar{\phi}$ and $\hat{g}^\star = p + p(1-p)w$ satisfy recurrence (5.15) for $p \leq 0.344$ (root of the polynomial $-1 + 5p - 7p^2 + 3p^3 - p^4$) and some $w < 1$.*

Proof.

Claim 1. *For states (a,b,c) with $a < b+1$:*

$$\bar{\phi}(a,b,c) = \phi(a,b,c) = \max\{\bar{\phi}_M(a,b,c), \bar{\phi}_R(a,b,c), \bar{\phi}_C(a,b,c)\}.$$

In this case $\bar{\phi}_R(a,b,c) = -\infty$ and therefore $\bar{\phi}(a,b,c) = \phi(a,b,c)$ which satisfies (5.12) and (5.15) by definition, as releasing is only possible for $a \geq b+1$.

Claim 2. *For states (a,b,c) with $a > b+1$:*

$$\bar{\phi}(a,b,c) = \bar{\phi}_M(a,b,c) = \max\{\bar{\phi}_M(a,b,c), \bar{\phi}_R(a,b,c), \bar{\phi}_C(a,b,c)\}.$$

By definition, $\bar{\phi}(a,b,c) = \bar{\phi}_M(a,b,c)$. We have:

$$\bar{\phi}(a,b,c) - \bar{\phi}_R(a,b,c) = \frac{b(2-4p) + (1-p)(2-3p)(1-pw)}{1-2p} > 0.$$

Since $p \leq 0.344$ we know that $\phi(a,b,c)$ corresponds to the potential of the honest mining strategy, in the immediate release case. Therefore:

$$\bar{\phi}_C(a,b,c) = \max_{s=0,\ldots,b-1} \bar{\phi}(0,s,1) = \max_{s=0,\ldots,b-1} \phi(0,s,1) = \phi(0,0,1) \leq w \cdot \frac{p}{1-p}$$

as $(0,0,1)$ and $(0,0,0)$ are the only mining states and using Lemma 5.4 for the last

inequality. Also:

$$\begin{aligned}\bar{\phi}_R(a,b,c) &= \bar{\phi}(a-b-1,0,0)+b+1+c\cdot w-\hat{g}^\star\\ &\geq 1-p(1-p)w\\ &\geq w\cdot\frac{p}{1-p}\\ &\geq \bar{\phi}_C(a,b,c),\end{aligned}$$

for $p < 0.344$.

Claim 3. *For states* $(b+1,b,c)$*:*

$$\bar{\phi}(b+1,b,c)=\bar{\phi}_R(b+1,b,c)=\max\{\bar{\phi}_M(b+1,b,c),\bar{\phi}_M(b+1,b,c),\bar{\phi}_C(b+1,b,c)\}.$$

Exactly as before, we have $\bar{\phi}_R(b+1,b,c)=b+1+c\cdot w-\hat{g}^\star\geq\bar{\phi}_C(b+1,b,c)$. For $\bar{\phi}_M(b+1,b,c)$ we have:

$$\begin{aligned}\bar{\phi}_M(b+1,b,c) &= p\bar{\phi}(b+2,b,c)+(1-p)\left(\bar{\phi}(b+1,b+1,c)-g\right)\\ &\leq p((b+2)\lambda+b\mu+\kappa+c\cdot w)+(1-p)\left((b+1+c\cdot w)\frac{p}{1-p}-g\right),\end{aligned}$$

by the definition of $\bar{\phi}$ and Lemma 5.4. Then:

$$\begin{aligned}&\bar{\phi}_R(a,b,c)-\bar{\phi}_M(a,b,c)\\ &\qquad\geq b(1-2p)^2+(p-1)\left(p^3w-p^2(w+1)+p(2c\cdot w+5)-c\cdot w\right)+1.\end{aligned}$$

Since $b\geq 0$ we only need: $(p-1)\left(p^3w-p^2(w+1)+p(2c\cdot w+5)-c\cdot w\right)+1\geq 0$. Setting $w=1$, it holds for $p\leq 0.344$ for $c=0$ and $p\leq 0.442$ for $c=1$. For smaller values of p it is not necessary to set w to it's highest value. □

Now, we can use the equivalent of Lemma 5.3(whose proof has very minor differ-

ences) for the strategic release case to get that:

$$\hat{g}^\star = \lim_{k\to\infty} \frac{\hat{g}_k(a,b,c)}{k} \leq \lim_{k\to\infty} \frac{\bar{\phi}(a,b,c) + k\cdot(p + p(1-p)w)}{k} = p + p(1-p)w.$$

This shows that his gain is at most what he would get by playing FRONTIER, hence a best response. □

Through a procedure similar to Section 5.4.1, adjusting the LP for this case, we obtain the following graph for $d = 8$ for the minimum value of w required.

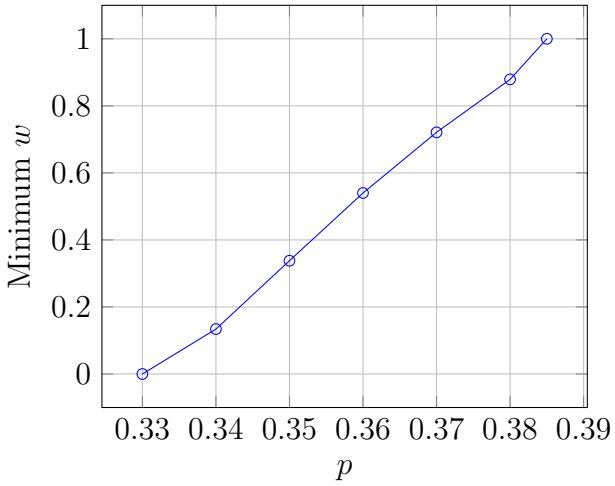

Figure 5.10: Minimum values of w for the strategic release case.

Contrary to Figure 5.8, for $p = 0.385$ we have $w \approx 1$. As $d \to \infty$, we reach $w = 1$ for $p \approx 0.38$.

5.6 Transaction Fees and Pay-Forward

Pay forward can also be used to eliminate the instability that will arise when block rewards are eliminated. As shown in [20], since blocks are not mined exactly every 10 minutes, there is high variance in rewards from transaction fees. Miners benefit by undercutting: mining one block behind the end of the chain and leaving some trans-

actions for the next miner, hoping to incentivise others to mine from their block. This leads to unstable behaviour, with an ever increasing sum of uncollected transactions caused by aggressive undercutting. To avoid this, we propose a small protocol change using pay-forward to even out the rewards.

Let us assume that transaction fees do not fluctuate significantly, and let's say that the average transaction reward per block is $f^\star$. We can try to simulate fixed block rewards as follows: Each miner is paid forward from the previous miner a value s, collects regular transaction fees f, and pays forward a value s', so his total reward from the block is $s + f - s'$. Assume that the suggested (honest) mining strategy is that the miner selects s' so that $s + f - s' = f^\star$. In this way, the reward for the block is fixed to $f^\star$, the same for all blocks. If a block does not have a total reward of exactly $f^\star$, it will be considered by the other miners as invalid and it will not be included in the blockchain. This protocol change ensures that all blocks will have the same reward.

For such a scheme to work we need to start with a relatively high initial buffer $s = B$, to guarantee that even if some successive blocks include almost no transaction fees, i.e., $f \approx 0$, there is still enough value in s to pay the miners. We start with a sufficiently large B (calculated given the transaction reward data from previous blocks) so that by concentration bounds with high probability every one of the next few thousand honest miners will get payment equal to $f^\star$. This is sufficient time for the protocol to set a new target block reward $f^\star$. This can be done in the same way that the crypto-puzzle difficulty is currently adjusted: at the beginning of each epoch, the value $f^\star$ is recomputed based on the transaction fees of the previous epoch. The calculation is such that assuming that the next epoch has the same transaction fees distribution as the previous and by taking into account the remaining pay-forward buffer B', the pay-forward amount at the end of the next epoch will be B in expectation, as shown in the Figure 5.11. This effectively splits the transaction

fees to all miners almost evenly and leads us back to the usual blockchain with block rewards.

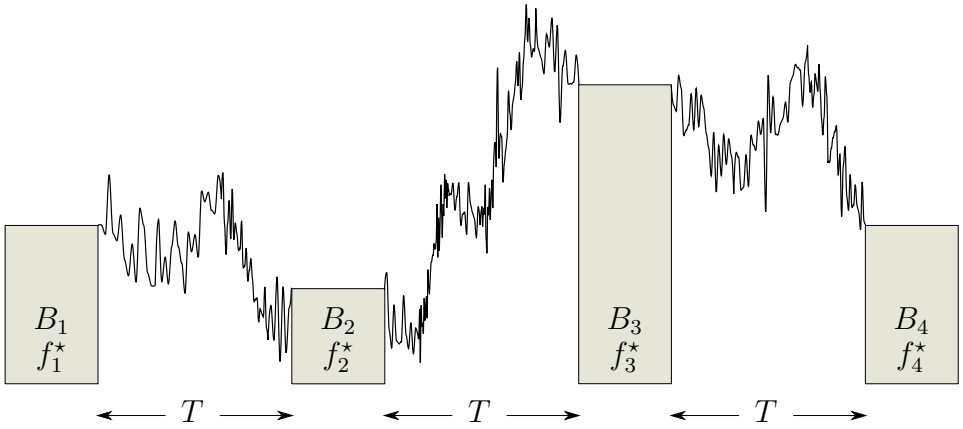

Figure 5.11: Buffer size and rescaling of f^* after each epoch.

5.7 Conclusions

We have studied the stochastic games that arise from the bitcoin protocol when miners pay forward some amount in order to incentivise others to mine from their block. In particular, we have studied two different scenarios, the immediate release and the strategic release case. We considered a simple class of pay-forward strategies in which the players with small hash power pay-forward a fixed amount. For the immediate release case, we prove that if the hash power of the biggest miner p is less than 0.5, then there is a pay forward amount w such that his best response is to play Frontier, given that the other miners also play Frontier and pay forward w. Similarly, for the strategic release case, we provide a proof that the threshold exceeds 0.344, and give computational results that put it higher at 0.38.

We have however, only analysed a small spectrum of strategies, where all players but one play Frontier(w), without showing which strategy profile is actually a PNE. Although playing Frontier(w) is enough to guarantee the proper execution of the protocol (ensuring that there is no strategic mining), it might still be the case that

small miners might prefer not to pay forward and every now let the strategic miner overtake their branch. The first step towards understanding this direction, is to try to minimise the amount paid forward by honest miners. Let w_k^i be the minimum amount needed to be added to the pay forward by the i-th consecutive honest miner, to make sure the strategic miner plays FRONTIER in the immediate release case. In Figure 5.8, we presented these results for w_1^1, where of course any other miner add 0. In practice however, the first miner might be tempted to pay forward a smaller amount and let the burden on the next honest miner. Using similar techniques, we can experimentally calculate w_2^i for different values of p.

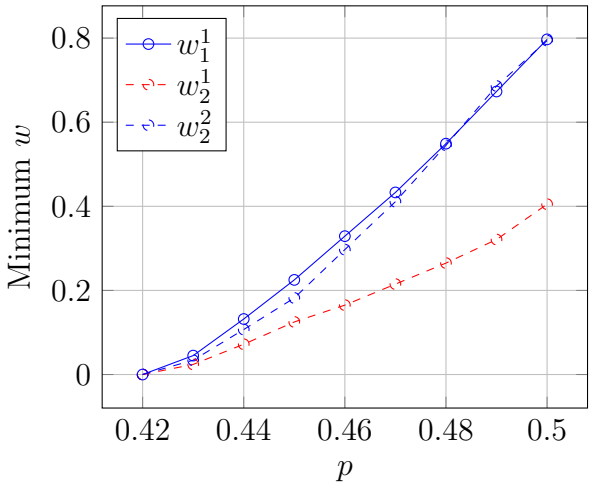

Figure 5.12: w_1^1, w_2^1 and w_2^2 for the immediate release case.

It seems that the first miner can pay significantly less, while the second about the same as before for the same result. Note however, that the entire set of honest miners pays significantly more overall. However, from the perspective of an honest miner with insignificant computational power, this makes more sense. Since it extremely unlikely that he would mine a second block, making sure he pays less to keep it might be beneficial. It would be very interesting to see how w_k^i evolved, for larger values of k. This result could be the first stepping stone towards finding a complete PNE for this enhanced blockchain mining game with pay forward.

We have also initiated the study of extending the Bitcoin protocol with pay forward to remove the instability that will arise when transaction fees will become the dominant source of income for miners. In our new protocol, miners *have to* pay forward an exact amount, to evenly distribute rewards. From a game theoretic perspective, the strategies of the players (and their payoffs) are exactly the same as the standard blockchain protocol used in practice and studied in [44]. An interesting research direction is to understand the mining games that arise in the proposed extension of the protocol, when players can pay forward any amount the want, as studied in Sections 5.4 and 5.5, but without coinbase rewards.

Chapter 6

Future Directions

We present what we consider to be the most fruitful avenues for future research, taking off where Chapters 3 and 4 left. Maintaining the connection to online algorithms, there are two main directions:

- Focusing more on the intermediary, it is perhaps too great of an assumption to consider markets where buyers and sellers are many, but all have to trade with the same central authority. Most likely, they would be receiving offers from many intermediaries, each trying to maximise their own revenue (having them compete for the social welfare does not sound particularly plausible). Specifically, at each step every one of m different intermediaries will simultaneously make an offer to the agent. Then, the agent will naturally select the most appealing offer. The payoff of each intermediary could be considered to be just his revenue or, perhaps more interestingly, his revenue divided by the total revenue generated. It is very interesting to see how the (mixed) strategies of each intermediary would evolve over time, as intermediaries who start accumulating money can afford to take greater risks or undercut their competitors. This setting is not particularly close to the prophet inequality and it would be more focused without the adversary, whose role is more unclear as there is no specific

global objective for him to undermine.

- To enhance the game theoretic nature of this setting, we can turn our attention to the most straightforward extension: just by including different item types. Agents are still assumed to be unit demand, but each seller would bring a different, unique item to the market. At each step, when an agent arrives he can express his value only for the previously revealed items and sellers could also be interested in buying some different item. In graph theoretic terms, the goal is to find a maximum weight matching in a bipartite graph, where at each step a vertex is revealed, along with a price. By paying this price we can 'buy' the node. Nodes that appear later will also contain the weight of incident edges to previously 'bought' nodes. This sufficiently changes the monotone structure of secretary type problems, where usually a decision is a local optimum to a global objective. In other words, in the classical setting (or in [43]) the only downside to buying an item (or matching an edge) is that a more valuable one later cannot be acquired. A first attempt suggests that it is unlikely that the competitive ratio (for either objective) of a truthful mechanism is bounded by a constant, but much more work is necessary. We think that this setting would be of significant interest to the online and mechanism design community.

Bibliography

[1] Saeed Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing*, 43(2):930–972, 2014.

[2] Barry C. Arnold and Richard A. Groeneveld. Bounds on Expectations of Linear Systematic Statistics Based on Dependent Samples. *The Annals of Statistics*, 7(1):220–223, jan 1979.

[3] Moshe Babaioff, Liad Blumrosen, Shaddin Dughmi, and Yaron Singer. Posting Prices with Unknown Distributions. In *Innovations in Computer Science (ICS)*, jan 2011.

[4] Moshe Babaioff, Yang Cai, Yannai A. Gonczarowski, and Mingfei Zhao. The best of both worlds: Asymptotically efficient mechanisms with a guarantee on the expected gains-from-trade. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 373–373, 2018.

[5] Moshe Babaioff, Shahar Dobzinski, Sigal Oren, and Aviv Zohar. On bitcoin and red balloons. *SIGecom Exch.*, 10(3):5–9, December 2011.

[6] Moshe Babaioff, Shaddin Dughmi, Robert D. Kleinberg, and Aleksandrs Slivkins. Dynamic pricing with limited supply. *ACM Trans. Economics and Comput.*, 3(1):4, 2015.

[7] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary

problems, and online mechanisms. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 434–443, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[8] Adam Back. Hashcash - a denial of service counter-measure, 1997.

[9] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Yaron Singer. Learning on a budget: posted price mechanisms for online procurement. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, pages 128–145. ACM, 2012.

[10] Ziv Bar-Yossef, Kirsten Hildrum, and Felix Wu. Incentive-compatible online auctions for digital goods. In *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '02, pages 964–970, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics.

[11] Richard E. Barlow and Albert W. Marshall. Bounds for Distributions with Monotone Hazard Rate, I. *The Annals of Mathematical Statistics*, 35(3):1234–1257, sep 1964.

[12] Avrim Blum and Jason D. Hartline. Near-optimal online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1156–1163. Society for Industrial and Applied Mathematics, 2005.

[13] Avrim Blum, Tuomas Sandholm, and Martin Zinkevich. Online algorithms for market clearing. *Journal of the ACM (JACM)*, 53(5):845–879, 2006.

[14] Liad Blumrosen and Shahar Dobzinski. (Almost) Efficient Mechanisms for Bilateral Trading. *arXiv preprint arXiv:1604.04876*, 2016.

[15] Liad Blumrosen and Thomas Holenstein. Posted prices vs. negotiations: An asymptotic analysis. In *Proceedings of the 9th ACM Conference on Electronic Commerce*, EC '08, pages 49–49, New York, NY, USA, 2008. ACM.

[16] Liad Blumrosen and Yehonatan Mizrahi. Approximating gains-from-trade in bilateral trading. In Yang Cai and Adrian Vetta, editors, *Web and Internet Economics*, pages 400–413, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[17] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In *2015 IEEE Symposium on Security and Privacy*, pages 104–121, May 2015.

[18] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[19] Johannes Brustle, Yang Cai, Fa Wu, and Mingfei Zhao. Approximating gains from trade in two-sided markets via simple mechanisms. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, EC '17, pages 589–590. ACM, 2017.

[20] Miles Carlsten, Harry Kalodner, S Matthew Weinberg, and Arvind Narayanan. On the instability of bitcoin without the block reward. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 154–167. ACM, 2016.

[21] Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 311–320. ACM, 2010.

[22] Edward H Clarke. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.

[23] Riccardo Colini-Baldeschi, Bart de Keijzer, Stefano Leonardi, and Stefano Turchetta. Approximately efficient double auctions with strong budget balance.

In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1424–1443. Society for Industrial and Applied Mathematics, 2016.

[24] Riccardo Colini-Baldeschi, Paul Goldberg, Bart de Keijzer, Stefano Leonardi, and Stefano Turchetta. Fixed price approximability of the optimal gain from trade. In Nikhil R. Devanur and Pinyan Lu, editors, *Web and Internet Economics*, pages 146–160, Cham, 2017. Springer International Publishing.

[25] Riccardo Colini-Baldeschi, Paul W. Goldberg, Bart de Keijzer, Stefano Leonardi, Tim Roughgarden, and Stefano Turchetta. Approximately efficient two-sided combinatorial auctions. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, EC '17, pages 591–608. ACM, 2017.

[26] Nicolas T. Courtois and Lear Bahack. On subversive miner strategies and block withholding attack in bitcoin digital currency. *CoRR*, abs/1402.1718, 2014.

[27] X Deng, P Goldberg, B Tang, and J Zhang. Revenue maximization in a bayesian double auction market. *Theoretical Computer Science*, 2014.

[28] Nedialko B. Dimitrov and C. Greg Plaxton. Competitive weighted matching in transversal matroids. *Algorithmica*, 62(1):333–348, Feb 2012.

[29] Michael Dinitz. Recent advances on the matroid secretary problem. *ACM SIGACT News*, 44(2):126–142, 2013.

[30] Cynthia Dwork and Moni Naor. Pricing via processing or combatting junk mail. In *Annual International Cryptology Conference*, pages 139–147. Springer, 1992.

[31] Eugene B Dynkin. The optimum choice of the instant for stopping a markov process. In *Soviet Math. Dokl*, volume 4, pages 627–629, 1963.

[32] I. Eyal. The miner's dilemma. In *2015 IEEE Symposium on Security and Privacy*, pages 89–103, May 2015.

[33] Ittay Eyal, Adem Efe Gencer, Emin Gün Sirer, and Robbert Van Renesse. Bitcoin-ng: A scalable blockchain protocol. In *NSDI*, pages 45–59, 2016.

[34] Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. *Commun. ACM*, 61(7):95–102, June 2018.

[35] Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial auctions via posted prices. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 123–135. Society for Industrial and Applied Mathematics, 2015.

[36] Moran Feldman and Rica Gonen. Online truthful mechanisms for multi-sided markets. *CoRR*, abs/1604.04859, 2016.

[37] Moran Feldman and Rica Gonen. Removal and threshold pricing: Truthful two-sided markets with multi-dimensional participants. In Xiaotie Deng, editor, *Proceedings of the Symposium on Algorithmic Game Theory (SAGT)*, pages 163–175. Springer, 2018.

[38] Matthias Gerstgrasser, Paul W Goldberg, and Elias Koutsoupias. Revenue maximization for market intermediation with correlated priors. In *International Symposium on Algorithmic Game Theory*, pages 273–285. Springer, 2016.

[39] Yiannis Giannakopoulos and Maria Kyropoulou. The vcg mechanism for bayesian scheduling. *ACM Transactions on Economics and Computation (TEAC)*, 5(4):19, 2017.

[40] Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[41] Mohammad Taghi Hajiaghayi, Robert Kleinberg, and Tuomas Sandholm. Automated online mechanism design and prophet inequalities. In *Proceedings of the*

22Nd National Conference on Artificial Intelligence - Volume 1, AAAI'07, pages 58–65. AAAI Press, 2007.

[42] Ari Juels and John G Brainard. Client puzzles: A cryptographic countermeasure against connection depletion attacks. In *NDSS*, volume 99, pages 151–165, 1999.

[43] Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In Hans L. Bodlaender and Giuseppe F. Italiano, editors, *Algorithms – ESA 2013*, pages 589–600, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[44] Aggelos Kiayias, Elias Koutsoupias, Maria Kyropoulou, and Yiannis Tselekounis. Blockchain mining games. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 365–382, New York, NY, USA, 2016. ACM.

[45] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 357–388, Cham, 2017. Springer International Publishing.

[46] Robert Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 630–631. Society for Industrial and Applied Mathematics, 2005.

[47] 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–, Washington, DC, USA, 2003. IEEE Computer Society.

[48] Robert Kleinberg and Seth Matthew Weinberg. Matroid prophet inequalities. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 123–136. ACM, 2012.

[49] Robert Kleinberg and Seth Matthew Weinberg. Matroid Prophet Inequalities. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 123–136, New York, NY, USA, 2012. ACM.

[50] Nitish Korula and Martin Pál. Algorithms for secretary problems on graphs and hypergraphs. In *Proceedings of the 36th Internatilonal Collogquium on Automata, Languages and Programming: Part II*, ICALP '09, pages 508–520, Berlin, Heidelberg, 2009. Springer-Verlag.

[51] Elias Koutsoupias and George Pierrakos. On the competitive ratio of online sampling auctions. *ACM Transactions on Economics and Computation*, 1(2):10, 2013.

[52] Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bull. Amer. Math. Soc.*, 83(4):745–747, 07 1977.

[53] Vijay Krishna. *Auction theory*. Academic press, 2009.

[54] Joshua A Kroll, Ian C Davey, and Edward W Felten. The economics of bitcoin mining, or bitcoin in the presence of adversaries. In *Proceedings of WEIS*, volume 2013, page 11, 2013.

[55] Yoad Lewenberg, Yoram Bachrach, Yonatan Sompolinsky, Aviv Zohar, and Jeffrey S. Rosenschein. Bitcoin mining pools: A cooperative game theoretic analysis. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pages 919–927, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems.

[56] R Preston McAfee. A dominant strategy double auction. *Journal of economic Theory*, 56(2):434–450, 1992.

[57] R. Preston McAfee. The gains from trade under fixed price mechanisms. *Applied Economics Research Bulletin*, 1(1):1–10, 2008.

[58] Roger B Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

[59] Roger B Myerson and Mark A Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 29(2):265–281, 1983.

[60] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system," http://bitcoin.org/bitcoin.pdf.

[61] Rad Niazadeh, Yang Yuan, and Robert Kleinberg. Simple and near-optimal mechanisms for market intermediation. In *International Conference on Web and Internet Economics*, pages 386–399. Springer, 2014.

[62] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[63] David C. Parkes. Online mechanisms. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani, editors, *Algorithmic Game Theory*, chapter 16. Cambridge University Press, New York, NY, USA, 2007.

[64] Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments. *draft version 0.5*, 9:14, 2016.

[65] Ronald L Rivest, Adi Shamir, and David A Wagner. Time-lock puzzles and timed-release crypto. 1996.

[66] Meni Rosenfeld. Analysis of bitcoin pooled mining reward systems. *CoRR*, abs/1112.4980, 2011.

[67] Meni Rosenfeld. Analysis of hashrate-based double spending. *arXiv preprint arXiv:1402.2009*, 2014.

[68] Aviad Rubinstein. Beyond matroids: Secretary problem and prophet inequality with general constraints. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 324–332. ACM, 2016.

[69] Ester Samuel-Cahn et al. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *the Annals of Probability*, 12(4):1213–1216, 1984.

[70] Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. Optimal selfish mining strategies in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 515–532. Springer, 2016.

[71] Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. SPECTRE: A fast and scalable cryptocurrency protocol. *IACR Cryptology ePrint Archive*, 2016:1159, 2016.

[72] Yonatan Sompolinsky and Aviv Zohar. Secure high-rate transaction processing in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 507–527. Springer, 2015.

[73] Yonatan Sompolinsky and Aviv Zohar. PHANTOM: A scalable blockdag protocol. *IACR Cryptology ePrint Archive*, 2018:104, 2018.

[74] F. Tschorsch and B. Scheuermann. Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys Tutorials*, 18(3):2084–2123, thirdquarter 2016.

[75] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.

[76] Qiqi Yan. Mechanism design via correlation gap. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11, pages 710–719. SIAM, 2011.