Segmenting Two-Sided Markets

Siddhartha Banerjee
Cornell University
sbanerjee@cornell.edu

Sreenivas Gollapudi
Google Research
sgollapu@google.com

Kostas Kollias
Google Research
kostaskollias@google.com

Kamesh Munagala
Duke University
kamesh@cs.duke.edu

ABSTRACT

Recent years have witnessed the rise of many successful e-commerce marketplace platforms like the Amazon marketplace, AirBnB, Uber/Lyft, and Upwork, where a central platform mediates economic transactions between buyers and sellers. A common feature of many of these two-sided marketplaces is that the platform has full control over search and discovery, but prices are determined by the buyers and sellers. Motivated by this, we study the algorithmic aspects of market segmentation via directed discovery in two-sided markets with endogenous prices. We consider a model where an online platform knows each buyer/seller's characteristics, and associated demand/supply elasticities. Moreover, the platform can use discovery mechanisms (search, recommendation, etc.) to control which buyers/sellers are visible to each other. We develop efficient algorithms for throughput (i.e. volume of trade) and welfare maximization with provable guarantees under a variety of assumptions on the demand and supply functions. We also test the validity of our assumptions on demand curves inferred from NYC taxicab log-data, as well as show the performance of our algorithms on synthetic experiments.

Keywords

online marketplaces, directed discovery, market segmentation

1. INTRODUCTION

Though markets are an ancient institution, they have been transformed in recent years by the rise of online marketplaces. Many of today's most important online marketplaces are platforms with both buyers and sellers. These include marketplaces for goods (Amazon, eBay, Etsy), and increasingly, for services: transportation (Lyft, Uber); physical and virtual work (Taskrabbit, Postmates, Upwork); lodging (Airbnb); shipping and delivery (Google Express, Amazon Fresh, Shyp); etc. Unlike traditional markets, these online platforms enable more fine-grained monitoring of participants, and more detailed control via pricing, terms of trade, visibility and directed search, information revelation and recommendation, etc. The challenge of harnessing this increase in data and control has led to a growing literature in the design of online marketplaces.

The focus of this work is on *the role of directed discovery mechanisms in matching demand and supply in modern online marketplaces*. In particular, we consider marketplaces exhibiting four characteristics:

i. Buyers/sellers are *horizontally differentiated* – each having heterogenous valuations and preferences for agents on the other side of the market.

ii. Each agent has a *public type*, i.e., a known list of characteristics. An agents' valuation is independently distributed conditioned on her public type.

iii. The platform has full control over *directed discovery* mechanisms – which sellers (buyers) are visible to a buyer (seller) on the other side of the market.

iv. *Transaction prices are endogenously determined by the agents*, based on who they are shown on the other side of the market.

These features are common in many online marketplaces. For example, in AirBnB, although listing prices are set by the 'hosts' (i.e., *sellers*), the platform can control which listings are visible to a 'guest' (i.e., *buyer*). Similarly in Upwork, the price of a job is determined via negotiation between contractors (buyers) and freelancers (sellers); the platform determines which job-postings are seen by which freelancers. In all these settings, the platform has side-information on an agents' characteristics (i.e., their public type), but is unaware of their exact utility except in aggregate.

Techniques to segment buyers via direct control of prices are studied in economics under the heading *price discrimination.* These techniques however do not extend to the settings illustrated above, where direct price control is not allowed. This motivates developing a theory for *search-based market segmentation*, where a platform can segment buyers and sellers in a market based on their types, but prices emerge endogenously (via market clearing) based on which segments have access to each other. We are interested in the algorithmic challenges of optimal segmentation of marketplaces via directed-discovery.

1.1 Our Model and Results

We now describe our setting in brief, and summarize our contributions. We present the formal model in Section 3

WWW 2017, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145/3038912.3052578

Agent types and valuations: We consider non-atomic agents, where each arriving buyer/seller belongs to one of a finite number of demand and supply *types*. For any demand-type i (or supply-type j), a continuum of agents arrive at the marketplace at a rate d_i (s_j). Moreover, for any given price p, the platform knows the fraction of arriving buyers (sellers) of that type willing to transact; these correspond to the supply and demand curves for each type. The supply curves are non-decreasing, while the demand curves are non-increasing.

In Section 2, we use the New York City Taxicab dataset to test our assumptions of heterogeneity in agent-valuation distributions across public types. We segment passengers (buyers) based on their trip destination, and use the presence of negotiated fares to estimate the demand curves from log data. Our imputed curves follow natural distributions, and are heterogenous across segments; this provides support for our model. Since estimating demand curves from log data is typically difficult due to the censored nature of the data and the lack of randomized controls, our analysis may be of independent interest.

Agent compatibility and directed discovery mechanisms: A buyer/seller's type also determines the set of compatible types on the other side of the market (i.e., types of agents with whom she can transact). For instance, on Upwork, the requirements of a job should be reasonably close to the skills of the freelancer; on AirBnB, the rough location the guest is searching for should be close to the location of the host's listing; and so on. As we will see below, our problem becomes interesting and challenging in setting where not all matches are feasible.

The platform only controls which buyer/seller types have access to each other in the marketplace; subsequently, transactions take place at endogenously defined *market-clearing prices* (cf. Section 3.1). The algorithmic challenge is to decide the *bipartite visibility graph* between agent types in order to maximize a given objective. The objectives we consider are *throughput* (i.e., rate of successful matches), and *welfare* (i.e., social surplus generated by the matches); the former is a proxy for revenue in settings where the platform receives a fixed commission per match.

Summary of our Results.

In the above setting, we want efficient algorithms to choose visibility graphs that (approximately) optimize throughput and welfare. The main challenge in doing so is that throughput and welfare are (approximately) super-additive [1] (cf. Section 4.1). For such functions, obtaining even a coarse approximation of the maximum objective is NP-Hard; in fact, we show in Theorem 4.3 that both throughput and welfare are NP-Hard to approximate beyond a constant multiplicative factor.

In view of the above, our aim is to find properties of the supply and demand curves that are plausible in practice, but moreover, admit polynomial-time approximation algorithms. To this end, we show the following:

- (Section 4.3) For general supply and demand curves, we provide algorithms that obtain a 4 approximation to the optimal segmentation for throughput, and a 3.164 approximation for welfare.
- (Section 4.5.1) For concave supply and demand curves, we provide a simpler algorithm that gets a 4 approximation for throughput and 8 approximation for welfare, and moreover, is oblivious to the exact curves, only needing to know the maximum supply or demand for each type.
- (Section 4.5.2) For identical and log-concave supply and demand curves, we provide a simple greedy algorithm which improves the approximation ratio for throughput to 3.164.
- (Section 5) Finally, we show via simulations the efficacy of pooling in markets compared to a natural baseline where supply is matched to demand under optimal uniform pricing.

A key technique underlying all our results is a common structural characterization (cf. Lemma 4.4): we show that for *any interaction graphs and arbitrary demand and supply curves*, there exists a 2-approximate solution where each segment either contains only one demand type or only one supply type. Using this result, we show that the throughput (resp. welfare) functions become sub-additive (resp. submodular) instead of being super-additive! This makes the problem tractable and amenable to existing techniques.

1.2 Related Work

Two-sided markets have a rich literature in Economics. There is a large body of work on two-sided *matching markets* [18, 3] where one/both sides of the market are assumed to be atomic, and prices can be set per agent. In contrast, marketplace platforms typically deal with a large mass of agents on both sides of the market, with market-clearing determining the equilibrium prices. Algorithmically, this makes the segmentation problem NP-Hard in our setting, while it is poly time if the price can depend on the matched agents.

Our work is closer in spirit to price-theoretic models of two-sided markets with non-atomic agents. Much of this work assumes however that prices can be set for different sides of the market with the goal of getting agents to report their types truthfully [12], maximizing social surplus or revenue of the platform [20], or modeling competing platforms [2, 16]. In contrast, we focus on *search discrimination* with endogenous prices, where the platform only controls which type of users on either side are visible to each other. This model is new to the best of our knowledge and is a relevant model for two-sided platforms in diverse contexts.

Though search in two-sided markets has been studied in the context of labor markets [9, 17] and dating platforms [13], these models do not have prices; instead they focus on how market segmentation determines probability of discovery, which in turn can lead to larger welfare.

In terms of algorithmic techniques, partitioning items into sets in order to maximize welfare is a well-studied problem, with constant factor approximation algorithms for subadditive [10] and submodular [19] welfare functions; for the latter, a classic greedy algorithm [15] gives a 2-approximation. Our functions are approximately super-additive, and in general, partitioning is a computationally hard problem. There has been work on developing good approximation algorithms for restricted classes of super-additive functions [11, 1]; however, our functions do not fall into these classes.

[1] Here, we consider throughput/welfare as set functions of the supply and demand types that are pooled together.

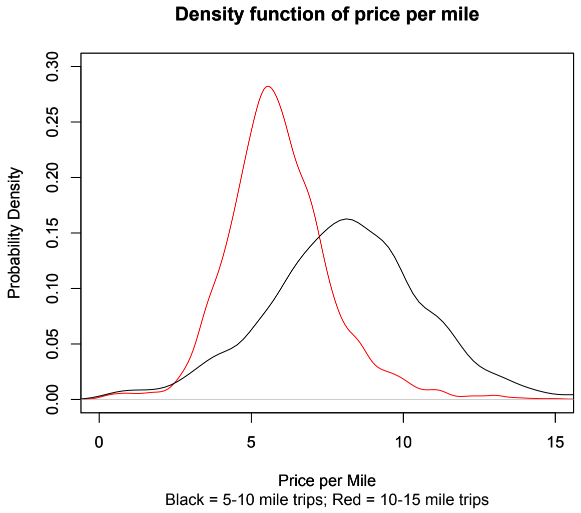

Figure 1: Density function of price-per-mile for different ranges of distance travelled.

2. MOTIVATING OUR PROBLEM

To provide some context for our modeling choices, we first use log data from a real-world setting to infer supply/demand curves which we show *a*) exhibit natural shapes within segments, and *b*) heterogeneity across segments. Estimating demand curves from log data is typically difficult due to data censoring (only successful transactions are recorded), fixed prices and the lack of randomized trials; and to this end, our analysis may be of independent interest.

We analyze large-scale log data for New York taxi cabs [2]. A brief description of the dataset: (refer [5] for more details): We use yellow taxi trip records for the months of January and June 2015 (thereby accounting for seasonal variations); each month consisting of around 12 million trip records. Each record comprises pick-up and drop-off times/locations, itemized fares and rate codes. Although most fares are based on set formulas, we exploit the presence of the *negotiated fares* [3]; these serve as a proxy natural experiment for estimating demand curves.

We restrict to rides originating in Manhattan, and use the trip destinations to be the public type of customers. We filter the data for each destination cluster to include records with at least 1 mile distance travelled, and price-per-mile $\in [0.5, 20]$. This removes spurious entries by the cab drivers as well as rides in extreme weather conditions.

To understand the demand function, in Figure 1 we bucket trips by distance traveled and plot the density function of price per mile for two different ranges of distances travelled: 10-15 miles and 5-10 miles. Unsurprisingly, the average price per mile is higher for shorter distances compared to the longer distances. More interestingly, the curves are close to Gaussian.

Next we analyze the yellow cab data from June 2015 for studying heterogeneity between different demand segments based on trip source and destination (which we take to be the public type of customers). We first perform k-center clustering of the sources and destinations in the trip records by greedily picking cluster centers and assigning all points within 10 miles geodesic distance to this cluster. Since most of the rides originate in Manhattan, we restrict our analysis to include only this source cluster. For each destination cluster with at least 50 destination points, we compute the mean distance travelled, mean price/mile and standard deviation of price/mile; this is shown in Table 1 for a sample of destination clusters.

Count	Mean	Stddev	Destination	Distance
116	4.83	1.55	Woodbridge, NJ	36.86
244	5.05	1.40	Morris Plains, NJ	34.69
711	5.24	2.48	JFK Airport	17.50
644	5.41	1.49	Irvington, NY	23.57
258	5.42	1.61	Plainview, Long Island	29.72
349	5.45	1.49	Glen Cove, Queens	20.49
328	5.57	1.99	South Orange NJ	25.94
418	5.99	1.55	Stamford CT	32.56
1128	6.20	2.16	Glen Rock, NJ	19.03
349	6.60	3.04	Fort Lee/GWB, NJ	12.96

Table 1: Price-per-mile differentiation between different destinations in the NYC cab system.

We observe that prices depend on two important factors: distance travelled and the destination cluster.

1. Very short rides (within NYC) have high price/mile.
2. Rides to Stamford, CT, have statistically higher prices than rides to New Jersey with comparable distance travelled. Similarly, rides to JFK are cheaper than comparable rides within NYC or to Queens.

In summary, this suggests that using destination as a public demand-type leads to heterogenous demand segments, with well-behaved demand curves.

3. PROBLEM FORMULATION

We consider the following problem setup. Each arriving seller belongs to one of n supply-types in set $\mathcal{S}$; similarly, each buyer belongs to one of m demand-types in set $\mathcal{D}$. Every agent's supply/demand type is public. We use *node* and *type* interchangeably.

Each supply type $j \in \mathcal{S}$ is associated with a non-decreasing *reservation-cost distribution* $F_j(p)$, which denotes the fraction of agents whose private cost of supplying (reservation cost) is at most p. Let s_j denote the arrival rate of agents of supply-type j; then the rate of type j agents at price p (i.e., the *supply curve*) is $s_jF_j(p)$. We assume F_j has associated density f_j, i.e., $F_j(p) = \int_{x=0}^{p} f_j(x)dx$.

Similarly, each demand type $i \in \mathcal{D}$ is associated with a non-increasing *demand curve* $d_iH_i(p)$, where d_i denotes the arrival rate of agents of this type, and $H_i(p)$ is the inverse CDF of the agents' reservation-value distribution (i.e., $H_i(p)$ is the fraction of agents of type i willing to transact at price p). We denote g_i as the associated density function, i.e., $H_i(p) = \int_{x=p}^{\infty} g_i(x)dx$.

An agent's type also determines the set of compatible types on the other side of the market; we encode this via an *interaction graph* $G(\mathcal{S} \cup \mathcal{D}, E)$, where an edge between $i \in \mathcal{D}$ and $j \in \mathcal{S}$ indicates the corresponding demand and supply types can potentially be matched. This can be thought of as

[2] http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

[3] These are defined by the NYC government to be 'out-of-town' trips with 'a Flat Rate negotiated between the Driver and Passenger'; in the dataset, they have RateCodeID 5.

modeling a $\{0,1\}$ preference between types, or alternately, an interaction cost of 0 or ∞ [4]; we describe later how some of our techniques extend to other models for interaction costs.

3.1 Equilibrium and Objectives

Given interaction graph $G(\mathcal{S} \cup \mathcal{D}, E)$, we assume the platform can realize any *visibility* subgraph by controlling which agent types see each other in the marketplace. The visibility subgraph then leads to equilibrium *market-clearing* prices. For ease of exposition, we restrict ourself to considering visibility subgraphs which partition the nodes into *demand/supply pools*, wherein all buyer and seller types are connected. Formally, a *pool* $\mathcal{P}$ is a subset of $\mathcal{S} \cup \mathcal{D}$ such that for every $i \in \mathcal{D} \cap \mathcal{P}$ and $j \in \mathcal{S} \cap \mathcal{P}$, we have $(i,j) \in E$. In the full version [5], we discuss extensions to general visibility subgraphs.

For any pool $\mathcal{P}$, we assume a separate market is created for the associated demand and supply types. The supply and demand in this pool equilibrate at a common *market-clearing* price $p(\mathcal{P})$, such that the supply is matched as much as possible to the demand

$$p(\mathcal{P}) = \operatorname{argmax}_p \min\left(\sum_{i \in \mathcal{D}\cap\mathcal{P}} d_i H_i(p), \sum_{j \in \mathcal{S}\cap\mathcal{P}} s_j F_j(p)\right)$$

In order to simplify notation, for a pool $\mathcal{P}$ we define

$$H_{\mathcal{P}}(p) = \sum_{i \in \mathcal{D}\cap\mathcal{P}} d_i H_i(p)\ ,\ F_{\mathcal{P}}(p) = \sum_{j \in \mathcal{S}\cap\mathcal{P}} s_j F_j(p)$$

Given the above equilibrium description, the marketplace directed search problem consists of designing a visibility subgraph that partitions $\mathcal{S} \cup \mathcal{D}$ into a set of *disjoint* pools $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_K$ in order to maximize one of two objective functions:

Throughput.

The throughput $\mathcal{T}(\mathcal{P})$ of the pool $\mathcal{P}$ is the rate of transactions in the pool at equilibrium.

$$\mathcal{T}(\mathcal{P}) = \min\left(H_{\mathcal{P}}(p(\mathcal{P})), F_{\mathcal{P}}(p(\mathcal{P}))\right)$$

Note that if the platform charges a fixed amount per match, the throughput is proportional to platform revenue.

Welfare.

The welfare $\mathcal{W}(\mathcal{P})$ of $\mathcal{P}$ is the sum of buyer and seller surplus for agents in pool $\mathcal{P}$

$$\mathcal{W}(\mathcal{P}) = \sum_{i \in \mathcal{D}\cap\mathcal{P}} d_i \mathbb{E}\left[(V - p(\mathcal{P}))^+\right] + \sum_{j \in \mathcal{S}\cap\mathcal{P}} s_j \mathbb{E}\left[(p(\mathcal{P}) - C)^+\right]$$

where V and C are the reservation value/cost of buyers and sellers respectively, and where $x^+ = \max(x,0)$. Equivalently, the welfare of a pool is the area under the intersection of the supply and demand curves:

$$\mathcal{W}(\mathcal{P}) = \int_{v=0}^{p(\mathcal{P})} F_{\mathcal{P}}(v)dv + \int_{v=p(\mathcal{P})}^{\infty} H_{\mathcal{P}}(v)dv \qquad (1)$$

Finally, given a partition of $\mathcal{S} \cup \mathcal{D}$ into pools $\{\mathcal{P}_i\}_{i\in[K]}$, the total throughput/welfare is given by

$$\text{Throughput } = \sum_{k=1}^{K} \mathcal{T}(\mathcal{P}_k)\ ,\ \text{Welfare } = \sum_{k=1}^{K} \mathcal{W}(\mathcal{P}_k)$$

[4] For example, a guest in AirBnB may search for listings only in a small area based on its distance from a central location.

4. POOLING ALGORITHMS

In this section, we present constant factor approximation algorithms for the case where the interaction graph and the demand and supply curves are either arbitrary or satisfy certain restrictions. The main technical challenge is that neither the throughput nor the welfare are sub-additive functions of the pool $\mathcal{P}$. In particular, we show below that for the complete interaction graph, the welfare is *always* super-additive, while throughput is *always* approximately super-additive (within a factor 2). For general super-additive functions, computing the optimal partitioning is a computationally hard problem, even in an approximation sense. To circumvent this problem, we consider a restricted type of pooling solution and show that this type of solution approximates the overall optimum. Once we perform this restriction, we show somewhat surprisingly that the problem becomes approximately sub-additive.

4.1 (Approximate) Super-additivity

We first show that welfare is always a super-additive function, while throughput is approximately super-additive when the interaction graph is complete.

Welfare.

For any pool $\mathcal{P}$, the welfare is given by Equation (1). We first show the following.

THEOREM 4.1. *If the interaction graph is a complete graph, then $\mathcal{W}(\mathcal{P})$ is a super-additive function of $\mathcal{P}$. Therefore, pooling all demand and supply nodes into one pool is optimal for welfare.*

PROOF. Since all demand curves are non-increasing while the supply curves are non-decreasing, it is easy to check that the area under the intersection of the curves can be written as a minimizer over prices p as:

$$\mathcal{W}(\mathcal{P}) = \min_p\left(\int_{v=0}^{p(\mathcal{P})} F_{\mathcal{P}}(v)dv + \int_{v=p(\mathcal{P})}^{\infty} H_{\mathcal{P}}(v)dv\right)$$

For each price p, the inner term is an additive function of $\mathcal{P}$. The minimum of additive functions is super-additive, completing the proof. □

Throughput.

We next consider the throughput objective on a complete interaction graph.

THEOREM 4.2. *On a complete interaction graph, merging all nodes into a single pool has throughput at least half of the throughput-optimal partition, and this bound is tight.*

PROOF. Let $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_k\}$ be any partition of a given set of nodes into pools with total throughput $\mathcal{T}$, and let $\mathcal{P}'$ be a single pool comprising of all the nodes. We will prove that $\mathcal{T}(\mathcal{P}') \geq \mathcal{T}/2$. Recall from Section 3.1, that the equilibrium price of a pool is the one that maximizes the throughput. Hence, it suffices to show that there exists some $\hat{p}$ for which

$$\min(H_{\mathcal{P}'}(\hat{p}), F_{\mathcal{P}'}(\hat{p})) \geq \mathcal{T}/2$$

Let p_k be the equilibrium price of pool $\mathcal{P}_k$ and suppose the pools have been named in non-decreasing price order.

Our choice for $\hat{p}$ is the price p_q of pool $\mathcal{P}_q$, where q satisfies

$$q = \min\left\{k \;\middle|\; \sum_{i \le k} \mathcal{T}(\mathcal{P}_i) \ge \frac{\mathcal{T}}{2}\right\}. \tag{2}$$

In simpler words, q is the smallest index such that all pools preceding and including the q-th one are responsible for at least half the throughput. Since, by definition, $\sum_{i \le q-1} \mathcal{T}(\mathcal{P}_i) < \mathcal{T}/2$, it follows that

$$\sum_{i \ge q} \mathcal{T}(\mathcal{P}_i) \ge \frac{\mathcal{T}}{2}. \tag{3}$$

Note that all buyers who buy in pools $\{\mathcal{P}_i\}_{i \ge q}$, are willing to buy in $\mathcal{P}'$ as well, since $\hat{p}$ is at most the equilibrium price of each of these pools. Similarly, all sellers who sell in pools $\{\mathcal{P}_i\}_{i \le q}$ are willing to sell in $\mathcal{P}'$ as well, since $\hat{p}$ is at least the equilibrium price of each of these pools. All these buyers contribute to $H_{\mathcal{P}'}(\hat{p})$ and all these sellers contribute to $F_{\mathcal{P}'}(\hat{p})$. Combining with (2) and (3), it follows that $H_{\mathcal{P}'}(\hat{p}) \ge \mathcal{T}/2$ and $F_{\mathcal{P}'}(\hat{p}) \ge \mathcal{T}/2$, which completes the proof.

To show that the factor 2 is tight, we construct the following example: Given $\epsilon > 0$, the market comprises two supply types with rates $s_1 = 1/\epsilon$ and $s_2 = 1$, and two demand types with rates $d_1 = 1$ and $d_2 = 1/\epsilon$. p_1, p, and p_2 are the set of possible prices, with $p_1 < p < p_2$. Both demand types have the same demand curve H, with $H(p_1) = 1$, $H(p) = \epsilon$, $H(p_2) = \epsilon$; similarly both supply types have the same supply curve F with $F(p_1) = \epsilon$, $F(p) = \epsilon$, and $F(p_2) = 1$. Thus, we have $d_1 H(p_1) = s_1 F(p_1)$, $d_2 H(p_2) = s_2 F(p_2)$ and $(d_1 + d_2)H(p) = (s_1 + s_2)F(p)$; thus p_1 is the equilibrium price for the pool that includes the first supply type and the first demand type (denoted pool $\mathcal{P}_1$), p_2 is the equilibrium price for the pool that includes the second supply type and the second demand type (pool $\mathcal{P}_2$), and p is the equilibrium point of the pool that includes all types (pool $\mathcal{P}$). Moreover, the throughput of pool $\mathcal{P}_1$ is $d_1 H(p_1) = 1$, of pool $\mathcal{P}_2$ is $d_2 H(p_2) = 1$, and for pool $\mathcal{P}$ is $(d_1 + d_2)H(p) = 1 + \epsilon$. Hence, the partition of types as $\{\mathcal{P}_1, \mathcal{P}_2\}$ induces throughput that is a $(1+\epsilon)/2$ fraction of the throughput of partition $\{\mathcal{P}\}$. This completes the proof. □

On the positive side, in Section 4.5.2, we show that for *log-concave* functions f and g, pooling is indeed throughput-optimal for a complete interaction graph.

Finally, we note that a reduction to an example given in [8] (where we replace agents by demands and items by supply) shows that it is NP-Hard to obtain better than a 16/15 approximation for throughput.

Theorem 4.3. *Even when the supply and demand curves are concave, it is* NP-Hard *to obtain better than a* 16/15 *approximation to throughput (resp. welfare).*

4.2 A Restricted Pooling Solution

Our main technical contribution in this work is to show that a *restricted class of pooling policies*, wherein each pool either has one supply node or one demand node, gives a 2 approximation for both welfare and throughput. In subsequent sections, we show how this structural result can be used to develop efficient approximation algorithms.

Formally, we define the following two set functions:

Definition 1. *For demand type $i \in \mathcal{D}$ and $S \subseteq \mathcal{S}$, let $\mathcal{T}_i(S)$ denote the throughput of the pool $\{i, S\}$, i.e.:*

$$\mathcal{T}_i(S) = \max_p \min\left(H_i(p), \sum_{j \in S} F_j(p)\right)$$

We refer to pool $\{i, S\}$ as a supply-pool *with* center i. *Similarly, we denote the welfare of $\{i, S\}$ by $\mathcal{W}_i(S)$.*

Definition 2. *For supply type $j \in \mathcal{S}$ and $D \subseteq \mathcal{D}$, let $\mathcal{T}_j(D)$ denote the throughput of the pool $\{D, j\}$, i.e.:*

$$\mathcal{T}_j(D) = \max_p \min\left(F_j(p), \sum_{i \in D} H_i(p)\right)$$

We refer to pool $\{D, j\}$ as a demand-pool *with* center j. *Similarly, we denote the welfare of $\{D, j\}$ by $\mathcal{W}_j(D)$.*

Our main structural result is the following:

Lemma 4.4. *Given arbitrary interaction graph G and demand and supply functions, for both welfare and throughput, there is a 2-approximate solution where either all pools are supply-pools, or all pools are demand-pools.*

Proof. For both throughput and welfare, the proof comprises three parts: (i) upper bounding the objective via an *LP relaxation*, (ii) characterizing a *structural property* of the optimal LP solution, and (iii) *constructing feasible pools* from the LP solution.

LP Relaxation.

First consider throughput. Let $\mathcal{P}$ be some pool in the optimal solution, with equilibrium price $p(\mathcal{P})$. We claim that the throughput of $\mathcal{P}$ can be computed by the following linear program $LP_{\mathcal{T}}$ [5]:

$$\begin{array}{rcll} LP_{\mathcal{T}} : \text{Maximize} & & \sum_{(i,j) \in E} f_{ij} & \\ \sum_{j | (i,j) \in E} f_{ij} & \le & H_i(p(\mathcal{P})) & \forall i \in \mathcal{P} \cap \mathcal{D} \\ \sum_{i | (i,j) \in E} f_{ij} & \le & F_j(p(\mathcal{P})) & \forall j \in \mathcal{P} \cap \mathcal{S} \\ f_{ij} & \ge & 0 & \forall (i,j) \in E \end{array}$$

This can be seen via a simple max-flow min-cut argument on a network with source node s, destination node t and a node for each demand type i and supply type j in $\mathcal{P}$; there exists an edge from s to each demand node i, with capacity $H_i(p(\mathcal{P}))$, and from each supply node j to t, with capacity $F_j(p(\mathcal{P}))$, and all edges from demand nodes to supply nodes exist and have infinite capacity.

Next, for welfare, consider any pool $\mathcal{P}$ in the optimal solution, with market-clearing price $p(\mathcal{P})$. For all supply and demand nodes in $\mathcal{P}$, define $b_i \triangleq d_i H_i(p(\mathcal{P}))$ and $a_j \triangleq s_j F_j(p(\mathcal{P}))$; since $p(\mathcal{P})$ is market-clearing, we have $\sum_{j \in S} a_j = \sum_{i \in D} b_i$. We also define the equilibrium surplus for each demand and supply node in $\mathcal{P}$ as

$$A_j = \int_{p=0}^{p(\mathcal{P})} s_j F_j(p) dp \ , \ B_i = \int_{p=p(\mathcal{P})}^{\infty} d_i H_i(p) dp$$

We now claim that the welfare of pool $\mathcal{P}$ can be upper

[5] Note that this is a thought experiment since we do not know the pools in OPT – however we can use it to infer the structure of the optimal solution.

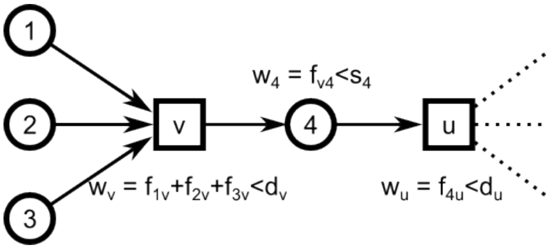

Figure 2: Converting the LP solution to feasible pools: Supply nodes $\{1, 2, 3, 4, \ldots\}$ are drawn as circles and demand nodes $\{v, w, \ldots\}$ as squares. Here, we orient the edges from the supply nodes $\{1, 2, 3\}$ to demand node v, then from v to supply node 4, and then 4 to u, assigning weights as described below. Subsequently, we find a maximum independent set in the resulting graph to obtain a feasible pooling.

bounded by the following linear program $LP_{\mathcal{W}}$:

$$\begin{array}{rcll} LP_{\mathcal{W}}: \text{Maximize} & & \sum_{(i,j)\in E} f_{ij}\left(\frac{A_j}{a_j} + \frac{B_i}{b_i}\right) & \\ \sum_{j|(i,j)\in E} f_{ij} & \leq & b_i & \forall i \in \mathcal{P} \cap \mathcal{D} \\ \sum_{i|(i,j)\in E} f_{ij} & \leq & a_j & \forall j \in \mathcal{P} \cap \mathcal{S} \\ f_{ij} & \geq & 0 & \forall (i,j) \in E \end{array}$$

This follows from a similar flow argument as above: first, since $\sum_{j\in S} a_j = \sum_{i\in D} b_i$, there is a flow f such that all inequalities above are tight. However, for this flow, the objective is simply $\sum_{j\in S} A_j + \sum_{i\in D} B_i$, which is exactly the welfare of $\mathcal{P}$.

Structure of LP Optimum.

Next, we observe that for *any* pool $\mathcal{P}$, the optimal solutions f^*_{ij} to both the above LPs admit a useful structural characterization:

LEMMA 4.5. *Let f^* denote the optimal solution to $LP_{\mathcal{T}}$ (or $LP_{\mathcal{W}}$); then the edges $(i, j) \in E$ with $f^*_{ij} > 0$ define a forest on $\mathcal{P} \cap (\mathcal{S} \cup \mathcal{D})$.*

The above lemma is adapted from a similar lemma for the maximum budgeted allocation problem [8]; essentially, it shows that given any cycle in the support of f^*, we can redistribute the flows to get an acyclic support while preserving the objective.

Finding a Feasible Pooling..

Finally, for both objectives, we show how to convert the LP solution f^* into a feasible pooling while losing a factor of 2. This follows an outline similar to [8, 14] for budgeted allocations.

First, for throughput, consider the forest output induced by f^* from $LP_{\mathcal{T}}$. Let T denote some tree in this forest; note that any path in this tree has alternate nodes from $\mathcal{S}$ and $\mathcal{D}$ respectively. Find a node $v \in T$ all but one of whose neighbors are leaves – we call these leaves the *children* of v and the remaining neighbor as the *parent* of v. Every tree with fractional f^*_{ij} has at least one such node. Since nodes in $\mathcal{S}$ and $\mathcal{D}$ are symmetric in $LP_{\mathcal{T}}$, w.l.o.g. assume $v \in \mathcal{D}$. We assign v a weight $w_v = \sum_{j \text{ child of } v} f^*_{jv}$, and orient the edges from the children of v to v and from v to the parent of v. Finally, we iterate on the remaining graph after removing v and its children (see Figure 2).

Note that the constraints of $L_{\mathcal{T}}$ imply $w_v \leq H_v(p(\mathcal{P}))$, and $f_{jv} \leq F_j(p(\mathcal{P}))$ for all children j; thus, a pool comprising v and its children has throughput at least w_v. At the end, all nodes except the original leaves of T have weights corresponding to the throughput of a pool formed by that node and its downstream neighbors in T [6]. Since each f_{ij} contributes to exactly one w_v, we have $\sum_{v\in T} w_v = \sum_{(i,j)\in T} f^*_{ij},$, which is exactly the objective of $LP_{\mathcal{T}}$. Moreover, either the nodes in $\mathcal{S} \cap \mathcal{P}$, or the nodes in $\mathcal{D} \cap \mathcal{P}$, have at least half the total weight. In the former case, we get a collection of supply-pools with at least 1/2 the value of $LP_{\mathcal{T}}$; in the latter case, we get a collection of demand-pools with at least 1/2 the value of $LP_{\mathcal{T}}$.

For welfare, let f^* be the solution to $LP_{\mathcal{W}}$. For any supply-pool (j, D') in $\mathcal{P}$ (i.e., $j \in \mathcal{P} \cap \mathcal{S}, D' \subseteq \mathcal{P} \cap \mathcal{D}$), we define the LP value of (j, D') as

$$LP(j, D') = \sum_{i\in D'} f^*_{ij}\left(\frac{A_j}{a_j} + \frac{B_i}{b_i}\right)$$

Similarly we define the LP value of demand-pools (i, S') with $i \in \mathcal{D} \cap \mathcal{P}, S' \subseteq \mathcal{P} \cap \mathcal{S}$. Now using exactly the same argument as for throughput, we have that $\mathcal{P}$ can be partitioned into a collection of either demand-pools or supply-pools such that the total LP value of these pools is at least 1/2 the value of $LP_{\mathcal{W}}$. Suppose this is obtained by supply-pools: then for supply-pool (j, D')

$$LP_{\mathcal{W}}(j, D') = \frac{\sum_{i\in D'} f^*_{ij}}{a_j} \cdot A_j + \sum_{i\in D'} \frac{f^*_{ij}}{b_i} \cdot B_i$$

This is precisely the welfare of an alternate problem where the supply curve for node j is scaled down by factor $\frac{\sum_{i\in D'} f^*_{ij}}{a_j}$, and for each $i \in D'$, the demand curve is scaled down by factor $\frac{f^*_{ij}}{b_i}$. Note that all these factors are at most 1 since f^* is feasible for $LP_{\mathcal{W}}$. Therefore, the welfare of pool (j, D') is at least $LP_{\mathcal{W}}(j, D')$, so that the welfare of the pools we construct is at least 1/2 the value of $LP_{\mathcal{W}}$, which in turn is the welfare of pool $\mathcal{P}$ in the optimal solution. □

To summarize: Lemma 4.4 uses an LP-relaxation argument to show that for arbitrary interaction graphs G and demand and supply functions, and for both welfare and throughput, there is a 2-approximate solution comprising only supply-pools, or only demand-pools. Moreover, in Fig. 3, we show that LP-relaxation based pooling cannot yield better than a 3/2 approximation.

4.3 General Demand and Supply Curves

We show how Lemma 4.4 can be used to get approximation algorithms for general supply and demand curves. In particular, we present a 3.164 approximation for welfare, and a 4 approximation for throughput.

4.3.1 *Welfare and Submodularity*

For welfare maximization, in addition to Lemma 4.4, we need an additional technical lemma:

LEMMA 4.6. *Let $\mathcal{W}(S, D)$ denote the welfare of pool $\mathcal{P} = S \cup D$, where $S \subseteq \mathcal{S}$ and $D \subseteq \mathcal{D}$.*

[6] We say that u is downstream of w if the edge (u, w) is oriented from u to w.

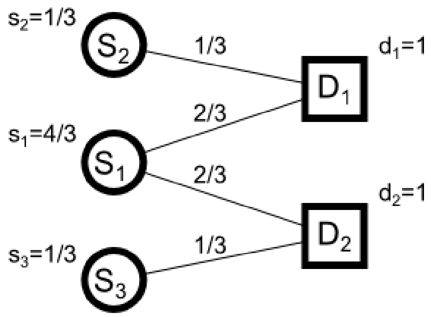

Figure 3: Lower bound on pooling via LP relaxation: Consider the depicted setting with supply types $\{S_1, S_2, S_3\}$, demand types $\{D_1, D_2\}$, and capacities $\{s_1, s_2, s_3, d_1, d_2\}$. The LP solution, shown on the edges, has value 2; the only feasible solutions are to either match both D_1 and D_2 to S_1 (value = $4/3$) or match D_1 to S_1 and S_3 to D_2 (value = $4/3$). The gap between any LP relaxation and feasible solution is therefore at least $3/2$.

1. *For fixed S, $\mathcal{W}(S, D)$ is a non-decreasing submodular function of D.*

2. *For fixed D, $\mathcal{W}(S, D)$ is a non-decreasing submodular function of S.*

Proof. We prove the first part; the second part is symmetric. Consider sets $D_1 \subset D_2$, and $t \notin D_2$. Let p_1 denote the clearing price of $S \cup D_1$, p_2 that of $S \cup D_1 \cup \{t\}$; p_3 that of $S \cup D_2$, and p_4 that of $S \cup D_2 \cup \{t\}$. Note that $p_1 \leq p_2$, and $p_3 \leq p_4$. Now we have two cases:
(i) If $p_2 \leq p_3$, then

$$\mathcal{W}(S, D_1 \cup \{t\}) - \mathcal{W}(S, D_1) \geq \int_{p=p_2}^{\infty} d_t H_t(p) dp, \text{ and}$$
$$\mathcal{W}(S, D_2 \cup \{t\}) - \mathcal{W}(S, D_2) \leq \int_{p=p_3}^{\infty} d_t H_t(p) dp$$

Since $p_3 \geq p_2$ by assumption, the second term is at most the first, showing submodularity.
(ii) If $p_2 > p_3$, then consider the difference:

$$M = \big(\mathcal{W}(S, D_2) - \mathcal{W}(S, D_1)\big) - \big(\mathcal{W}(S, D_2 \cup \{t\}) - \mathcal{W}(S, D_1 \cup \{t\})\big)$$

Note that

$$H_{(S,D_2)}(p) - H_{(S,D_1)}(p) = H_{(S,D_2 \cup \{t\})}(p) - H_{(S,D_1 \cup \{t\})}(p)$$

We have that M is at least:

$$M \geq \int_{p=p_3}^{\infty} \big(H_{(S,D_2)}(p) - H_{(S,D_1)}(p)\big)\, dp - \int_{p=p_2}^{\infty} \big(H_{(S,D_2)}(p) - H_{(S,D_1)}(p)\big)\, dp$$

Since $p_2 > p_3$, the above term is non-negative, which again shows submodularity. □

We now combine Lemmas 4.4 and 4.6: Lemma 4.4 proves there is a 2-approximate solution to the welfare where all pools are either demand-pools or supply-pools – we can thus find approximately-optimal solutions of each type and take the better of the two. We show how to find an approximate solution comprising of supply-pools – the process is symmetric for demand-pools.

By Lemma 4.6, $\mathcal{W}_i(S)$ is a non-decreasing submodular function of S. Thus, the problem of maximizing welfare via supply-pools is equivalent to splitting the supply nodes $\mathcal{S}$ among $i \in \mathcal{D}$ so as to maximize $\sum_i \mathcal{W}_i(S_i)$. This is isomorphic to the problem of *welfare-maximizing resource allocation with indivisible items and submodular utility functions* [19]. This problem has a known $e/(e-1)$ approximation via a continuous greedy algorithm [19], and a 2 approximation via the standard greedy algorithm [15]. Putting things together, we have:

Theorem 4.7. *For arbitrary graphs, and general demand and supply curves, there is an efficient $2e/(e-1)$ approximation to the welfare-optimal solution. Furthermore, a natural greedy algorithm gives a 4 approximation.*

4.3.2 Throughput and Sub-additivity

We next present a pseudo-polynomial time algorithm that computes a 2 approximation to the optimal throughput with supply-pools, and similarly, with demand-pools; we can then take the maximum of these two solutions. Combined with Lemma 4.4, this yields a 4 approximation to the optimal throughput.

Suppose demand nodes are labelled $\mathcal{D} = \{1, 2, \ldots, n\}$. We assume transaction prices are drawn from a discrete set of possible prices $\mathcal{Q}$, and further, $d_i H_i(p)$ is an integer in $[0, R]$, and similarly, for $j \in \mathcal{S}$, the value $s_j F_j(p)$ is an integer in $[0, R]$. The goal is to find a partition of $\mathcal{S} \cup \mathcal{D}$ into supply-pools $(1, S_1), (2, S_2), \ldots (n, S_n)$, so as to maximize throughput $= \sum_{i \in [n]} \mathcal{T}_i(S_i)$. Our running time will depend polynomially on $|\mathcal{Q}|, R, m$, and n.

Lemma 4.8. *For any $i \in \mathcal{D}$, $\mathcal{T}_i(S)$ is a sub-additive function over sets $S \subseteq \mathcal{S}$.*

Proof. Note that for a supply pool (i, S)

$$\mathcal{T}_i(S) = \max_{p>0} \left[\min\left(d_i H_i(p), \textstyle\sum_{j \in S} s_j F_j(p)\right)\right].$$

For any p, the inner term is the minimum of a fixed value and an additive function in S, hence it is submodular in S, and hence subadditive. The max of a collection of subadditive functions is also subadditive, hence the outer maximization preserves subadditivity. □

The throughput maximization problem for supply-pools is thus isomorphic to *welfare-maximizing resource allocation with indivisible items and subadditive utility functions.* The best known approximation for this problem is a 2 approximation [10], given an efficient algorithm for the following *demand oracle* problem:

Given shadow prices q_j for $j \in \mathcal{S}$, find $i \in \mathcal{D}$ and $S \subseteq \mathcal{S}$ that maximizes

$$\max_{p \in \mathcal{Q}} \left[\min\left(d_i H_i(p), \sum_{j \in S} s_j F_j(p)\right)\right] - \sum_{j \in S} q_j$$

The above maximization problem can be solved by dynamic programming in time poly$(n, R, |\mathcal{Q}|)$. Taking the maximum over all $p \in \mathcal{Q}$ and all $i \in \mathcal{D}$ gives us:

Theorem 4.9. *For arbitrary demand and supply curves, there is a pseudo-polynomial time 4 approximation to the throughput-optimal pooling solution.*

4.4 General Visibility Graphs

In the discussion so far, we have assumed that the platform chooses visibility graphs such that each connected component (i.e., each pool $\mathcal{P}$) contains a complete bipartite graph. The algorithms we present, however, achieve the same approximation with respect to more general visibility graph. In particular, given any visibility subgraph, the transactions at equilibrium result in flows f_{ij} between $i \in \mathcal{D} \cap \mathcal{P}$ and $j \in \mathcal{S} \cap \mathcal{P}$, which induce an *equilibrium flow graph.* Pools now correspond to connected components of this equilibrium flow graph, with each pool $\mathcal{P}$ corresponding to a market clearing price $p(\mathcal{P})$ and flows $\{f_{ij}\}$ that satisfy:

$$\forall i \in \mathcal{D} \cap \mathcal{P}: \quad \sum_{j \in \mathcal{S} \cap \mathcal{P}} f_{ij} = s_i H_i(p(\mathcal{P}))$$
$$\forall j \in \mathcal{S} \cap \mathcal{P}: \quad \sum_{i \in \mathcal{D} \cap \mathcal{P}} f_{ij} = d_j F_j(p(\mathcal{P}))$$

The existence of a single market clearing price in each component follows from the fact that in equilibrium, each supply node routes to the visible demand node with largest price, and each demand node routes to the visible supply node with smallest price – consequently, all transaction prices in a pool must be the same. Our structural characterization in Lemma 4.4 essentially compares against an optimal visibility graph (and resulting equilibrium) of this form.

4.5 Efficient Algorithms for Specific Curves

Finally, we show how the structural characterization in Lemma 4.4 admits efficient approximation algorithms under additional structure on the demand and supply curves.

4.5.1 Concave Supply and Demand Curves

First, we consider settings where both supply and demand curves are *concave*. We assume the prices p have domain $[0, p_{\max}]$; now, we have the following claim.

Claim 1. *Assuming the demand and supply curves are continuous and concave, then for any pool $\mathcal{P}$, there exist $\alpha_{\mathcal{T}}(P) \in [1/2, 1]$ and $\alpha_{\mathcal{W}}(\mathcal{P}) \in [1/4, 1]$ such that*

$$\mathcal{T}(\mathcal{P}) = \alpha_{\mathcal{T}}(P) \cdot \min\left(\sum_{i \in \mathcal{P} \cap \mathcal{D}} d_i, \sum_{j \in \mathcal{P} \cap \mathcal{S}} s_j\right)$$
$$\mathcal{W}(\mathcal{P}) = \alpha_{\mathcal{W}}(P) \cdot p_{\max} \cdot \min\left(\sum_{i \in \mathcal{D} \cap \mathcal{P}} d_i, \sum_{j \in \mathcal{S} \cap \mathcal{P}} s_j\right)$$

The proof of this claim follows from a simple geometric argument, illustrated in Figure 4 (cf. [5] for details). Using this, we can replace $H_i(p(\mathcal{P}))$ with d_i and $F_j(p(\mathcal{P}))$ with s_j in the programs $LP_{\mathcal{T}}$ and $LP_{\mathcal{W}}$. Combining with Lemma 4.4, we now get an efficient approximation algorithm which moreover is *oblivious* to the exact demand and supply functions, and only requires knowledge of $\{s_j\}$ and $\{d_i\}$:

Theorem 4.10. *When all supply and demand curves are concave, there is a 4 approximation to throughput, and an 8 approximation to welfare. Moreover, the algorithms are* oblivious *to the exact shapes of the curves.*

4.5.2 Identical Log-Concave Curves

Finally, we show that a simple greedy algorithm is a 4 approximation when we have identical *Log-Concave* demand and supply curves [4]. Formally, we assume that $F_i = F$ for all demand-types $i \in \mathcal{D}$, and $H_j = H$ for all supply-types $j \in \mathcal{S}$; further, we assume the corresponding density functions f and g are continuous and log-concave. This is relevant since many commonly encountered density functions such as Exponential, Uniform, Gamma, Poisson, Gaussian, etc. are log-concave [7].

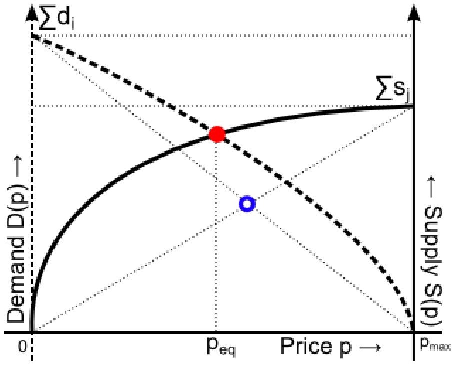

Figure 4: Potential demand $D(p) = \sum_{i \in \mathcal{P} \cap \mathcal{D}} d_i H_i(p)$ (dashed curve) and supply $D(p) = \sum_{j \in \mathcal{P} \cap \mathcal{S}} s_j F_i(p)$ (solid curve) in pool $\mathcal{P}$ as a function of price. The market-clearing price p_{eq} and match rate $\mathcal{T}(\mathcal{P})$ correspond to the solid circle at their intersection. Since $D(p)$ and $S(p)$ are concave, they lie above the lines joining 0 and $\sum_{i \in \mathcal{P} \cap \mathcal{D}} d_i, \sum_{j \in \mathcal{P} \cap \mathcal{S}} s_j$ (respectively for demand and supply); their intersection, indicated by the hollow circle, is less than $\mathcal{T}(\mathcal{P})$, and also equals $\alpha \cdot \min\left(\sum_{i \in \mathcal{P} \cap \mathcal{D}} d_i, \sum_{j \in \mathcal{P} \cap \mathcal{S}} s_j\right)$ for some $\alpha \in [1/2, 1]$. Similarly, for welfare, note that the area under the demand and supply curves includes the area of the triangle formed by points $(0,0)$, $(p_{\max}, 0)$, and the hollow circle.

We again use Lemma 4.4, and take the better of pooling solutions based on demand-pools, and on supply-pools. Recall that the throughput functions $\mathcal{T}_i(S), \mathcal{T}_j(D)$ are subadditive for general curves 4.3. In this setting however, we have a stronger result:

Lemma 4.11. *Suppose the density functions for the supply side (respectively demand side) are identical and log-concave. Then, for any $i \in \mathcal{D}$, $\mathcal{T}_i(S)$ is submodular in $S \subseteq \mathcal{S}$ and for any $j \in \mathcal{S}$, $\mathcal{T}_j(D)$ is submodular in $D \subseteq \mathcal{D}$.*

Proof. Consider some set $S \subseteq \mathcal{S}$ and $i \in \mathcal{D}$ and let $t = \frac{\sum_{j \in S} s_j}{d_i}$. Suppose S is assigned to i; then, the equilibrium price p is given by $tF(p) = H(p)$. Let $t = \alpha(p) = H(p)/F(p)$ and let $\beta = \alpha^{-1}$. Then the throughput as a function of t is simply $d_i H(\beta(t))$. Since i is fixed, showing $\mathcal{T}_i(S)$ is submodular in S follows by showing that the function $H(\beta(t))$ is concave in t.

First note that $\alpha(p)$ is decreasing in p since $H(p)$ is a decreasing function and $F(p)$ is an increasing function. The throughput $H(p)$ is also a decreasing function of p. Note that

$$\frac{dH(p)}{dt} = \frac{\frac{dH(p)}{dp}}{dt/dp} = \frac{g(p)(F(p))^2}{g(p)F(p) + f(p)H(p)}$$

To show submodularity, we need to show that the above expression is a decreasing function of t, but since t is decreasing in p, we need to show this is increasing in p. This means we need to show $1/F(p) + f(p)/(F(p))^2 \times H(p)/g(p)$ is decreasing in p. Since $F(p)$ is increasing in p, the first

[7] Our analysis suggests that the NYC taxi demand-curves are Gaussian, which are log-concave.

term is obvious. Since $f(p)$ is log-concave, $f(p)/F(p)$ is decreasing in p, and so is $f(p)/(F(p))^2$. Similarly, since g is log-concave, so is $H(p)$, so that $-g(p)/H(p)$ is decreasing in p, so that $g(p)/H(p)$ is increasing in p, so that $H(p)/g(p)$ is decreasing in p. This completes the proof that the throughput is concave in t, so that the throughput is a submodular set function.

A similar proof holds when supply types are the centers. Here, suppose for pool (D, i), the equilibrium price is given by $F(p) = xH(p)$, so that the throughput is $\mathcal{T}_j(D) = s_i F(p)$. Then, we need to show that $F(p)$ is a concave function of x. Note that

$$\frac{dF(p)}{dx} = \frac{\frac{dF(p)}{dp}}{dx/dp} = \frac{1}{\frac{1}{H(p)}\left(1 + \frac{g(p)}{H(p)}\frac{F(p)}{f(p)}\right)}$$

Since x is an increasing function of the equilibrium price p, we need to show that the above is decreasing in p. This follows since $H(p)$ is decreasing in p, while $g(p)/H(p)$ is increasing and $f(p)/F(p)$ is decreasing in p. □

As a consequence, we can use the ideas outlined in Section 4.3.1 to get:

THEOREM 4.12. *When supply and demand functions are identical and log-concave, there is a $2e/(e-1) \approx 3.164$ approximation to the throughput. Furthermore, a simple greedy algorithm yields a 4 approximation.*

Moreover, Lemma 4.11 also proves that pooling is indeed optimal for throughput when the interaction graph is a complete graph and the supply and demand curves are identical and log-concave (in contrast to Theorem 4.2, which shows it can be off by a factor of at most 2).

THEOREM 4.13. *For a complete interaction graph with identical log-concave demand and supply functions, throughput is maximized by pooling together all agent types.*

PROOF. Define $r(x) = F(p(x))$, where $p(x)$ is the solution to $xH(p) = F(p)$ – in other words, $r(x)$ is throughput with unit supply volume and demand volume x. Lemma 4.11 implies that $r(x)$ is concave in x

Consider any partition into more than one pools. In pool $\mathcal{P}_k$ the equilibrium price p_k satisfies $d_k H(p_k) = s_k F(p_k)$, which gives $d_k H(p_k) = s_k \cdot r\left(\frac{d_k}{s_k}\right)$. Then the total throughput over all pools is $r^{\mathrm{np}} = \sum_{k=1}^{K} s_k \cdot r\left(\frac{d_k}{s_k}\right)$.

Now consider a single pooled market with total demand volume $d = \sum_{k=1}^{K} d_k$ and total supply volume $s = \sum_{k=1}^{K} s_k$. The equilibrium price p satisfies $dH(p) = sF(p)$ and the throughput is $r^{\mathrm{p}} = s \cdot r\left(\frac{d}{s}\right)$. Then, using Jensen's inequality, we get

$$r^{\mathrm{np}} = s\sum_{k=1}^{K}\frac{s_k}{s}\cdot r\left(\frac{d_k}{s_k}\right) \leq s\cdot r\left(\sum_{k=1}^{K}\frac{s_k}{s}\frac{d_k}{s_k}\right) = r^{\mathrm{p}}.$$

This completes the proof. □

5. SIMULATIONS

In this section, we show results of our simulations to show how pooling performs with respect to the following natural baseline: All demand and supply nodes lie in a single pool; this pool has a uniform price; supply is matched to demand by computing a max-flow between them with values induced by the optimal uniform price. (See the Section 4.4 for a justification of uniform prices within a pool.) In our experiments we generate local markets, each characterized by one Gaussian supply and one Gaussian demand with an edge between them. For simplicity of exposition, each market i gets richer in the market clearing price as i gets larger, i.e., the Gaussian means are approximately i.

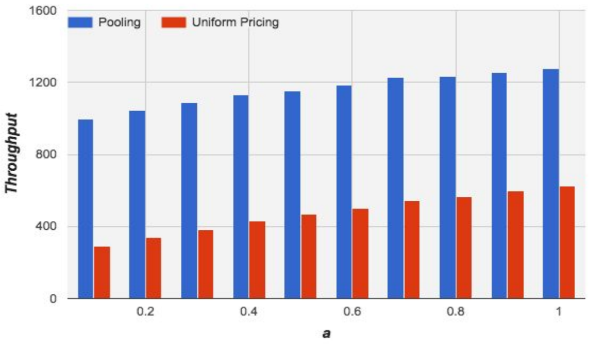

Figure 5: The relative performance of pooling compared to the baseline.

In order to make the graph admit matchings across markets, the supply i might also have an edge to demand j with probability $a/(|i-j|)$, for different values of a, which essentially controls the density of the graph. This models local bias in matching supply to demand. We created 40 markets using the above approach.

While the mean for supply/demand i is approximately i, we kept the variance to 1. The size of each supply and demand is drawn uniformly in $[1, 100]$. The resulting markets had an average degree $\in [2, 7]$. Figure 5 illustrates the relative performance of pooling.

6. CONCLUSION

While our work is a first step in the algorithmic study of two-sided platforms with non-atomic users, we list some open questions below:

- *Complex constraints on pools*: e.g., flow constraints in an electricity market with generators and consumers [7].
- *Incorporating heterogenous service-costs*: Our work extends to handle certain models for service costs; however handling general models of cost depending on supply and demand types is an open problem.
- *Posted prices with endogenous customer choice*: An alternate setting is to set a price for each pool, and allow agents to selfishly choose their optimal pool. This has applications in settings where each pool corresponds to a time interval, and the platform matches riders and rides with the goal of minimizing congestion (cf [12, 6] for related models). Do such settings admit to efficiently computable Walrasian equilibria?

7. ACKNOWLEDGMENTS

Banerjee acknowledges suppor from the ARL, under grant W911NF-17-1-0094. Munagala is supported by NSF grants CCF-1408784, CCF-1637397, and IIS-1447554.

8. REFERENCES

[1] I. Abraham, M. Babaioff, S. Dughmi, and T. Roughgarden. Combinatorial auctions with restricted complements. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, EC '12, pages 3–16, 2012.

[2] M. Armstrong. Competition in two-sided markets. *RAND Journal of Economics*, 37(3):668–691, 2006.

[3] E. M. Azevedo and J. D. Leshno. A supply and demand framework for two-sided matching markets. *Available at SSRN 2260567*, 2014.

[4] M. Bagnoli and T. Bergstrom. Log-concave probability and its applications. *Economic Theory*, 26(2):445–469, 2005.

[5] S. Banerjee, S. Gollapudi, K. Kollias, and K. Munagala. Segmenting two-sided marketplaces. *Available at SSRN 2920034*, 2016.

[6] O. Besbes and I. Lobel. Intertemporal price discrimination: Structure and computation of optimal policies. *Management Science*, 61(1):92–110, 2015.

[7] S. Bose, D. W. H. Cai, S. H. Low, and A. Wierman. The role of a market maker in networked cournot competition. *CoRR*, abs/1403.7286, 2014.

[8] D. Chakrabarty and G. Goel. On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and GAP. *SIAM Journal on Computing*, 39(6):2189–2211, 2010.

[9] P. A. Diamond. Aggregate Demand Management in Search Equilibrium. *Journal of Political Economy*, 90(5):881–94, 1982.

[10] U. Feige. On maximizing welfare when utility functions are subadditive. *SIAM Journal on Computing*, 39(1):122–142, 2009.

[11] U. Feige, M. Feldman, N. Immorlica, R. Izsak, B. Lucier, and V. Syrgkanis. A unifying hierarchy of valuations with complements and substitutes. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 872–878, 2015.

[12] R. Gomes and A. Pavan. Price discrimination in many-to-many matching markets. Technical report, Northwestern University, Center for Mathematical Studies in Economics and Management Science, 2011.

[13] H. Halaburda, M. J. Piskorski, and P. Yildirim. Competing by restricting choice: the case of search platforms. *Harvard Business School Strategy Unit Working Paper*, 10-098, 2015.

[14] J. K. Lenstra, D. B. Shmoys, and É. Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming*, 46(1):259–271, 1990.

[15] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

[16] J.-C. Rochet and J. Tirole. Platform Competition in Two-Sided Markets. *Journal of the European Economic Association*, 1(4):990–1029, 2003.

[17] R. Rogerson, R. Shimer, and R. Wright. Search-theoretic models of the labor market: A survey. *Journal of Economic Literature*, 43(4):959–988, 2005.

[18] L. S. Shapley and M. Shubik. The assignment game I: The core. *International Journal of Game Theory*, 1(1):111–130, 1971.

[19] J. Vondrak. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 67–74, 2008.

[20] E. G. Weyl. A price theory of multi-sided platforms. *The American Economic Review*, pages 1642–1672, 2010.