Alcom-FT Technical Report Series
ALCOMFT-TR-03-59

Computing Equilibria for Congestion Games with (Im)perfect Information*

Rene Beier† Artur Czumaj‡ Piotr Krysta§ Berthold Vöcking¶

Abstract

We study algorithmic questions concerning a basic microeconomic congestion game in which there is a single provider that offers a service to a set of potential customers. Each customer has a particular demand of service and the behavior of the customers is determined by utility functions that are non-increasing in the congestion. Customers decide whether to join or leave the service based on the experienced congestion and the offered prices. Following standard game theoretic assumptions, we assume each customer behaves in the most rational way. If the prices of service are fixed, then such a customer behavior leads to a pure, not necessarily unique Nash equilibrium among the customers. In order to evaluate marketing strategies, the service provider is interested in estimating its revenue under the best and worst customer equilibria. We study the complexity of this problem under different models of information available to the provider.

- First we consider the classical model in which the provider has perfect knowledge of the behavior of all customers. We present a complete characterization of the *complexity of computing optimal pricing strategies* and of *computing best and worst equilibria*. Basically, we show that most of these problems are inapproximable in the worst case but admit an "average-case FPAS." Our average case analysis covers general distributions for customer demands and utility thresholds. Furthermore, we generalize our analysis to robust equilibria in which players change their strategies only when this promises a significant improvement in the utility.
- We extend our analysis to a more realistic model in which the provider has *incomplete information*. Following the game theoretic framework of Bayesian games introduced by Harsanyi, we assume that the provider is aware of probability distributions describing the behavior of the customers and aims at estimating its expected revenue under best and worst equilibria. Somewhat counterintuitive, we obtain an FPRAS for the equilibria problem in the model with imperfect information although the problem with perfect information is inapproximable under the worst case measures. In particular, the worst case complexity of the considered stochastic equilibria problems increases with the precision of the available knowledge.

1 Introduction

We investigate computational aspects of a classical economic game in a market in which a single provider offers a service to a set of potential customers. We consider a selfish provider whose goal is to maximize its revenue. Each customer is assumed to have a particular demand of service and the quality of service decreases with the *congestion*, i.e., the sum of the served demands. We consider a model in which the customers do not cooperate with each other and the customer behavior is determined by *utility functions*. The utility functions are assumed to be non-increasing in the congestion and they specify whether a customer joins or leaves the service based on the offered prices and the experienced congestion. If all prices are fixed then such a customers behavior leads to an equilibrium among the

*Research partially supported by NSF grant CCR-0105701, by DFG grant Vo889/1-1 and by the IST program of the EU under contract number IST-1999-14186 (ALCOM-FT).

†Max-Planck-Institut für Informatik, Stuhlsatzenhausweg 85, 66123 Saarbruecken, Germany. rbeier@mpi-sb.mpg.de.
‡Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA. czumaj@cis.njit.edu.
§Dept. of Computer Science, Dortmund University, Baroper Str. 301, 44221 Dortmund, Germany. krysta@ls2.cs.uni-dortmund.de.
¶Dept. of Computer Science, Dortmund University, Baroper Str. 301, 44221 Dortmund, Germany. voecking@ls2.cs.uni-dortmund.de.

customers, commonly known as a *Nash equilibrium*. Therefore, from the perspective of the provider it is important to understand the behavior of the customers in Nash equilibria. In general, the provider may be interested in many possible scenarios. For example, if she is optimistic then she might aim at setting up prices such that the revenue under the *best customer equilibria* is maximized. A more pessimistic provider could be interested in maximizing the profit under *worst equilibria* or minimizing the *gap between best and worst equilibria* to minimize economic risk.

The model. In this paper we consider the following game that corresponds to the scenario described above. Suppose a provider wants to sell an on-line service to a set of n potential customers. Every customer i is assumed to have a *demand of service* w_i. For the time being let us assume that the provider committed to a particular *price vector*. Let p_i denote the *price* offered to customer $i \in [n] = \{1, \ldots, n\}$. Furthermore, let x_i denote an indicator variable that is 1 if customer i joins the service and 0 otherwise. Then, the *profit (revenue)* of the provider is defined as $\sum_i p_i x_i$. Due to congestion effects, the quality of service is assumed to be non-increasing with the *load*, which is the allocated demand $\sum_i w_i x_i$. Therefore, the utility of a customer is a non-increasing function of the load. For $i \in [n]$, let $u_i : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ denote the *utility function* of customer i mapping a load value to the maximum price that the customer is willing to pay for a service under this load. In this way, the *payoff* of a customer i is $u_i(\sum_j w_j x_j) - p_i$ in case he joins the service (i.e. $x_i = 1$), and is zero otherwise.

We assume a classical game theoretic setting in which a customer is willing to join the service if the payoff is non-negative, and he refuses the service otherwise. If the payoff is zero then the customer is ambivalent. An allocation of customers in which it is impossible for any single customer to improve his payoff is called a *customer Nash equilibrium*. In this paper we investigate two critical properties of customer Nash equilibria: which customers join the service and what is the profit obtained by the provider.

Let us observe that the region of allocations in an equilibrium is described by the following constraints:

$$\begin{aligned} x_i = 0 \quad &\Rightarrow \quad p_i \geq u_i(w_i + \textstyle\sum_j w_j x_j) \\ x_i = 1 \quad &\Rightarrow \quad p_i \leq u_i(\textstyle\sum_j w_j x_j) \\ &x_i \in \{0, 1\} \end{aligned}$$

More specifically, the equilibria defined by these constraints are called *pure equilibria*. If we relax the integrality and assume $x_i \in [0, 1]$ then we obtain so-called *mixed equilibria*. In this study, we will focus on pure equilibria.

Assuming that the prices are fixed, we can simplify the description of pure customer equilibria as follows. Since u_i are non-increasing functions, we can define *(utility) thresholds*[1] $b_i = u_i^{-1}(p_i)$, $i \in [n]$, and obtain

$$\begin{aligned} x_i = 0 \qquad &\Rightarrow \qquad w_i + \textstyle\sum_j w_j x_j \geq b_i \quad \text{(maximality constraint)} && (1) \\ x_i = 1 \qquad &\Rightarrow \qquad \textstyle\sum_j w_j x_j \leq b_i \qquad \text{(threshold constraint)} && (2) \\ &x_i \in \{0, 1\}. && (3) \end{aligned}$$

Observe that the problem of maximizing $\sum_i p_i x_i$ subject to the constraints above (the *best equilibria* problem) is a generalization of the maximum knapsack problem. The only difference is that in the knapsack problem all utility thresholds b_i are identical. Of course, in the typical formulation of the knapsack problem the maximality constraint is not specified explicitly as it is fulfilled implicitly by the profit maximization. However, if customers have different thresholds then the maximality constraint is crucial. The problem of minimizing $\sum_i p_i x_i$ subject to the constraints above is called the *worst equilibria* problem.

Congestion games. Assuming fixed prices, our game is called a *congestion game* in the literature. This class of games has been introduced by Rosenthal [31] and since then studied extensively (see, e.g., [7, 16, 24, 25, 26, 35]). In Rosenthals's model players choose a subset of available facilities, and each facility has an associated utility function. A payoff to a player from a facility is a function depending on the number of all players that choose this facility. Rosenthal shows that there always exists a pure strategy Nash equilibrium in his model. A model that is closer to our

[1] If function u_i is not strictly decreasing, we extend the definition of u_i^{-1} in the natural (obvious) way.

has been studied by Milchtaich [24, 25]. In a so-called *unweighted crowding (congestion) game*, every player has a player-specific payoff function that depends on the strategy he plays and on the number of players that choose the same strategy. In a *weighted crowding (congestion) game* each player i has a weight w_i, and the payoff function of a player depends on the sum of the weights of all players choosing the same strategy. In this terminology, our game is a weighted congestion game with player-specific payoff functions in which each player has two strategies. Milchtaich [24] shows that such games always admit a pure Nash equilibria, i.e., there exists a vector $x \in \{0,1\}^n$ (usually many such vectors) that fulfills the maximality and the threshold constraints defined above.

1.1 Summary of new results

The main theme of this paper is to understand the complexity of determining three important properties of the game described above: what is the best pricing scheme for the provider, which customers will join the service in customer best/worst Nash equilibria, and what is the provider's profit in best/worst Nash equilibria.

Basic model. We begin with the classical model in which the provider has perfect knowledge about the behavior of all customers. We provide the first thorough study of this model and present a complete characterization of the complexity of computing best and worst equilibria, and of computing optimal pricing strategies.

- We give an *FPAS*[2] for computing revenue maximizing prices assuming that the provider can offer an individual price to every customer independent of the customers demand.
- Individual pricing is often considered unrealistic. Economic literature usually assumes that prices are functions of the demands. Under this assumption, we show the existence of a *pseudopolynomial time algorithm*.
- We show that the pseudopolynomial time algorithm cannot be transformed into an FPAS. In fact, we prove that even under uniform pricing functions (i.e., prices are equal to demands) the problem is *inapproximable*[3]. Even for flat rate pricing (i.e., fixed price, same for all customers) the best possible approximation ratio is $\Omega(\sqrt{n})$.

In addition, we study more *robust equilibria* concepts that avoid thrashing effects. We show that even under these robust equilibria, most of the considered problems remain inapproximable. Only some special, somewhat restrictive cases, e.g., if all customers have the same utility threshold, admit efficient approximation algorithms.

Average case analysis. Despite the pseudopolynomial algorithm, most of the problems are inapproximable under the worst case measures. In economic literature one typically does not consider the worst case scenarios (which almost never appear in real life applications) but rather make some stochastic assumptions about the input. Therefore we turn our attention to *average case analysis* and our goal is to consider input distributions that are as general as possible.

The pseudopolynomial algorithms for the equilibria problems are based on a reduction to the interval knapsack problem corresponding to the standard 0/1 knapsack problem with an additional lower bound on the knapsack packing. This problem cannot be approximated in its general form as setting the interval length to some small number allows to decide the NP-hard subset-sum problem based on approximate solutions for the interval knapsack problem. We can prove, however, that there is an *adaptive approximation scheme* whose running time depends linearly on the reciprocal of the length of the interval. This adaptive algorithm is the basis for our average case analysis of the equilibria problem.

In our average case analysis, we consider a very general model, where the *demands and utility thresholds may have arbitrary, continuous probability distributions*[4] *with bounded mean and density*. Different demands and different thresholds can have different distributions. Our main contribution here is the design and the analysis of an approximation scheme whose expected running time depends on the maximum density and the maximum expectation over all probability distributions. To give a simple example of the performance of this approximation scheme, if

[2] FPAS or FPTAS stands for a *fully polynomial time approximation scheme*. Also, FPRAS means a *fully polynomial time randomized approximation scheme*.

[3] Throughout this paper, the term *"inapproximable"* means that for any polynomial time computable function $\alpha(n)$, the considered problem cannot be approximated within a factor of $\alpha(n)$ in polynomial time, unless $\mathsf{P} = \mathsf{NP}$.

[4] All our results can easily be generalized to discrete distributions as well.

all demands are uniformly sampled from $[0, 1]$ and all utility thresholds are sampled from $[0, n]$, then we obtain an "*average-case FPAS*." Observe, that under such a distribution, the ratio between the smallest and largest input numbers is only polynomial. This, however, by no means implies that the pseudopolynomial time algorithm can be used to obtain an approximate solution. The crucial property that gives us the average-case FPAS is that there are no small intervals in the interval knapsack problem to which we reduce the equilibria problems.

Models with incomplete information. Finally, following economic literature, we turn our attention to a more realistic model in which the provider has *incomplete information about the customer behavior*. Following the game theoretic framework of Harsanyi [14, 15], we assume that the provider is aware of probability distributions describing the customer behavior and aims at estimating its expected revenue under best and worst equilibria. On a first view, this Bayesian variant of the equilibria problem might seem even harder than the problem under perfect information. However, we can prove the existence of an *FPRAS* (fully polynomial time randomized approximation scheme) under some mild assumptions about the imperfect information. (Notice that the running time bound does not refer to the random input anymore but to the coin flips made by the algorithm.) In principle, our analysis covers general probability distributions with bounded domain. If the distributions are "well dispersing" then we achieve polynomial running time. The more "concentrated" the distributions are, the larger the running time becomes. Thus, somewhat surprisingly, the complexity of the equilibria problems increases with the precision of the available knowledge.

1.2 Previous and related work

Previous work in (economic) game theory that considered the same or similar to our models has been already discussed in the introduction under "congestion games." From the computational point of view there are known results on the complexity of computing equilibria in a parallel link game [8, 9], in normal form games and Bayesian games [3] (see also the references therein). These papers mostly prove that computing equilibria with some properties is hard. Complexity of equilibria in a market exchange game was studied in [4, 5]. For example Gairing et al. [9] have given an FPAS for computing the best equilibria in a simple parallel link game.

We now briefly discuss some previous work on algorithmic aspects of games with incomplete information. A classical model of auction design is given a seller who wants to sell a single (or multiple) item(s) to potential customers. Each customer has a private valuation of the item. The seller only knows a probability distribution over possible values of the valuations and her goal is to design a strategy (auction) of which customer gets the item and for what price in order to maximize her expected revenue. This simple model does not consider congestion issues. Ronen [29] has given a 2-approximate auction in this model. These results also hold in a Bayesian context. No deterministic polynomial time auction from a natural class of auctions can do better than a $4/3$-approximation, see Ronen and Saberi [30].

Several studies, e.g., [1, 2, 12], deal with average case analysis for the knapsack problem. In contrast to our work, these studies assume that profits are independent of the weights (i.e. demands in our terminology[5]), and they study exact algorithms. In our analysis, profits are functions of weights. In this case, one cannot hope to get exact algorithms, unless one breaks so-called knapsack crypto-systems whose hardness is based on the hardness of random subset-sum instances (i.e., knapsack with profits equal to weights) [18].

In the stochastic knapsack problem there are given n items with random weights with associated profits and the knapsack capacity. The objective is to find a knapsack packing that maximizes the expected profit or maximizes the profit under an additionally given overflow probability. Kleinberg et al. [19] have given a $O(1)$-approximation algorithm for general independent distributions. Goel and Indyk [11] improve on this result by giving approximation schemes for particular distributions, e.g., exponential or Bernoulli. Translating our result into this setting, we do not compute a particular knapsack packing but the expected profit of optimal knapsack packings. Other differences are that we assume that profits depend on weights (demands), and we consider harder problems which in contrast to the standard knapsack cannot be approximated within any factor if the weights are adversarial.

More detailed references especially to the related work in economics and game theory were either discussed before in the introduction or will appear further in the text.

[5]In our considerations related to knapsacks we will rather use the term weight instead of demand.

Outline. Sections 2 through 7 describe our results in more detail, and the most interesting proofs are presented in Section 8. The remaining proofs are shifted to the Appendix.

2 An FPAS for individual pricing

We first consider the simplest (but also the least realistic) model in which the provider can offer an individual price to every customer. Prices need not depend on the demand of the customer, that is, the provider is allowed to use an arbitrary price vector $p_1, \ldots, p_n$. Under such assumptions, the maximality constraint in the description of the customer equilibria can be avoided by assigning very high prices to those customers that are unwanted. As a consequence, for the optimal individual price vector there is only one equilibria and the provider does not have to distinguish between the best and worst equilibria. We assume that the utility functions $u_i(\cdot)$ are known and $u_i(\cdot)$ as well as $u_i^{-1}(\cdot)$ can be evaluated in constant time. In such a model we can provide a strong algorithmic result (see Appendix A for the proof).

Theorem 1 *Under individual pricing there is a price vector that maximizes the provider's profit and uniquely determines the behavior of each customer. Furthermore, there is an FPAS for approximating such an optimal price vector and for approximating the optimal equilibrium.*

Individual pricing, however, is usually considered to be an unrealistic scenario for an on-line service provider. Most studies in economics deal with *pricing schemes* in which prices are monotone functions of the demand, see, e.g., [13, 21, 27, 32, 33, 34].

3 Pseudopolynomial algorithms for more realistic pricing schemes

Economic literature focuses mostly on the following classes of pricing schemes: *uniform pricing*: a function of the form $p(w) = aw$, with $a > 0$ and w being the demand of a customer; *linear pricing*: $p(w) = aw+b$ with $a > 0, b \geq 0$; *block pricing*: p is a positive, piece-wise linear function; and *flat rate*: some constant function. In the context of an on-line service provider, block pricing seems to be the most relevant pricing scheme. In practice, customers are offered a small collection of so-called *pricing plans* (i.e., linear price functions) from which each customer can choose the best offer according to his demand.

Previous work in economics (see, e.g., [6, 13, 20, 21, 23, 27, 32, 33, 34]) focuses mostly on explicit analytical descriptions of equilibria. With this knowledge, pricing schemes are then computed basically by enumerating all possible pricing schemes from a given class of pricing schemes. However, the computation of the equilibria values of our congestion game is NP-hard and hence does not have such a closed form representation. Consequently, the provider needs more advanced algorithmic solutions in order to compute the value of worst and best equilibria. In fact, we can prove that there is a pseudo-polynomial time algorithm for computing best and worst customer equilibria under arbitrary price vectors. Recall that prices translate into thresholds, that is, the input consists of the thresholds $b_1, \ldots, b_n$, demands $w_1, \ldots, w_n$, and prices $p_1, \ldots, p_n$. All these values are assumed to be positive integers.

Theorem 2 *There is a pseudopolynomial time algorithm for computing the best (worst) customer equilibria for any given price vector.*

This algorithm is based on a similarity of the equilibria problem and the knapsack problem. In fact, we reduce the equilibria problem to a variant of the knapsack problem that we call *interval knapsack*, and then we show that the pseudopolynomial time algorithm for knapsack also works for the interval knapsack problem. The proof of this theorem is presented in Section 8.1.

The theorem has immediate consequences for the calculation of *optimal pricing schemes* for best or worst customer equilibria. For example, consider block pricing with a constant number of pricing plans. Obviously, there is only a pseudopolynomial number of ways to choose a constant number of different piece-wise linear functions. Thus, all block pricing schemes can be checked in pseudopolynomial time in order to determine which one is the best.

4 Inapproximability of equilibria and of pricing

Having a pseudopolynomial time algorithm for a problem, it is often possible to transform it into an FPAS. Unfortunately, this approach does not work for computing best and worst equilibria. We show that these problems are essentially inapproximable, even under uniform pricing, which basically means that prices are equal to demands. (The proofs of Theorem 3 and 5 are deferred to Appendix B.)

Theorem 3 *The best and worst customer equilibria problems are inapproximable under uniform pricing.*

This negative result implies that not only the best (worst) equilibria problem for an arbitrary uniform pricing function is inapproximable but so is the problem of computing best (worst) equilibria problem for the *optimal uniform pricing function*. To see this, suppose the input of the uniform pricing problem are integers $b_1, \ldots, b_n$ and demands $w_1, \ldots, w_n$. Recall, that uniform pricing is a pricing function $p_i = aw_i$ for some $a > 0$. Assume utility functions of the form $u_i(x) = w_i$ if $x \le b_i$ and 0 otherwise. Then, obviously, the optimal uniform pricing scheme is $p_i = w_i$ for all $i \in [n]$. An approximation of the revenue that can be obtained by these prices under best (worst) equilibria, however, gives also an approximation of the value of the underlying best (worst) equilibria itself and, hence, is not possible.

Corollary 4 *The best and worst customer equilibria problems are inapproximable even when the optimal uniform pricing function is given.*

Still, this inapproximability result leaves some hope that one might be able to compute the optimal pricing function without being able to approximate the revenue that it yields. In fact, in the above example the computation of the best pricing scheme itself was trivial. The following theorem shows that not only computing the optimal revenue but also computing the prices that lead to this revenue is difficult by itself. We restrict our attention to the simplest possible pricing scheme: flat rates and the best equilibria problem.

Theorem 5 *For a flat rate $a > 0$, let $P(a)$ be the profit of the best equilibria under this flat rate. Let opt $= \max_a P(a)$. A flat rate $a > 0$ with $P(a) \ge$ opt$/\sqrt{n}$ cannot be calculated in polynomial time, unless* $\mathsf{P} = \mathsf{NP}$.

5 (In)Approximability of robust equilibria

All inapproximability results above rely on a very careful and adversarial choice of the utility thresholds such that even a slight change of thresholds leads to thrashing effects. This raises the questions of how the complexity of the problem will change if we consider more "robust" definitions of equilibria. It is very reasonable to assume that customers move to a different strategy only if this promises a significant improvement in the payoff. This leads to *δ-robust equilibria* defined as follows. For some $\delta > 0$,

$$\begin{aligned} x_i = 0 \quad &\Rightarrow \quad w_i + \textstyle\sum_j w_j x_j \ge (1-\delta) b_i \\ x_i = 1 \quad &\Rightarrow \quad \textstyle\sum_j w_j x_j \le (1+\delta) b_i \\ &x_i \in \{0, 1\}. \end{aligned}$$

What are the effects of such a relaxation on the approximability of the equilibria problems? It turns out that the problems basically remain inapproximable even under robust equilibria. Only some special, somewhat artificial cases in which all customers have the same utility threshold become approximable now.

Theorem 6 *For any $0 \le \delta < \frac{1}{8}$, best and worst δ-robust equilibria are inapproximable (even under uniform pricing).*

Theorem 7 *If $b_1 = b_2 = \cdots = b_n$ then the worst δ-robust equilibria problem admits an FPAS if $\delta > 0$ and is inapproximable if $\delta = 0$ (assuming arbitrary choices of prices and demands).*

The proof of Theorem 6 can be seen as a "robust" variant of the proof of Theorem 3. The proof of Theorem 6 relies on an analysis of the *min-max knapsack* problem, i.e., the problem of packing a knapsack with minimum value such that the weight is maximal. We give a complete characterization of the complexity of this problem that might be of independent interest. Before it was only known that the min max knapsack problem is NP-hard [22]. Despite the similarity of the min-max knapsack problem to the standard knapsack problem, we prove that the *min-max knapsack problem is inapproximable* although the knapsack admits an FPAS. If the profits are equal to the weights, however, then there is an FPAS. The same holds if one introduces robustness. All these results and their proofs are moved to Appendix C.

6 Adaptive algorithms and average case analysis

We continue our analysis of interval knapsack. It is easy to see that in its general form the interval knapsack problem cannot be approximated because any algorithm that for arbitrarily small interval length approximates profits in polynomial time could be used to decide subset-sum instances. If the specified interval is large, however, then there is no direct relation to the hardness of subset-sum instances. In fact, we can give an approximation algorithm for interval knapsack whose running time adapts to the length of the interval. The proof of the next theorem is in Section 8.2.

Theorem 8 *For any $\epsilon \in [0,1]$, there is an $(1-\epsilon)$-approximation algorithm for the interval knapsack problem with interval boundaries L and R having running time $O\left(\frac{n^3 R}{(R-L)\epsilon} \log\left(\frac{\sum_i p_i}{P_{\min}}\right)\right)$, where $P_{\min} = \min_i(p_i)$.*

This result is a key to our average case analysis of the equilibrium problem for general probability distributions. We consider the scenario in which demands and thresholds have independent continuous probability distributions with density function f_i and g_i and distribution functions F_i and G_i ($i \in [n]$), respectively. Prices are assumed to be functions of the demands. We assume that the pricing function is non-negative, non-decreasing and concave. This is a very natural assumption since rational customers can always achieve concave pricing functions by splitting their demands. For this reason, basically all pricing schemes considered in the economic literature (see Section 3) are concave. Furthermore, we assume that the pricing function can be evaluated in constant time.

The running time of any efficient average case algorithm for the equilibrium problem must depend on the input probability distributions. Otherwise, one could bypass the inapproximability results presented in the previous sections by encoding worst case instances into distributions with very high density. In fact, the running of our algorithm increases with the maximum density of the underlying probability distributions. Let ϕ denote the maximum density over all distributions, i.e., $\phi = \max_i \max_{x>0}\{f_i(x), g_i(x)\}$. Let μ be the maximum expectation over all these distributions, that is, $\mu = \max_i\{\int_0^\infty x f_i(x)dx, \int_0^\infty x g_i(x)dx\}$. The proof of the next result can be found in Section 8.3.

Theorem 9 *Assume that the pricing function is a non-negative, non-decreasing, and concave function of demands, and ϕ and μ are as defined above. For any $\epsilon \in [0,1]$, there is an $(1-\epsilon)$-approximation algorithm for the best (worst) equilibria problem with expected running time poly$(n, \frac{1}{\epsilon}, \phi\mu)$.*

The term $\frac{1}{\phi\mu}$ can be seen as a measure of the amount of randomness available. (Observe that one can always scale the distributions so that $\mu = 1$; this scaling does not change the value of $\phi\mu$.) If all input variables have identical uniform distribution then $\phi\mu = \frac{1}{2}$, which is the smallest possible value. Next, suppose an adversary specifies possibly different Gaussian distributions (conditioned on positive values) for the utility thresholds and demands. Then μ (by definition) corresponds to the maximum expectation of any of these values and $\phi = \Theta(\frac{1}{\sigma})$, with σ denoting the minimum standard deviation over all these distributions. Thus the running time is poly$(n, \frac{1}{\epsilon}, \frac{\mu}{\sigma})$. Hence, the more dispersed the distributions are, the smaller the running time is.

We notice that these results are not directly related to the average case analysis for the knapsack problem in [1, 2, 12]. For details, see our discussion in Section 1.2.

7 Algorithms for imperfect information

Let us briefly summarize the results discussed so far. On one hand, our analyses show complete inapproximability for worst case instances. On the other hand, we have a very general average case analysis that yields efficient algorithms for various input distributions. In this section, we show that the latter result is the key to obtain efficient algorithms for estimating equilibria with *imperfect information*.

We follow the framework of Harsanyi [14, 15], who described a very elegant model to study games in which the players have incomplete information. The so-called *Harsanyi transformation*, one of the ideas for which he was awarded the Nobel Prize together with Nash and Selten, converts games with incomplete information into games with complete but imperfect information. Harsanyi considers players who have different utilities as being of different *types*. He proposed that such games be modeled by having Nature move first and choose each players type according to a probability distribution. Players initially know only their own type and the distributions for the other players, but not the outcome of the random choices by Nature. In this kind of game, players aim at maximizing the expected payoff.

If we consider the Harsanyi framework in our setting, there is only one player that needs to compute its strategy based on imperfect information. Indeed, in realistic scenarios it is reasonable to assume that the provider does not know exactly how the customers will behave under a given price vector. In contrast, customers can be assumed to converge to a Nash equilibrium by best response strategies without knowing the utilities of other players (this follows from analysis in [24, 25]). If prices are fixed, then the customer behavior is determined by demands and utility thresholds; these parameters define the Harsanyi types. The online provider does not know the types of the customers but she knows probability distributions on these types. In order to maximize its expected profit, she is interested in estimating the expected revenue under best and worst equilibria.

Somewhat counterintuitively, we prove that, if the information available to the provider is not too "precise," then the stochastic equilibria problems obtained by the Harsanyi transformation have smaller worst case complexity than their deterministic counterparts. Suppose demands and utility thresholds have independent continuous probability distributions with distribution functions F_i and G_i ($i \in [n]$), respectively. Prices are assumed to be defined by a non-negative, non-decreasing, concave function of the demands. We need some further technical assumptions:

(1) there exists M such that $F_i(M) = 1$ and $G_i(M) = 1$, and

(2) $\mathbf{Pr}\left[\exists i : w_i \leq b_i\right] \geq \frac{1}{2}$.

The first assumption means that the domain of the probability distributions of the demands and utility thresholds are bounded. This property holds for uniform distributions. For other distributions, one might need to cut the tails at some position where the probability becomes negligibly small. The second assumption is mild and seems to be natural as well. It says that the market for the offered service is non-empty with probability at least $\frac{1}{2}$.

Following a practitioner approach to obtain an estimation of a random variable, we sample from the distribution and use the adaptive algorithm from Theorem 9 to calculate the value of the best (worst) equilibria. Applying the assumptions (1) and (2) in a Hoeffding bound, yields that, for every $\beta, \epsilon > 0$, one needs only $O\left(\left(\frac{n}{\epsilon}\right)^2 \log\left(\frac{1}{\beta}\right)\right)$ samples in order to get a $(1 \pm \epsilon)$-approximation of the expected revenue. Now, the key argument is that sampling generates average case but not worst case instances. This way, we obtain an efficient randomized approximation algorithm with running time poly$(n, \frac{1}{\epsilon}, \phi\mu, \log(\frac{1}{\beta}))$.

In this scenario, the term $\frac{1}{\phi\mu}$ should not be interpreted as an indicator of how much randomization is available, but rather $\phi\mu$ should be interpreted as a measure for the degree of precision of the information that is available to the provider. If $\phi\mu$ is constant then the provider essentially knows nothing. Full knowledge corresponds to $\phi\mu = \infty$. If the degree of precision is bounded by a polynomial, e.g., if all variables have possibly different uniform distributions, each of which covering a polynomial fraction of the domain $[0, M]$, then we obtain an FPRAS (fully polynomial time randomized approximation scheme).

Theorem 10 *Suppose assumptions (1) and (2) are satisfied and $\mu\phi$ is bounded polynomially. Then there is an FPRAS for the best (worst) equilibria problem with imperfect information.*

Observe, that by standard techniques we can achieve deterministic polynomial time instead of time bounds that only hold on expectation. Then, only the approximation guarantee depends on the random coin flips made by the algorithm.

8 Algorithms and proofs

8.1 Reduction to the interval knapsack problem

In this section we present a reduction from the customer equilibria problems to the interval knapsack problem. It will serve as the basis for solving the equilibria problem in pseudo-polynomial time and approximately in polynomial time. We define the interval knapsack problem to be the standard 0/1 knapsack problem with an additional lower bound for the weight of knapsacks: $\max\{\sum_{i=1}^n p_i x_i \,|\, L \leq \sum_{i=1}^n w_i x_i \leq R, x_i \in \{0,1\}\}$. (A knapsack is feasible if its weight $\sum_{i=1}^n w_i x_i$ falls into the interval $[L, R]$.) The minimization version asks for the minimum profit knapsack. Notice, that the interval knapsack problem is inherently inapproximable, because for $L = R$ it requires to decide the subset-sum problem. On the other hand, the problem can be solved in pseudopolynomial time $O(n \sum_{i=1}^n w_i)$ by a straightforward adaptation of the well known dynamic programming approach for the 0/1 knapsack problem. Thus, the following lemma directly implies Theorem 2.

Lemma 11 *A solution to the best (worst) customer equilibria problem can be obtained by solving at most $2n-1$ interval knapsack problems.*

Proof. Let us recall the formulation of the equilibria problem: our goal is to maximize (minimize) $\sum_{i=1}^n p_i x_i$ under constraints (1) — (maximality condition), (2) — (threshold condition), and (3). A solution x with load $W = \sum_{i=1}^n w_i x_i$ is feasible iff the following two conditions hold for all $1 \leq i \leq n$: (i) *Maximality condition:* if $W < b_i - w_i$ then $x_i = 1$, and (ii) *Threshold condition:* if $W > b_i$ then $x_i = 0$.

The Maximality and Threshold conditions partition the non-negative numbers for each customer i into 3 disjoint intervals: $[0, b_i - w_i), [b_i - w_i, b_i]$ and $(b_i, \infty]$. Any feasible solution x with load W, depending on the interval W falls into, must satisfy either $x_i = 1$, or $x_i \in \{0,1\}$, or $x_i = 0$. We overlay these intervals for all n customers to partition $[0, \infty]$ into up to $2n+1$ so-called elementary intervals (each additional customer can divide at most 2 existing elementary intervals). Note that elementary intervals can consist of just one number and they can be opened or closed on both sides independently. Each elementary interval I partitions the customers into three sets

$$\mathrm{IN}_I = \{i \,|\, I \subseteq [0, b_i - w_i)\}\ ,\quad \mathrm{FREE}_I = \{i \,|\, I \subseteq [b_i - w_i, b_i]\}\ ,\quad \mathrm{OUT}_I = \{i \,|\, I \subseteq (b_i, \infty]\}\ .$$

If the load of a solution in equilibrium falls into the elementary interval I then no customer in OUT_I has joined the service, all customers in IN_I must have joined the service, and there are no restrictions for customers in FREE_I. Therefore, we partition the solution space for the equilibria problem according to the load of potential solutions. For each elementary interval I we solve the following interval knapsack problem:

$$\max\,(\min) \sum_{i \in \mathrm{FREE}_I} p_i x_i \qquad \text{subject to } \Big(\sum_{i \in \mathrm{FREE}_I} w_i\, x_i + \sum_{i \in \mathrm{IN}_I} w_i\Big) \in I \qquad \text{and} \qquad x_i \in \{0,1\}\ .$$

The optimal solution to the equilibria problem is the maximum (minimum) over all $2n+1$ solutions of the corresponding interval knapsack problems. For elementary intervals that are open we can use the corresponding closed interval, because at an open end the restriction of a variable having value 0 or 1 ends and the variable becomes free. Therefore the solution stays feasible for the equilibria problem. The first and the last of the elementary intervals are trivial intervals, since they correspond to the complete and empty knapsack, which can be tested separately. □

8.2 Adaptive algorithms for the interval knapsack problem

In this section, we present an algorithm for the interval knapsack problem that adapts to the length of the specified interval. The running time of this algorithm increases linearly with the reciprocal of the interval. In particular, we prove Theorem 8.

We begin with the algorithmic ideas. We use a two phase approach. In each phase, we solve a relaxed interval knapsack problem. The solution to the first problem might miss solutions close to the right boundary and the solution to the second problem might miss solutions close to the left boundary. The union of both solutions will cover all feasible solutions. The two relaxed problems are solved following the standard dynamic programming algorithm with one additional nice trick: we expand the dynamic programming table by one dimension used to represent an additional vector of artificial profits corresponding to rounded weights. The original weights allow us to strictly enforce one of the two constraints while the rounded weights (= additional profits) enable us to take care also for the other constraint.

In more detail, we divide the interval $I = [L, R]$ equally into two parts I_l and I_r. Rounding the weights virtually shifts knapsack packing along the weight dimension. In phase 1, we round up, shifting packings from I_l to the right, while in phase 2 we round down. When calculating with sufficient accuracy, packings from I_l will not be shifted beyond R (analogous for packings from I_r). Thus, the required accuracy depends on the ratio $R/(R - L)$. Let $M = (L + R)/2$. Let $\mathrm{Opt}_{A,B}$ denote the value of an optimal solution to the interval knapsack problem with interval $[A, B]$. An approximate solution to interval knapsack can be found by solving the following two relaxed subproblems:

- Compute solution S_1 with $w(S_1) \in [L, R]$ and $p(S_1) \geq (1 - \epsilon) \cdot \mathrm{Opt}_{L,M}$.
- Compute solution S_2 with $w(S_2) \in [L, R]$ and $p(S_2) \geq (1 - \epsilon) \cdot \mathrm{Opt}_{M,R}$.

Clearly, $\max\{p(S_1), p(S_2)\} \geq (1 - \epsilon) \cdot \mathrm{Opt}_{L,R}$. We focus on the first subproblem, the other one can be solved analogously. Let $\bar{w}_i := \lceil w_i/F \rceil \cdot F$ and $\bar{p}_i := \lceil p_i/G \rceil \cdot G$, for all $i \in [n]$, denote rounded weights and profits, respectively, where the values of F and G will be set momentarily. For any $S \subseteq [n]$, define $\bar{w}(S) = \sum_{i \in S} \bar{w}_i$ and $\bar{p}(S) = \sum_{i \in S} \bar{p}_i$. Let $T(i, w, p)$ denote the largest weight among all knapsack packings $S \subseteq [i]$ with $\bar{w}(S) = w$ and $\bar{p}(S) = p$. The following formula gives a recursive definition for all non-trivial values of $T(\cdot)$.

$$T(i + 1, w, p) = \max\{T(i, w, p), T(i, w - \bar{w}_{i+1}, p - \bar{p}_{i+1})\} \ .$$

Rounding weights to multiples of F introduces an absolute error strictly less than nF. Setting $F := (R - L)/(2n)$ ensures $nF \leq (R - L)/2$. Therefore, $w(S) \in [L, M] \Rightarrow \bar{w}(S) \in [L, R]$. The other scaling factor is defined by $G = \epsilon P/(4n)$, where P is chosen sufficiently large. Following a dynamic programming approach, we compute the $T(i, w, p)$ values for all *evaluation points*: $i \in \{1, \ldots, n\}$ in dimension 1, $w \in \{F, 2F, \ldots, \lceil R/F \rceil F\}$ in dimension 2, and $p \in \{G, 2G, \ldots, \lceil P/G \rceil G\}$ in dimension 3, of the three dimensional table. Now, define

$$\alpha(p) = \begin{cases} 1 & \text{if there exists an evaluation point } (n, w, p) : w \in [L, R], T(n, w, p) \in [L, R], \\ 0 & \text{otherwise.} \end{cases}$$

Observe that infeasible subsets (i.e., $w(S) \notin [L, R]$) have no influence on $\alpha(p)$. Sets S with $w(S) > R$ are mapped to an evaluation point with second coordinate $\bar{w}(S) > R$. Sets S with $w(S) < L$ might be mapped to evaluation points with second coordinate in $[L, M]$ but they are explicitly filtered out by checking $T(n, w, p) \in [L, R]$. Let S^* denote a set with optimal profit in $[L, M]$, i.e., S^* yields profit $\mathrm{Opt}_{L,M}$. Then, we conclude that $\max\{p | \alpha(p) = 1\} = \bar{p}(S)$, for some set S with weight in $[L, R]$ and $\bar{p}(S) \geq \bar{p}(S^*)$.

The size of the dynamic programming table is $n\lceil R/F \rceil \lceil 2P/G \rceil = O\left(\frac{n^3 R}{(R-L)\epsilon}\right)$. P needs to be chosen sufficiently large so that all solutions are covered. W.l.o.g., assume $\epsilon \leq \frac{1}{2}$. Then choosing $P \geq 2 \cdot \mathrm{Opt}_{L,M}$ is sufficient. Furthermore, if the computed solution S satisfies $p(S) \geq P/4$ then $\bar{p}(S) \geq (1 - \epsilon) \cdot \mathrm{Opt}_{L,M}$, which follows analogously to the calculation for the approximation factor of the standard knapsack problem. The problem that remains to be solved is how to determine the right choice for P. We do this in form of a geometric search. We start by setting $P := 2 \sum_i p_i$. If we do not find any feasible solution then we immediately stop and return $\mathrm{Opt}_{L,M} = 0$. Otherwise, we find a solution S whose profit $\bar{p}(S)$ is maximal among all computed solutions. If $\bar{p}(S) \geq \frac{1}{2}P$ then

$$p(S) \ \geq \ (1 - \epsilon) \cdot \bar{p}(S) \ \geq \ \frac{1}{2} \cdot \bar{p}(S) \ \geq \ \frac{1}{4} \cdot P \ ,$$

so that S is the $(1 - \epsilon)$-approximation that we are looking for. If $\bar{p}(S) < \frac{1}{2}P$ then we increase the accuracy by decreasing the value of P by a factor of 2 until we find a solution S with $\bar{w}(S) \in [\frac{1}{2}P, P]$. The maximum number of iterations that we need to execute is $O\left(\log\left(\frac{\sum_i p_i}{P_{\min}}\right)\right)$. Thus the overall running time is $O\left(\frac{n^3 R}{(R-L)\epsilon}\log\left(\frac{\sum_i p_i}{P_{\min}}\right)\right)$. This proves Theorem 8.

For the average case analysis in the next section, we need this result in a slightly different form. Let $W_{\max}$ and $W_{\min}$ denote the maximum and minimum weight, respectively. If profits are defined by a non-decreasing, non-negative, concave function then $\frac{P_{\max}}{P_{\min}} \leq \frac{W_{\max}}{W_{\min}}$. This yields the following corollary.

Corollary 12 *Suppose profits in the interval knapsack problem are defined by a non-negative, non-decreasing, concave function. Then the algorithm above solves the interval knapsack problem in time* $O\left(\frac{n^3 R}{(R-L)\epsilon}\log\left(\frac{nW_{\max}}{W_{\min}}\right)\right)$.

8.3 Estimating equilibria under imperfect information

We now prove Theorem 9. Let $\epsilon \in [0,1]$ be fixed. We first use a reduction to the interval knapsack problems described in Section 8.1. Each interval knapsack problem will be solved using the algorithm described in Section 8.2. Let $S = \{0\} \cup \{s_1, \ldots, s_{2n}\}$ denote the set of boundary points of the intervals, with $s_1 \leq \ldots \leq s_{2n}$. For analytical purposes, we consider the intervals for all pairs $\{L, R\} \subseteq S$, $I = \{(L,R) : \{L,R\} \subseteq S, L < R\}$.

Consider any L, R with $(L,R) \in I$. Note that each L, R is of the form b_i or $b_i - w_i$ where w_i and b_i are non-negative random variables with density functions f_i and g_i respectively, with μ being an upper bound on all the expectations and ϕ – an upper bound on the values of all functions f_i, g_i. We need to modify the algorithm as follows. If for some i, the sampled value $L = b_i - w_i$ is negative, then our algorithm takes into account only all intervals of the form $(0,R)$ for all $R \in S \cap \mathbb{R}_+$. We assume thus that set I does not contain any interval (L,R) with $L < 0$, but $(0,R)$ instead. Moreover, if we are given an interval $(b_i - w_i, b_i) \in I$, then the algorithm solves only the interval knapsack problem in intervals (L, b_i) and $(b_i - w_i, R)$ where $L = \max\{L' \in S : L' < b_i - w_i\}$ and $R = \min\{R' \in S : R' > b_i\}$. That is, the algorithm does not consider interval $(b_i - w_i, b_i)$ itself. Thus, our algorithm considers only intervals $(L,R) \in I$ of the form $L = b_i - x_i$, $R = b_j - x_j$, where $i \neq j$ and $x_i \in \{0, w_i\}$, $x_j \in \{0, w_j\}$, and does not consider intervals of the form $(b_i - w_i, b_i)$.

Let us fix an interval $(L,R) \in I$. Let $T(L,R)$ be a random variable denoting the running time of our algorithm for the interval knapsack problem on interval (L,R). Let $\ell = \mu 2^{-n}$. If $w_i < \ell$ for some $i \in \{1, \ldots, n\}$, then we use a brute force exact algorithm with running time $O(2^n)$. Note, that this part has a $O(\phi\mu)$ expected running time, since $\mathbf{Pr}[w_i \leq \ell] = \int_0^\ell g_i(x)dx \leq \phi\mu/2^n$. In the same way the algorithm also checks if $R/(R-L) \leq 2^n$.

If $R/(R-L) \leq 2^n$, and $w_i \geq \ell$ for $i = 1, \ldots, n$, then we call the adaptive algorithm from Section 8.2. The running time of this algorithm is $O\left(\frac{n^3 R}{(R-L)\epsilon}\log\left(\frac{nW_{\max}}{W_{\min}}\right)\right)$ by Corollary 12, where $W_{\max} = \max_i\{w_i\}$ and $W_{\min} = \min_i\{w_i\}$.

Our algorithm only considers intervals $(L,R) \in I$ of the form $L = b_i - x_i$, $R = b_j - x_j$, where $i \neq j$ and $x_i \in \{0, w_i\}$, $x_j \in \{0, w_j\}$. Thus, for the analysis of the expected running time of the part $O\left(\frac{n^3 R}{(R-L)\epsilon}\right)$ we can assume that possible values of $x_i = w_i$ that may appear in L, R are worst case fixed numbers. This can only decrease the expectations of L, R and so μ is still an upper bound on these expectations, and this does not affect the ϕ upper bound. This means that we can treat the two parts $O\left(\frac{n^3 R}{(R-L)\epsilon}\right)$ and $O\left(\log\left(\frac{nW_{\max}}{W_{\min}}\right)\right)$ in the running time independently. The overall expected running time over all the interval knapsack problems is by linearity of expectation bounded by:

$$\mathbf{E}\left[\sum_{(L,R)\in I} T(L,R)\right] \leq O\left(n^2\right) \cdot \max_{L,R} \mathbf{E}\left[T(R,L)\right] = O\left(\frac{n^5}{\epsilon}\right) \cdot \max_{L,R} \mathbf{E}\left[\frac{R}{R-L}\right] \cdot \mathbf{E}\left[\log\left(\frac{nW_{\max}}{W_{\min}}\right)\right] \quad .$$

The next two lemmas (for the proofs, see Appendix D and E) prove upper bounds on $\mathbf{E}\left[\frac{R}{R-L}\right]$ and on $\mathbf{E}\left[\log\left(\frac{nW_{\max}}{W_{\min}}\right)\right]$. After applying these lemmas we can obtain the following upper bound on the expected running time:

$$O\left(\frac{n^5}{\epsilon} \cdot ((n+1)\phi\mu + 1) \cdot (\log(\phi\mu n + 1) + 3\log(n))\right) \quad .$$

Lemma 13 *Let b_i, w_i be given continuous random variables with density functions $f_i(x)$, $g_i(y)$ respectively, for $i = 1, 2$. Let $\phi = \sup\{f_i(x), g_i(x) : x \in \mathbb{R}, i = 1, 2\}$, $\mu = \max\{\mathbf{E}[b_1], \mathbf{E}[b_2]\}$, $n \geq 2$. Define new random variables $X = \max\{b_1 - w_1, 0\}$ and $Y = \max\{b_2 - w_2, 0\}$ and $Z = \min\left(\frac{\max(X,Y)}{|X-Y|}, 2^n\right)$. Then, the expectation of random variable Z can be bounded as: $\mathbf{E}[Z] \leq 2(n+1)\phi\mu + 2$.*

Lemma 14 . *Let $X_1, X_2, \ldots, X_n$ be n given non-negative, continuous random variables with density functions $f_1(x)$, $f_2(x), \ldots, f_n(x)$ respectively. Let $\phi = \sup\{f_i(x) : x \geq 0; i = 1, \ldots, n\}$, $\mu = \max_i(\mathbf{E}[X_i])$, $n \geq 2$, and $\ell = \mu 2^{-n}$. Define $h(x_1, \ldots, x_n) = \log \frac{\max(x_1,\ldots,x_n)}{\min(x_1,\ldots,x_n)}$ if $x_1, \ldots, x_n \geq \ell$ and $h(x_1, \ldots, x_n) = 0$ otherwise. Then the expectation of random variable $Z = h(X_1, \ldots, X_n)$ can be bounded as: $\mathbf{E}[Z] \leq \log(\phi\mu n + 1) + 2\log(n)$.*

References

[1] R. Beier and B. Vöcking. Random knapsack in expected polynomial time. *35th STOC*, 2003.

[2] R. Beier and B. Vöcking. Probabilistic Analysis of Knapsack Core Algorithms. Submitted to *15th SODA'04*, 2003.

[3] V. Conitzer and T. Sandholm. Complexity results about Nash equilibria. *TR CMU-CS-02-135*. Carnegie-Mellon U., 2002.

[4] X. Deng, C. H. Papadimitriou, and S. Safra. On the complexity of equilibria. *34th STOC*, 67–71, 2002.

[5] N. R. Devanur, C. H. Papadimitriou, A. Saberi, and V. V. Vazirani. Market equilibrium via a primal-dual-type algorithm. *43rd FOCS*, 2002.

[6] P. A. Diamond. Welfare analysis of imperfect information equilibria. *The Bell Journal of Economics*, 9(1): 82–105, 1978.

[7] G. Facchini, F. Van Megen, P. Borm, and S. Tijs. Congestion models and weighted Bayesian potential games. *Theory and Decision*, 42: 193–206, 1997.

[8] D. Fotakis, S. Kontogiannis, E. Koutsoupias, M. Mavronicolas, and P. Spirakis. The structure and complexity of Nash equilibria for a selfish routing game. *29th ICALP*, 123–134, 2002.

[9] R. Feldmann, M. Gairing, T. Lücking, B. Monien, and M. Rode. Nashification and the coordination ratio for a selfish routing game. *30th ICALP*, 2003.

[10] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. Freeman, 1979.

[11] A. Goel and P. Indyk. Stochastic load balancing and related problems. *40th FOCS*, 1999.

[12] A. Goldberg and A. Marchetti-Spaccamela. On finding the exact solution to a zero-one knapsack problem. *16th STOC*, 359–368, 1984.

[13] M. Harris and A. Raviv. A theory of monopoly pricing schemes with demand uncertainty. *The American Economic Review*, 71(3): 347–365, 1981.

[14] J. C. Harsanyi. Games with incomplete information played by Bayesian players, I, II, III. *Management Science*, 14: 159–182, 320–332, 468–502, 1968.

[15] J. C. Harsanyi. Games with randomly distributed payoffs. *International Journal of Game Theory*, 2: 1–23, 1973.

[16] R. Holzman, N. Law-Yone. Strong equilibrium in congestion games. *Games and Economic Behavior*, 21: 85–101, 1997.

[17] O. H. Ibarra and C. E. Kim. Fast approximation algorithms for the knapsack and sum of subset problems. *Journal of the ACM*, 22: 463–468, 1975.

[18] R. Impagliazzo and M. Naor. Efficient cryptographic schemes provably as secure as subset sum. *J. Cryptology*, 9(4): 199–216, 1996.

[19] J. Kleinberg, Y. Rabani, and É. Tardos. Allocating bandwidth for bursty connections. *29th STOC*, 664–673, 1997.

[20] A. S. Kyle. Informed speculation with imperfect competition. *The Review of Economic Studies*, 56(3): 317–355, 1989.

[21] H E. Leland and A. Meyer. Monopoly pricing structures with imperfect discrimination. *The Bell J. Economics*, 7(2): 449–462, 1976.

[22] D. Manlove. Minimaximal and maximinimal optimization problems: a partial order-based approach. PhD thesis, University of Glasgow, Dept. of Computing Science, June 1998. http://www.dcs.gla.ac.uk/ davidm/publications.html.

[23] E. Maskin and J. Riley. Monopoly with incomplete information. *The RAND Journal of Economics*, 15(2): 171–196, 1984.

[24] I. Milchtaich. Congestion games with player-specific payoff function. *Games and Economic Behavior*, 13: 111–124, 1996.

[25] I. Milchtaich. Crowding games are sequentially solvable. *International Journal of Game Theory*, 27: 501–509, 1998.

[26] D. Monderer and L. S. Shapley. Potential games. *Games and Economic Behavior*, 14: 124–143, 1996.

[27] W. Y. Oi. A Disneyland dilemma: Two part tariffs for a Mickey Mouse monopoly. *The Quarterly Journal of Economics*, 85(1): 77–96, 1971.

[28] G. Owen. *Game Theory*. Academic Press, 1995.

[29] A. Ronen. On approximating optimal auctions. *3rd ACM Conference on Electronic Commerce (EC '01)*, 2001.

[30] A. Ronen and A. Saberi. On the hardness optimal auctions. *43st FOCS*, 396–405, 2002.

[31] R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *Int. Journal of Game Theory*, 2: 65–67, 1973.

[32] R. Schmalensee. Monopolistic two-part pricing arrangements. *The Bell J. of Economics*, 12(2): 445–466, 1981.

[33] S. Scotchmer. Two-tier pricing of shared facilities in a free-entry equilibrium. *The RAND J. Economics*, 16: 456–472, 1985.

[34] R. W. Staiger and F. A. Wolak. Collusive pricing with capacity constraints in the presence of demand uncertainty. *The RAND Journal of Economics*, 23(2): 203–220, 1992.

[35] T. Ui. A Shapley value representation of potential games. *Games and Economic Behavior*, 31: 121–135, 2000.

Appendix

A Proof of Theorem 1 (FPAS for Individual Pricing)

Suppose that the customers demands and utility functions are specified by $w_1, \ldots, w_n$ and $u_1(\cdot), \ldots, u_n(\cdot)$, respectively. Observe, that for any subset $S \subseteq [n]$ of costumers we can choose a pricing scheme such that S is the only equilibrium: Let $load = \sum_{i \in S} w_i$. For all costumers $i \in S$ choose p_i from the interval $[0, u_i(load))$. For all costumers $i \notin S$ choose $p_i = \infty$ (alternatively, $p_i = u_i(\min_{i \in [n]} w_i)$). The maximum profit that can be obtained from S is $\sum_{i \in S} u_i(load)$. If the load of the system is pre-specified then $p_i = u_i(load)$ is the maximal price costumer i is willing to pay for the service under this load. Thus we can find an optimal individual pricing scheme by solving the following knapsack problem:

$$\begin{aligned} \text{maximize} \quad & \textstyle\sum_{i=1}^{n} x_i \cdot u_i(load) \\ \text{subject to} \quad & \textstyle\sum_{i=1}^{n} x_i \cdot w_i \le load \\ & x_i \in \{0,1\} \qquad\qquad \text{for every } 1 \le i \le n \ . \end{aligned}$$

Observe, that we can use inequality in the constraint, because costumers will join the service even when the actual load is smaller. Define $T = \max_{1 \le i \le n} u_i(w_i)$. Then the value of the optimal total profit is in the interval $[T, n\,T]$. Rounding individual profits to multiples of $F = \frac{T\,\varepsilon}{n}$ ensures an $(1-\varepsilon)$-approximation of the optimal total profit. So it suffices to find an optimal assignment of rounded profits. Since the maximum individual profit is T we have to consider at most n/ϵ values for each costumer. When fixing profit p_i, costumer i can accept a load value of at most $u^{-1}(p_i)$. For any s, $0 \le s \le \frac{2\,n^2}{\varepsilon}$, let $I_{i,s} = \max\{\ell : u_i(\ell) \ge Fs\}$. Observe that for every $\ell \in (I_{i,s+1}, I_{i,s}]$ (this interval may be empty) we have $Fs \le u_i(\ell) < F(s+1)$. This means that if the congestion is ℓ for some $\ell \in (I_{i,s+1}, I_{i,s}]$, then customer i can accept only the price smaller than or equal to Fs. Furthermore, since the service provider is seeking a maximal price, in the optimal solution if customer i is going to use the service, then the price must be set to exactly Fs. This motivates us for each customer i to consider $n/\varepsilon)$ intervals of distinct congestions of the service: each interval of the form $(I_{i,s+1}, I_{i,s}]$. Since, if the congestion is in that interval then the service provider will assign the price either $p_i = Fs$ or $p_i = \infty$, we can consider this case as the case in which "the price is fixed."

Now, we have to deal with all customers at the same time. Let us consider a *minimal* partition of $\mathbb{R}$ into a set of intervals $\mathbb{I}$ such that if x and y are in the same interval in $\mathbb{I}$ then for every i, $1 \le i \le n$, if $x \in (I_{i,s+1}, I_{i,s}]$ for some s, then $y \in (I_{i,s+1}, I_{i,s}]$. That is, $\mathbb{I}$ is a refinement of all intervals for all customers. Notice that there are $\mathcal{O}(n^2/\varepsilon)$ intervals in $\mathbb{I}$ and that each interval in $\mathbb{I}$ is of the form $(X, Y]$. Furthermore, let us observe that for every i, $1 \le i \le n$, for every $(X, Y] \in \mathbb{I}$, and for every $\ell \in (X, Y]$ with $(X, Y] \subseteq (I_{i,s+1}, I_{i,s}]$, if the congestion in the system is ℓ then the service provider will set either $p_i = Fs$ or $p_i = \infty$.

Let us take any interval $(X, Y]$ in $\mathbb{I}$. We show how to find an optimal rounded price scheme if the congestion is in the interval $(X, Y]$. For every i, $1 \le i \le n$, let $P_i\langle Y \rangle = Fs$, where s is such that $(X, Y] \subseteq (I_{i,s+1}, I_{i,s}]$.

An optimal rounded price vector $\mathbf{p} = (p_1, \ldots, p_n)$ under the assumptions the congestion is in the interval $(X, Y] \in \mathbb{I}$ is identical to the solution of the following optimization problem:

$$\begin{aligned} \text{maximize} \quad & \textstyle\sum_{i=1}^{n} x_i \cdot P_i\langle Y \rangle \\ \text{subject to} \quad & \textstyle\sum_{i=1}^{n} x_i \cdot w_i \le Y \\ & x_i \in \{0,1\} \qquad\qquad \text{for every } 1 \le i \le n \ , \end{aligned}$$

where for every i, $1 \le i \le n$, the value of p_i is either $P_i\langle Y \rangle$ if $x_i = 1$ and is $+\infty$ otherwise.

This knapsack problem can now be solved in time $n^2\varepsilon$ since there are only $n\varepsilon$ different values for the profits. Moreover, since the number of intervals we have to consider is $\mathcal{O}(n^2/\varepsilon)$, we obtain an FPAS with running time $\mathcal{O}(n^6\,\varepsilon^{-2})$. ☐

B Proof of Theorem 3 and 5 (Inapproximability)

B.1 Proof of Theorem 3

Assume uniform pricing with $p_i = aw_i$. We show, that for every choice of $a \in \mathbb{R}_+$ there exist demands w_i and utility functions u_i such that the value of the maximum (minimum) profit customer equilibria cannot be approximated. Instead of utility functions u_i we directly specify thresholds b_i. The u_i's can then be chosen accordingly to satisfy $b_i = u_i^{-1}(aw_i)$.

We use a reduction from an NP-hard problem called subset-sum [10]. Given a set $\{c_1, \ldots, c_n\}$, $0 < c_i \in \mathbb{N}$ and some number $B \in \mathbb{N}$, decide whether $\exists I \subseteq \{1, \ldots, n\}$ with $\sum_{i \in I} c_i = B$. W.l.o.g., we assume that all c_i's are even (otherwise we multiply all c_i's and B by 2.)

Maximization: Consider the following max profit customer equilibria problem with profit equal to demand. There are n regular customers $1, \ldots, n$ and two special customers: a "large" customer L and a "giant" customer G. The demands and thresholds are given in Table 1. The parameters l and g can be chosen to satisfy $B < l \leq C \ll g$ with $C = \sum_{1 \leq i \leq n} c_i$. We use customer G as an indicator for deciding the subset-sum problem. If there exists a solution I to the subset-sum problem, then x^* (as given in Table 1) is a feasible solution to the best equilibria problem with load $g + B$. If there is no solution I, then no feasible equilibrium x can contain customer G. First notice, that there is no feasible solution containing both customers L and G, because their cumulative demand is strictly larger than the threshold for G. Now assume some solution x contains G but not L. Since the demand of the regular customers cannot sum up to exactly B and moreover all w_i's are even, x has load at most $g + B - 2$, which violates the maximality condition for customer L. Hence any feasible equilibrium excludes G and has load at most $C + l$. Since g and therefore also $(g + B)/(C + l)$ can be chosen arbitrarily large, any approximation algorithm for the best equilibria problem can be used to decide the subset-sum problem.

customer	demands	thresholds	x^*
$1, \ldots, n$	$w_i = c_i$	$b_i = g + B$	$x_i^* = 1$ iff $i \in I$
L	$w_L = l$	$b_L = g + l + B - 1$	$x_L^* = 0$
G	$w_G = g$	$b_G = g + B$	$x_G^* = 1$

Table 1: Best equilibria instance and solution x^*.

Minimization: We construct a worst equilibria instance with n regular customers $(1, \ldots, n)$ having demands $w_i = a_i$ and thresholds $b_i = B$. Moreover there is only one special customer with demand g and threshold $B + g - 1$. If there exists a solution I to the subset-sum problem, then there is a solution x to the worst equilibria problem with load B. If there is no such solution I, then any solution x containing only regular customers has load at most $B - 2$, violating the maximality constraint for customer G. Thus any feasible solution must contain the special customer and has load at least g. Since g and therefore also g/B can be chosen arbitrarily large, any approximation algorithm for the worst equilibria problem can be used to decide the subset-sum problem. This completes the proof of Theorem 3.

B.2 Proof of Theorem 5

We give a reduction from the following NP-hard partition problem [10]: Given n numbers $a_1, \ldots, a_n$, decide whether there exists a subset $A \subset [n]$ with $\sum_{i \in A} a_i = \sum_{i \in [n] \setminus A} a_i = B$. W.l.o.g., assume that all a_i are even. Consider the following instance of the best customer equilibria problem. There are n small customers with demands $w_i = a_i$ and one giant customer G with demand $w_G = g$ (e.g. $g = 2B$ suffices). For all small costumers ($i \in [n]$) define utility function $u_i(x) = 2$, for $x \in [0, B]$ and $u_i(x) = 0$, for $x > B$. Moreover, $u_G(x) = \sqrt{n}$, for $x \in [0, B + g - 1]$ and $u_G(x) = 0$ when $x > B + g - 1$.

First observe that, under profit maximization, a reasonable provider chooses either flat rate $p = 2$ or $p = \sqrt{n}$.

For $p = 2$ the utility functions translate into thresholds $b_i = B$, for $i \in [n]$, and $b_G = B + g - 1$. Given a positive instance of the partition problem, the larger of the two subsets A or $[n] \setminus A$ defines an equilibrium with profit at least n. Given a negative instance of the partition problem, any solution without the giant costumer G has demand at most $B - 2$ because the a_i's are even, $b_i = B$ and the a_i's cannot sum to exactly B, which violates the maximality condition for G. Therefore, the only equilibrium is $\{G\}$ with profit 1.

For $p = \sqrt{n}$ we obtain thresholds $b_i = 0$ for $i \in [n]$ and $b_G = B + g - 1$. Thus only the giant customer can join the service and, hence, there is only one equilibrium, namely $\{G\}$ with profit $\sqrt{n}$.

Summarizing, we have seen that the maximum profit is either 2, $\sqrt{n}$ or $\geq n$. Therefore any approximation algorithm for computing a price p with approximation factor less than $\sqrt{n/2}$, implicitly decides the partition problem.

C Proofs of Theorems 6 and 7 (Robust Equilibria and Min-Max Knapsack)

C.1 Inapproximability of δ-robust equilibria: Theorem 6

We essentially reuse the proof for the inapproximability of the worst and best equilibria problem (Theorem 3).

Proof. *Minimization:* We use the reduction from the proof of Theorem 3 but with different threshold values for the constructed equilibria instance. Setting $b_G = (B - 1 + g)/(1 - \delta)$ for the giant item G and $b_i = B/(1 + \delta)$ for all regular items ($1 \leq i \leq n$), equilibria have to fulfill the following conditions:

$$x_G = 0 \quad \Rightarrow \quad \textstyle\sum_j w_j x_j \geq B - 1$$
$$x_i = 1 \quad \Rightarrow \quad \textstyle\sum_j w_j x_j \leq B$$

The other conditions are spared out since they are not critical. If demands can add up to exactly B then there is an equilibrium of load B. Otherwise any equilibrium must contain G since we assumed demands to be even numbers.

Maximization: Set $b_l = (g + B + l - 1)/(1 - \delta)$ (large item) and $b_G = b_i = (B + g)/(1 + \delta)$ for the giant item and all regular items i. Then equilibria have to fulfill the following conditions:

$$\begin{array}{rcl} x_L = 0 & \Rightarrow & \sum_j w_j x_j \geq B + g - 1 \\ x_G = 1 & \Rightarrow & \sum_j w_j x_j \leq B + g \\ x_i = 1 & \Rightarrow & \sum_j w_j x_j \leq B + g. \end{array}$$

Again, other conditions are spared out since they are not critical. If demands can add up to exactly B then there is an equilibrium of load $B + g$. Otherwise any equilibrium must contain item L and therefore cannot contain item G. □

C.2 Min-Max Knapsack and Proof of Theorem 7

Theorem 15 *Consider the min-max knapsack problem.*

a) If weights and profits are arbitrary then the problem is inapproximable.

c) If weights and profits are equal then the problem admits an FPAS.

b) The δ-robust min-max knapsack problem ($\delta \geq 0$) defined by

$$\min\{\sum_j p_j x_j \mid \sum_j w_j x_j \geq B(1+\delta) \wedge (x_i = 0) \Rightarrow w_i + \sum_j w_j x_j \geq B(1-\delta)\}$$

$$\begin{array}{rcl} & & \sum_j w_j x_j \geq B(1+\delta) \\ x_i = 0 & \Rightarrow & w_i + \sum_j w_j x_j \geq B(1-\delta) \end{array}$$

admits an FPAS with running time poly$(n, \frac{1}{\epsilon}, \frac{1}{\delta})$.

Proof. We prove part a, c, b in this order.
Part a: We give a reduction from subset-sum. Given set $\{a_1, \ldots, a_n\}$ $(0 < a_i \in \mathbb{N})$ and some number $B \in \mathbb{N}$, decide whether $\exists I \subseteq \{1, \ldots, n\}$ with $\sum_{i \in I} a_i = S$. Consider the following min-max knapsack problem. There are n regular items with weight $w_i = a_i$ and profit $p_i = 1$ for $1 \leq i \leq n$ and one special item with $w_0 = 1$ and $p_0 = g (\in \mathbb{N})$ arbitrary. The weight bound for the knapsack is $B = S$. If there exists a solution I to the subset-sum problem, then there exists a maximal knapsack with profit $|I| \leq n$ and weight B. Now assume there is no solution I. Any knapsack packing that uses only regular items and satisfies the threshold B has weight at most $B - 1$. Since $w_0 = 1$ these knapsacks are not maximal. Hence any maximal knapsack must contain the special item and, therefore, has profit at least g. Since g and therefore also g/n can be chosen arbitrarily large, any approximation algorithm for the min-max knapsack problem can be used to solve the subset-sum problem.
Part c: W.l.o.g. assume $w_1 \leq w_2 \leq \cdots \leq w_n \leq b$ and $\sum w_i > b$. First consider the case $\delta = 0$ which is equivalent to the min-max knapsack problem. We give a reduction from the min-max knapsack problem to n interval knapsack problems. For $1 \leq k \leq n$, the kth of these minimum interval knapsack problems is defined as follows:

Input: weights $w_1, \ldots, w_n$, profits $p_1, \ldots, p_n$, interval $(b - w_k, b]$.
Output: a subset $I \subset \{1, \ldots, n\}$ with $\{1, \ldots, k-1\} \subseteq I$, $\sum_{i \in I} w_i \in (b - w_k, b]$ and $\sum_{i \in I} p_i$ is minimized.

The set I corresponds to the knapsack. Observe that any feasible solutions to one of these n minimum interval knapsack problems is also feasible for the min-max knapsack problem. Clearly they satisfy the threshold constraint, but also the maximality constraint, i.e., $\sum_{i \in I} w_i > b - w_j$, for every $j \notin I$, because of the assumption $\{1, \ldots, k-1\} \subseteq I$ and $w_j \geq w_k$ for $j \in \{k, \ldots, n\}$. Observe, that for $\delta > 0$ we get n minimum interval knapsack problems with intervals $(b(1-\delta) - w_k, b(1+\delta)]$. Therefore, the interval length is at least $2\delta b$. We can use the approximation algorithm for the interval knapsack problem from Section 8.2. It can easily be adapted to the minimization problem with one open interval boundary. The time to approximate one minimum interval knapsack problem with parameter ϵ is $\frac{n^3}{2\delta\epsilon} \log \frac{\sum_i p_i}{P_{\min}}$.

Part b: Now assume $w_i = p_i$ for all $0 \leq i \leq n$. We show, that the min-max knapsack problem can be reduced to n minimal knapsack problems defined as follows:

Input: weights $w_1, \ldots, w_n$, profits $p_1, \ldots, p_n$, boundary $b - w_k]$.
Output: a subset $I \subset \{1, \ldots, n\}$ with $\{1, \ldots, k-1\} \subseteq I$, $\sum_{i \in I} w_i > b - w_k$ and $\sum_{i \in I} p_i$ is minimized.

Observe that these are the same problems as in part b, except that we now have no upper bound b on the weight of the subset. Any solution to one of these problems fulfills the maximality constraint of the min-max knapsack problem, as argued above. However, the threshold constraint of the min-max knapsack problem (i.e., $\sum_{i \in I} w_i \leq b$) might be violated.

In order to show that min-max knapsack can be reduced to the n minimal knapsack problems, fix any optimal solution for the min-max knapsack problem. Let I^* denote the corresponding knapsack. Let j denote the smallest index such that $j \notin I^*$. We observe that I^* is an optimal solution to the jth minimum knapsack problem. In particular, none of the n minimum knapsack problems has a solution of smaller weight than the weight of I^* because, otherwise, such a solution would contradict the optimality of I^*. Consequently, the min-max knapsack problem can be solved by solving the n minimum knapsack problems and returning a knapsack of smallest weight among them.

A straightforward adaptation of the well-known FPAS from [17] for the maximum knapsack problem yields an FPAS for the minimum knapsack problem as well. We use the above reduction from the min-max knapsack problem to n disjoint minimum knapsack problems and approximate their optimal solution within a factor of $1+\epsilon$. Let I' denote the knapsack of smallest weight among the n approximate solutions and let k' denote the index of the corresponding minimum knapsack problem. We split into two cases.

- **Case 1:** Suppose $\sum_{i \in I'} w_i \leq b$. In this case, I' fulfills the threshold constraint. Furthermore, I' fulfills the maximality constraint as it is a feasible solution for one of the n minimum knapsack problems. Consequently, I' is a feasible $(1+\epsilon)$-approximate solution for the min-max knapsack problem.
- **Case 2:** Now assume $\sum_{i \in I'} w_i > b$. From this assumption, we can conclude that the weight of the min-max knapsack I^* is larger than $b/(1+\epsilon)$ because
$$b \; < \; \sum_{i \in I'} w_i \; \leq \; (1+\epsilon) \sum_{i \in I^*} w_i \ .$$
In this case, we calculate any maximal knapsack, i.e., any knapsack that satisfies the threshold and maximality constraint. (For example, we could use a simple greedy algorithm to compute such a maximal knapsack.) Clearly, the weight of this maximal knapsack is lower-bounded by the weight of I^*, as I^* is the minimum profit maximal knapsack and profits are equal to weights. Furthermore, the weight of this knapsack is upper-bounded by the threshold b. Consequently, this knapsack is a $(1+\epsilon)$-approximate solution for the min-max knapsack problem.

This completes the proof of Theorem 15, and this also proves Theorem 7. □

D Proof of Lemma 13

Proof. The expectation can be upper-bounded as follows:

$$\mathbf{E}[Z] = \mathbf{E}\left[\min\left(\frac{\max(X,Y)}{|X-Y|}, 2^n\right)\right] \leq \mathbf{E}\left[\max\left(\min\left(\frac{X}{X-Y}, 2^n\right), 0\right) + \max\left(\min\left(\frac{Y}{Y-X}, 2^n\right), 0\right)\right].$$

So it only suffices to bound $\mathbf{E}\left[\max\left(\min\left(\frac{X}{X-Y}, 2^n\right), 0\right)\right]$.

Let $h(x,y) = \max\left(\min\left(\frac{x}{x-y}, 2^n\right), 0\right)$. Observe that $h(x,y) = h(x/y, 1)$ and $h(x,0) = 0$, and if $x = 0$, $y > 0$, then $h(x,y) = 1$. Also if $z > 1$, then $h(z,1) = 0$.

Observe, that the random variables X, Y have discontinuity at value 0, and $\mathbf{E}[X] \leq \mu$ and $\mathbf{E}[Y] \leq \mu$. Also, observe that in the range of values $(0, \infty]$ we can use f_1 and f_2 as density functions for X and Y.

Let $y > 0$ be fixed. Suppose first that $\mathbf{Pr}[X = 0] = 0$. Then we first compute:

$$\mathbf{E}[h(X,y)] = \int_0^\infty h(x,y) f_1(x) dx = \int_0^\infty h(x/y,1) f_1(x) dx.$$

By changing the variable to $z = x/y$, since $dz/dx = 1/y$, we obtain:

$$\mathbf{E}[h(X,y)] = y \int_0^\infty h(z,1) f_1(zy) dz \leq \phi y \int_0^\infty h(z,1) dz = \phi y \int_0^\infty \max\left(\min\left(\frac{1}{1-z}, 2^n\right), 0\right) dz =$$

$$\phi y \int_0^1 \max\left(\min\left(\frac{1}{1-z}, 2^n\right), 0\right) dz = \phi y \int_0^{\frac{2^n-1}{2^n}} \max\left(\min\left(\frac{1}{1-z}, 2^n\right), 0\right) dz +$$

$$+\phi y\int_{\frac{2^n-1}{2^n}}^{1}\max\left(\min\left(\frac{1}{1-z},2^n\right),0\right)dz=\phi y\int_0^{\frac{2^n-1}{2^n}}\max\left(\frac{1}{1-z},0\right)dz+\phi y\int_{\frac{2^n-1}{2^n}}^{1}2^n dz=$$

$$\phi y\int_0^{\frac{2^n-1}{2^n}}\frac{1}{1-z}dz+\phi y 2^n\frac{1}{2^n}=\phi y[-\ln(1-z)]_{z=0}^{z=\frac{2^n-1}{2^n}}+\phi y=(\ln(2^n)+1)\phi y\le(n+1)\phi y.$$

Suppose now that $\mathbf{Pr}\left[X=0\right]=p$. Then

$$\mathbf{E}\left[h(X,y)\right]=p\cdot\mathbf{E}\left[h(0,y)\right]+(1-p)\cdot\mathbf{E}\left[h(X,y)|X>0\right]=p+(1-p)\int_{0+}^{\infty}h(x,y)\bar{f}(x)dx,$$

where $\bar{f}$ is a density function of random variable $X|X>0$. Let us observe, that for any $x>0$ we have $\bar{f}(x)\le\frac{f_1(x)}{1-p}$. We can therefore write that:

$$\mathbf{E}\left[h(X,y)\right]\le p+\int_{0+}^{\infty}h(x,y)dx\le 1+(n+1)\phi y.$$

Finally, we observe that $h(x,0)=0$, so we have that:

$$\mathbf{E}\left[Z\right]=\mathbf{E}\left[h(X,Y)\right]=\int_{0+}^{\infty}\mathbf{E}\left[h(X,y)\right]f_2(y)dy\le\int_{0+}^{\infty}(1+(n+1)\phi y)f_2(y)dy=$$

$$\int_{0+}^{\infty}f_2(y)dy+(n+1)\phi\int_{0+}^{\infty}yf_2(y)dy\le 1+(n+1)\phi\mathbf{E}\left[Y\right]\le 1+(n+1)\phi\mu.$$

□

E Proof of Lemma 14

Proof. Using the linearity of expectation we can write:

$$\mathbf{E}\left[Z\right]=\mathbf{E}\left[\log\frac{\max(X_1,\ldots,X_n)/\mu}{\min(X_1,\ldots,X_n)/\mu}\right]=\mathbf{E}\left[\log\frac{\max(X_1,\ldots,X_n)}{\mu}\right]+\mathbf{E}\left[\log\frac{\mu}{\min(X_1,\ldots,X_n)}\right].$$

Since $\log$ is a concave function, we can apply the well-known Jensen's inequality to the first term:

$$\mathbf{E}\left[\log\frac{\max(X_1,\ldots,X_n)}{\mu}\right]\le\log\mathbf{E}\left[\frac{\max(X_1,\ldots,X_n)}{\mu}\right].$$

By using the linearity of expectation we obtain the following:

$$\log\mathbf{E}\left[\frac{\max(X_1,\ldots,X_n)}{\mu}\right]\le\log\mathbf{E}\left[\frac{1}{\mu}\sum_i X_i\right]=\log\left(\frac{1}{\mu}\sum_i\mathbf{E}\left[X_i\right]\right)\le\log\left(\frac{1}{\mu}n\mu\right)=\log(n).$$

Now we apply Jensen's inequality to the second term:

$$\mathbf{E}\left[\log\frac{\mu}{\min_i(X_i)}\right]\le\log\mathbf{E}\left[\frac{\mu}{\min_i(X_i)}\right]=\log\mathbf{E}\left[\max_i\frac{\mu}{X_i}\right]\le\log\mathbf{E}\left[\sum_i\frac{\mu}{X_i}\right]=\log\left(\sum_i\mathbf{E}\left[\frac{\mu}{X_i}\right]\right),$$

and so it suffices to bound:

$$\mathbf{E}\left[\frac{\mu}{X_i}\right]=\int_{\ell}^{\infty}\frac{\mu}{x}f_i(x)dx=\int_{\ell}^{\mu}\frac{\mu}{x}f_i(x)dx+\int_{\mu}^{\infty}\frac{\mu}{x}f_i(x)dx\le\phi\mu\int_{\ell}^{\mu}\frac{1}{x}dx+\int_{\mu}^{\infty}\frac{\mu}{\mu}f_i(x)dx=$$

$$\phi\mu[\ln(x)]_{x=\ell}^{x=\mu}+\int_{\mu}^{\infty}f_i(x)dx=\phi\mu(\ln(\mu)-\ln(\mu/2^n))+1\le\phi\mu n+1,$$

where we have used the fact that $\int_{\mu}^{\infty}f_i(x)dx\le\int_0^{\infty}f_i(x)dx=1$. Putting the two bounds together, we obtain $\mathbf{E}\left[Z\right]\le\log(\phi\mu n+1)+2\log(n)$. □