

Inequity Aversion Pricing over Social Networks: Approximation Algorithms and Hardness Results*

Georgios Amanatidis¹, Evangelos Markakis², and Krzysztof Sornat³

- 1 Athens University of Economics and Business, Athens, Greece
gamana@aueb.gr
- 2 Athens University of Economics and Business, Athens, Greece
markakis@gmail.com
- 3 University of Wrocław, Wrocław, Poland
krzysztof.sornat@cs.uni.wroc.pl

Abstract

We study a revenue maximization problem in the context of social networks. Namely, we consider a model introduced by Alon, Mansour, and Tennenholtz (EC 2013) that captures *inequity aversion*, i.e., prices offered to neighboring vertices should not be significantly different. We first provide approximation algorithms for a natural class of instances, referred to as the class of single-value revenue functions. Our results improve on the current state of the art, especially when the number of distinct prices is small. This applies, for example, to settings where the seller will only consider a fixed number of discount types or special offers. We then resolve one of the open questions posed in Alon et al., by establishing APX-hardness for the problem. Surprisingly, we further show that the problem is NP-complete even when the price differences are allowed to be relatively large. Finally, we also provide some extensions of the model of Alon et al., regarding the allowed set of prices.

1998 ACM Subject Classification F.2 Analysis of Algorithms and Problem Complexity

Keywords and phrases inequity aversion, social networks, revenue maximization

Digital Object Identifier 10.4230/LIPIcs.MFCS.2016.09

1 Introduction

We study a differential pricing optimization problem in the presence of network effects. Differential pricing is a well known practice in everyday life and refers to offering a different price to potential customers for the same service or good. Examples include offering cheaper prices when launching a new product, making special offers to gold and silver members of an airline miles program, offering discounts at stores during selected periods, and several others.

We are interested in studying differential pricing in the context of a social network. Imagine a network connecting individuals (who are seen as potential clients here) with their friends, family, or colleagues, i.e., with people who can exert some influence on them. One can have in mind other forms of abstract networks as well, e.g., a node could represent a geographic region, a neighborhood within a city, a type of profession, a social class, and edges can represent interactions or proximity. The presence of such a network creates *externality*

* This research was supported by National Science Centre, Poland, 2015/17/N/ST6/03684. It was also supported by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) - Research Funding Program: THALES.



© Georgios Amanatidis, Evangelos Markakis, and Krzysztof Sornat;
licensed under Creative Commons License CC-BY

41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016).

Editors: Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier; Article No. 09; pp. 09:1–09:13

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

effects, meaning that the decision of a node to acquire a new product or a new service is affected by the fact that some other nodes within her social circle (her neighborhood in the graph) already did so. A typical example of positive externalities is when someone becomes more likely to buy a new product due to the positive reviews by a friend who already bought it in the past. Modeling positive externalities has led to a series of works that study marketing strategies for maximizing the diffusion of a new product, [7, 14], or the total revenue achieved, [13] (see also the Related Work section).

However, there also exist negative externality effects that can arise in a network. One example is the purchase of a product with the intention to show off and be a locally unique owner, e.g., a new type of expensive car, or clothes (also referred to as *invidious consumption*, see [5]). In such a case, a node may be deterred from buying the same product, if a neighboring node already did so. A second example of negative externalities, which is the focus of our work, and arises from differential pricing, is *inequity aversion*, see e.g., [4] and [8]. This simply means that a customer may experience dissatisfaction if she realizes that other people within her social circle, were offered a better deal for the same service. Hence, significant price differences, can create a negative response of some customers towards a product. Inequity aversion can also arise under a different, but equally applicable, interpretation: nodes may correspond to retail stores and an edge can signify proximity, so that clients could choose among these stores. Again, having significantly different prices to the same products is not desirable.

To capture the need for avoiding such phenomena, the relatively recent work of [2] introduced a model for pricing nodes over a social network. The main idea is to impose constraints on each edge, specifying that the price difference between two neighbors should be bounded by some (endogenous) parameter, determined by the two neighbors. On top of this, the seller is also allowed to not make a price offer to some nodes (referred to as introducing *discontinuities*, see the related discussion in Section 2), in which case the difference constraints do not apply for the edges incident to these nodes. Assuming a finite set of available prices, unit-demand users, and digital goods (i.e., the supply can cover all the demand) the problem is to find a feasible price vector that satisfies the edge constraints and maximizes the total revenue. In its more general form the problem was shown to be NP-complete, but exact or approximation algorithms were derived for some interesting cases.

Contribution: We revisit the model introduced by [2] (namely Model II of [2], which is the more general one), and study the approximability of the underlying revenue maximization problem. We resolve one of the open questions posed in [2], regarding the complexity of the problem under the natural class of the so-called *single-value* revenue functions. Simply put, this means that the revenue extracted by each node is exactly the price offered to her, as long as the price does not exceed her valuation for the product (the usual assumption made in auction settings as well). We first establish APX-hardness for this class answering the question of [2], and we also show that the problem is NP-complete even when the price differences are allowed to be relatively large (a case that could be thought easier to handle). We then provide approximation algorithms that improve some of the currently known results. Our improvement is stronger when the number of distinct prices is small. This applies for example to many settings where the seller will only consider a fixed number of discount types or special offers to selected customers. As the number of available price offers becomes large, the performance of our algorithm degrades to a logarithmic approximation. Finally, we provide an extension of these results to a more general model where the allowed prices come from a set of k arbitrary integers, instead of using price sets of the form $\{1, 2, \dots, k\}$, as done in [2] (see Subsection 4.3).

Related Work: Price discrimination is well studied in various domains in economics and is also being applied to numerous real life scenarios. The algorithmic problem of differential pricing over social networks is a more recent topic, initiated by [13]. The work of [13] studied a model with positive externalities, where the valuation of a player may increase as more friends acquire a good, and analyzed the performance of a very intuitive class of pricing strategies. Further improvements on the performance of such strategies were obtained later on by [9]. The work of [1] also considers a pricing problem but in an iterative fashion, where the seller is allowed to reprice a good in future rounds. Revenue maximization under a mechanism design approach was also taken in [12] under positive network externalities. Finally, positive externalities have been used to model the diffusion of products on a network, see, among others, the exposition in [15].

Negative externalities within networks, as we focus on here, are less studied in the literature. For the concept of inequity aversion, see e.g., [4, 8]. The work most closely related to ours is [2], which introduced the model that we consider here. Efficient algorithms were obtained for the case where discontinuities are not allowed (even for more general revenue functions), and also for networks with bounded treewidth. An approximation ratio of $1/(\Delta + 1)$ was also provided, where Δ is the maximum degree. Similar results were shown for a stochastic version of the model. Finally, other types of negative externalities have been considered e.g., in [3, 5] which study the effects of invidious consumption.

2 Definitions and Preliminaries

The social network is represented as an undirected graph $G = (V, E)$, with $|V| = n$. We assume that a provider of some good or service has a finite set P of available prices that he could offer to the nodes. In most of our presentation, we assume as in [2], that the available prices are given by $P = \{1, 2, \dots, k\}$. In Subsection 4.3, we show how to extend the analysis when P is an arbitrary set of k positive integers, i.e., $P = \{p_1, p_2, \dots, p_k\}$.

We assume every node has a unit-demand for the same product and that the supply of the seller is enough to cover the demand of all nodes. For every node $v \in V$, we associate a revenue function $R_v : \{1, 2, \dots, k\} \mapsto \mathbb{N}$ that maps an offered price p_v to the revenue that the provider gains from this offer. In this paper, we focus on a simple and intuitive class of revenue functions, also studied in [2]. In particular, for a node $v \in V$, R_v is called a *single value revenue function*, if there exists a value $val(v)$ such that when offered a price p_v :

$$R_v(p_v) = \begin{cases} p_v & \text{if } val(v) \geq p_v \\ 0 & \text{if } val(v) < p_v \end{cases}$$

We assume from now on that every node has a single value revenue function. We also assume that $val(v) \in P$, for every $v \in V$. This is because for revenue maximization, that we are interested in, nodes with $val(v) > k$, can only yield a revenue of k , and could be replaced by $val(v) = k$, i.e., the highest possible price. Also for values that are less than k , and not integers, we can again extract only an integer revenue, given the form of P . Finally, any node v with $val(v) < 1$ can be deleted without affecting the optimal revenue (see the concept of *discontinuity* defined below), so we can completely ignore such nodes to begin with. Thus, we consider only instances with $val(v) \in \{1, 2, \dots, k\}$, $\forall v \in V$.

Given a vector $\mathbf{p} = (p_v)_{v \in V}$ of prices offered to the nodes, the total revenue is $R(\mathbf{p}) = \sum_{v \in V} R_v(p_v)$. Hence, our goal is to find a price vector that maximizes the total revenue. At the same time, however, we want to capture the effect of *inequity aversion* [4, 8] in social networks. This means that a node may experience dissatisfaction if she sees that other nodes

within her social circle, were offered a better deal for the same service. Hence, significant price differences, create negative externalities among users.

To avoid such situations the model introduced in [2] has constraints on each edge, stating that the price difference between two neighbors u, v is bounded, i.e., $p_u - p_v \leq \alpha(u, v)$ and $p_v - p_u \leq \alpha(v, u)$, for every $(u, v) \in E$. Here, $\alpha(\cdot, \cdot) \geq 0$ is integer-valued (given that the prices are also integers) and note that in general is non-symmetric. Furthermore, the seller is also allowed not to make an offer to certain nodes. Formally, this is captured by having one more price option, which we denote by \perp , with $R_v(\perp) = 0$. Setting $p_v = \perp$ to a node, means that the provider does not make any offer to v , and there is no price restriction on the edges that are incident to v . We can essentially think about this as deleting these vertices from the graph. We will refer to setting $p_v = \perp$ to a node $v \in V$, as introducing a *discontinuity* on v . Avoiding making an offer can be thought of as choosing not to promote a product or service within a certain region or within a certain social group. In terms of optimization, allowing discontinuities can help the seller in producing much higher revenue (than without discontinuities) as Proposition 3 in Section 3 states.

Given this model, the set of feasible price vectors is then: $\mathcal{F} = \{\mathbf{p} : \forall v \in V, p_v \in P \cup \{\perp\}, \text{ and } \forall (u, v) \in E, p_u \neq \perp \wedge p_v \neq \perp \Rightarrow p_u - p_v \leq \alpha(u, v) \wedge p_v - p_u \leq \alpha(v, u)\}$. Therefore, the problem we study is:

Inequity Aversion Pricing: Given a graph with edge constraints, and a single-value revenue function for each node, find a feasible price vector that maximizes the total revenue, i.e., find $\mathbf{p} \in \mathcal{F}$ that achieves $\max_{\mathbf{p} \in \mathcal{F}} \sum_{v \in V} R_v(p_v)$.

Some cases of this problem, as well as the variant where no discontinuities are allowed, are already known to be polynomial time solvable [2]. Regarding hardness, although the problem is NP-hard for more general revenue functions, it was posed as an open question whether NP-hardness still holds for single value revenue functions (the hardness result in [2] requires instances with revenue functions that cannot be captured by single value ones).

3 Warm-up: Basic Facts and Single-price Solutions

In this section, we present a simple algorithm and some basic observations, which we use later on, in Section 4.

Let $v_{\max} = \max_{v \in V} val(v) \leq k$, and $\text{MAX} = \sum_{v \in V} val(v)$. Given an instance of the problem, we denote by OPT the revenue of an optimal solution. The quantity MAX is clearly an upper bound on the optimal revenue, hence $\text{OPT} \leq \text{MAX}$.

We will refer to a solution as being a *single-price* solution, if it charges the same price to every node without introducing discontinuities. Note that this is always a feasible solution since all the edge constraints are satisfied. The revenue extracted by a single-price algorithm that uses the price of p for all nodes is equal to $p \cdot |\{v \in V : val(v) \geq p\}|$.

To understand whether single-price solution can be of any help for our setting, we can examine the performance of the best possible single price. The following observation suggests that we do not need to try too many values, even if v_{\max} is very large.

► **Lemma 1.** *In order to find the optimal single-price solution, it suffices to check at most $\min\{n, v_{\max}\}$ possible prices.*

Proof. There are at most $\min\{n, v_{\max}\}$ different values in the set $\{val(v) : v \in V\}$. It is never optimal to use any price $p \notin \{val(v) : v \in V\}$. Indeed, if $p \in (val(v_1), val(v_2))$, where $val(v_1)$ and $val(v_2)$ are two consecutive distinct values for some nodes $v_1, v_2 \in V$, then it is

strictly better to set the price to $val(v_2)$. For the same reason, it is suboptimal to set a price that is less than the minimum value across nodes, while if we use a price $p > v_{\max}$ then we gain no revenue. ◀

Hence in $O(\min\{n, v_{\max}\})$ steps, we can select the best single-price solution. Let us denote by R_{SP} the revenue raised by this solution. The performance of R_{SP} has been analyzed in a different context¹ by [11], where it was shown that it achieves a $\Theta(\ln n)$ -approximation. Here we give a slightly tighter statement, which we utilize in later sections for small values of v_{\max} .

► **Theorem 2.** *For any number n of agents, the optimal single-price solution achieves a $1/H_r$ -approximation, where $r = \min\{n, v_{\max}\}$, and H_ℓ is the ℓ -th harmonic number, i.e.,*

$$R_{\text{SP}} \geq \frac{\text{MAX}}{H_r} \geq \frac{\text{OPT}}{H_r}.$$

Furthermore, the approximation guarantee is tight.

The proof is deferred to the full version of the paper. One interesting point here, is that single-price solutions do not use any discontinuities. If R_{ND} is the maximum revenue without using any discontinuities, clearly $R_{\text{ND}} \geq R_{\text{SP}}$. And as we mentioned in Section 2, it is possible to find the optimal solution that does not use discontinuities in polynomial time; so why use something worse instead of R_{ND} ? Actually, besides being harder to argue about, R_{ND} turns out to be as bad an approximation as R_{SP} , in the worst case. Hence, the proposition below reveals that introducing discontinuities can cause a significant increase in the optimal revenue achievable by the seller, compared to what can be achieved without discontinuities.

► **Proposition 3.** *The optimal solution with no discontinuities achieves a $1/H_r$ -approximation, where $r = \min\{n, v_{\max}\}$, and this approximation guarantee is tight.*

The proof of Proposition 3 is deferred to the full version of the paper.

4 Approximation of Inequity Aversion Pricing

In this section we present new approximation algorithms for the problem by exploiting ways in which setting discontinuities in certain nodes can help. Our main result is an approximation algorithm, with a ratio of $(H_k - 0.25)^{-1}$. Even though asymptotically this is no better than the optimal single-price algorithm, it does yield better ratios for instances where k is a small constant. The motivation for studying cases where the set of available prices is a small constant is that a seller may be willing to offer only specific types of discount to selected customers, e.g., 20% or 30% off the regular price and so on, rather than using an arbitrary set of prices.

We start below with the case of $k = 2$, before we move to the more general case.

4.1 A 0.8-approximation Algorithm when $P = \{1, 2\}$ via Vertex Cover

In this subsection, we assume the available prices are 1, 2, or \perp . Despite this restriction, the problem still remains non-trivial, and it is currently not known if it is NP-complete

¹ The work of [11] studied an auction pricing problem without the presence of social networks.

Algorithm 1: A 0.8-approximation when $P = \{1, 2\}$

- 1 Given the graph $G = (V, E)$, construct the bipartite graph $G' = (V_1, V_2, E')$ with $V_i = \{v \in V : \text{val}(v) = i\}$ and $E' = \{(u, v) \in E : \text{val}(u) = 2, \text{val}(v) = 1, \alpha(u, v) = 0\}$
 - 2 Find a minimum vertex cover on G' , say $S \subseteq V$
 - 3 Set \perp to all vertices of S
 - 4 Set a price of 1 to every $v \in V_1 \setminus S$ and a price of 2 to every $v \in V_2 \setminus S$. Let R^* be the revenue obtained by this solution
 - 5 Compute the optimal single-price solution, as described in Section 3, with revenue R_{SP}
 - 6 Return the solution that achieves $\max\{R^*, R_{\text{SP}}\}$
-

or not. Given the discussion in Section 2, we will also assume that for every node $v \in V$, $\text{val}(v) \in \{1, 2\}$. For such instances we already have a $\frac{2}{3}$ -approximation by Theorem 2, that does not use discontinuities. The difficulty in improving this factor is in finding a way of selecting appropriate nodes to set to \perp .

Before we describe our algorithm, let us illustrate the main idea. Consider an instance of the problem on a graph $G = (V, E)$. Suppose we plan to find a feasible price vector, such that for each u , either $p_u = \perp$ or $p_u = \text{val}(u)$. Since the possible prices are only 1 and 2, if $\text{val}(u) = 1$, then for any $(u, v) \in E$, $\alpha(u, v)$ is not restrictive, while if $\text{val}(u) = 2$, then for any $(u, v) \in E$, $\alpha(u, v)$ is restrictive only if $\alpha(u, v) = 0$ and $\text{val}(v) = 1$. So, we could remove any edge except from edges in $E' = \{(u, v) \in E : \text{val}(u) = 2, \text{val}(v) = 1, \alpha(u, v) = 0\}$. Note that this defines a bipartite subgraph $G' = (V_1, V_2, E')$ of G , where $V_i = \{v \in V : \text{val}(v) = i\}$. Since this new instance has less restrictions, the optimal revenue OPT' is at least as good as the optimal revenue OPT of the original instance.

Consider a vertex cover S in G' . The crucial observation is that we can satisfy all the edge constraints regarding edges between V_1 and V_2 , by introducing discontinuities on the vertices of S . Since S covers all the edges between V_1 and V_2 , the edge constraints between V_1 and V_2 in the original graph G are now non-existent. If we also set a price of 1 on the remaining vertices of V_1 and a price of 2 on the remaining vertices of V_2 , all the original constraints are satisfied. Thus, we have constructed a feasible solution for G .

The revenue of such a solution is $\text{MAX} - \text{val}(S)$, where $\text{MAX} = \sum_{v \in V} \text{val}(v) = |V_1| + 2 \cdot |V_2|$ and $\text{val}(S) = \sum_{v \in S} \text{val}(v)$. Hence, the best outcome of such an algorithm is achieved when S is a minimum weighted vertex cover (using the values as weights) rather than just any vertex cover. For the analysis however, it suffices to compute just a minimum vertex cover (see the Remark after the proof of Theorem 4). Moreover, by the König–Egerváry Theorem, we can compute this in polynomial time for bipartite graphs (e.g., see [16]).

Finally, the algorithm compares the best of two outcomes, the solution outlined above and the solution discussed in Section 3. Hence, we define $\text{ALG} = \max\{R_{\text{SP}}, \text{MAX} - \text{val}(S)\}$, where R_{SP} is the maximum revenue achieved by setting a fixed price to every node.

► **Theorem 4.** *Algorithm 1 achieves a 0.8-approximation for the Inequity Aversion Pricing problem when $P = \{1, 2\}$. Furthermore, this ratio is tight.*

Proof. Let ALG denote the revenue obtained by Algorithm 1 and let β be its approximation ratio that we attempt to determine. Assume that $\beta < 0.8$. Then there exists some $\epsilon > 0$ such that $\beta = 0.8 - \epsilon$. To arrive at a contradiction, we are going to show that $\beta \geq \gamma = 0.8 - \epsilon/2$.

We will distinguish some cases, depending on the value of ALG . First of all, note that if $\text{ALG} \geq \gamma \cdot \text{MAX}$, then we trivially obtain a γ -approximation: $\frac{\text{ALG}}{\text{OPT}} \geq \frac{\gamma \cdot \text{MAX}}{\text{MAX}} \geq \gamma$. From now on, assume that $\text{ALG} < \gamma \cdot \text{MAX}$. The following turns out to be a very useful upper bound for OPT .

► **Claim 5.** Let S denote a minimum vertex cover in the graph G' (defined in step 1 of Algorithm 1). Then, $\text{OPT} \leq \text{OPT}' \leq \text{MAX} - |S|$.

Proof of Claim 5. The first inequality is straightforward (see also the discussion before the theorem). For the second inequality, note that by the König–Egerváry Theorem, the maximum matching in G' has the same cardinality as S . Let M be such a maximum matching. By the definition of G' , for each edge $(u, v) \in M$ the nodes u and v have different values, say $\text{val}(u) = 2$ and $\text{val}(v) = 1$. Because of (u, v) , an optimal solution must lose at least one unit of revenue in comparison with MAX. Indeed, since $\alpha(u, v) = 0$, an optimal solution either sets a discontinuity on one of these two nodes, or it sets the same price. If this common price is 1, we lose one unit from node v , whereas if it is 2 we do not extract revenue from u . The claim follows. ◀

We know that $\text{ALG} = \text{MAX} - \text{val}(S)$ and also $\text{val}(S) \leq 2|S|$. Thus, $|S| \geq \frac{1}{2}(\text{MAX} - \text{ALG})$. If we combine this with Claim 5, we have

$$\text{OPT} \leq \frac{1}{2}(\text{MAX} + \text{ALG}). \quad (1)$$

To proceed with the analysis, we divide the interval $[0, \gamma \cdot \text{MAX}]$ into smaller subintervals of the form $[\frac{i-1}{m} \cdot \gamma \cdot \text{MAX}, \frac{i}{m} \cdot \gamma \cdot \text{MAX})$ for some fixed large m and $i \in \{1, \dots, m\}$. Notice that m is just a parameter in the analysis and has nothing to do with the input. We consider cases depending on where exactly the value of ALG falls. In particular, let i^* be the following interval index: $i^* = \lceil \frac{m+2}{2-\gamma} \rceil$.

Case (i): $\text{ALG} \in [\frac{i-1}{m} \cdot \gamma \cdot \text{MAX}, \frac{i}{m} \cdot \gamma \cdot \text{MAX})$ with $i \geq i^*$.

Using inequality (1), we have:

$$\frac{\text{ALG}}{\text{OPT}} \geq \frac{\frac{i-1}{m} \cdot \gamma \cdot \text{MAX}}{\frac{1}{2}(\text{MAX} + \text{ALG})} \geq \frac{\frac{i-1}{m} \cdot \gamma \cdot \text{MAX}}{\frac{1}{2}(\text{MAX} + \frac{i}{m} \cdot \gamma \cdot \text{MAX})} = \frac{\frac{i-1}{m} \cdot \gamma}{\frac{1}{2}(1 + \frac{i}{m} \cdot \gamma)}.$$

In order to ensure a γ -approximation, it suffices to have

$$\frac{\frac{i-1}{m} \cdot \gamma}{\frac{1}{2}(1 + \frac{i}{m} \cdot \gamma)} \geq \gamma \iff \frac{i-1}{m} \geq \frac{1}{2} \left(1 + \frac{i}{m} \cdot \gamma\right) \iff i \geq \frac{m+2}{2-\gamma}.$$

But this last inequality holds since $i \geq i^*$. Therefore, in this case, the algorithm achieves a γ -approximation.

Case (ii): $\text{ALG} < \frac{i^*-1}{m} \cdot \gamma \cdot \text{MAX}$.

Again, we use inequality (1), but now the lower bound of ALG comes from Theorem 2, which gives a guarantee for the optimal single-price solution:

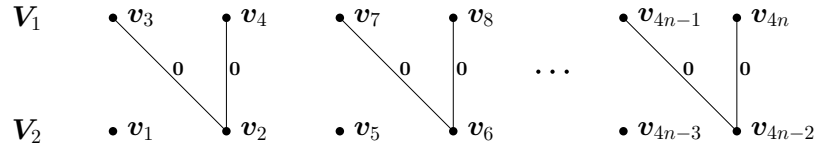
$$\frac{\text{ALG}}{\text{OPT}} \geq \frac{R_{\text{SP}}}{\frac{1}{2}(\text{MAX} + \text{ALG})} \geq \frac{\frac{1}{H_2} \text{MAX}}{\frac{1}{2} \text{MAX} (1 + \gamma \cdot \frac{i^*-1}{m})} = \frac{4/3}{1 + \gamma \cdot \frac{i^*-1}{m}}.$$

Like in case (i), it suffices to have

$$\frac{4/3}{1 + \gamma \cdot \frac{i^*-1}{m}} \geq \gamma \iff 4 \geq 3\gamma \left(1 + \gamma \cdot \frac{i^*-1}{m}\right).$$

Using an obvious upper bound for i^* , it suffices for γ to satisfy the following:

$$4 \geq 3\gamma + 3\gamma^2 \cdot \frac{\frac{m+2}{2-\gamma} + 1 - 1}{m} \iff \frac{6}{m} \gamma^2 + 10\gamma - 8 \leq 0.$$



■ **Figure 1** Algorithm 1 is tight on such instances. Only the relevant edges are shown.

Clearly, there is some $m^* \in \mathbb{N}$, such that

$$\frac{6}{m^*}(0.8 - \epsilon/2)^2 + 10(0.8 - \epsilon/2) - 8 \leq 0.$$

Thus, the approximation ratio β of Algorithm 1 is at least $0.8 - \epsilon/2$, which contradicts the choice of ϵ . Hence, $\beta \geq 0.8$.

To see why the ratio of the algorithm is tight, we can construct an infinite family of examples as follows: Consider a graph of 4 nodes $\{v_1, v_2, v_3, v_4\}$ such that $val(v_1) = val(v_2) = 2$, and $val(v_3) = val(v_4) = 1$. There are only two edges, namely (v_2, v_3) and (v_2, v_4) . Suppose $\alpha(\cdot, \cdot) = 0$. The optimal revenue here is 5 by offering a price of 1 to v_2, v_3, v_4 and a price of 2 to v_1 . On the other hand, the optimal single-price algorithm achieves a revenue of 4, either with a price of 1 or 2. Also, a minimum (weighted or not) vertex cover here is either $\{v_2\}$ or $\{v_3, v_4\}$. In both cases, the revenue by setting \perp to the vertex cover is 4. We can add many copies of this construction (and possibly some extra edges with $\alpha(e) \geq 1$ for a connected example) to turn this into an infinite family of tight examples. For an illustration, see Figure 1. ◀

► **Remark.** It seems appealing to try to exploit the fact that we can solve the minimum weighted vertex cover problem in polynomial time for bipartite graphs. However, as our analysis shows, using the weighted version of vertex cover, instead of the unweighted one, does not yield any better approximation.

4.2 An Approximation Algorithm for $k > 2$

We now consider the case where there are more than two available prices. In order to improve the approximation guarantee of Theorem 2, we reduce the problem to the case of $k = 2$, and use the results of the previous subsection.

Consider an instance of the problem, with available prices in $\{\perp, 1, 2, \dots, k\}$. As discussed in Section 2, we may assume that $val(v) \in \{1, 2, \dots, k\}$ for every $v \in V$. We create another instance, where we set the value of every node with $val(v) > 1$ to be equal to 2. We can then run Algorithm 1 from Subsection 4.1 on this new instance. At the same time, we can also compute the optimal single-price solution for the original instance, and pick the best among these two solutions. This yields Algorithm 2, described below.

Clearly, Algorithm 2 runs in polynomial time. Note that the solution returned by the algorithm is feasible. Any single-price solution is always feasible, while Algorithm 1 will produce a price vector that is feasible for I' , and therefore for I , since the edge restrictions in the two instances are the same. Even though asymptotically, this is still a logarithmic approximation, the algorithm achieves significantly better results for small values of k .

► **Theorem 6.** *Algorithm 2 achieves a $\frac{1}{H_{v_{\max}} - 0.25}$ -approximation ratio for Inequity Aversion Pricing when the available prices are $\{\perp, 1, 2, \dots, k\}$, with $k \geq 2$.*

Algorithm 2: An algorithm for $k > 2$

- 1 Given an instance I , construct a new instance I' , where for every $v \in V$,
 $val'(v) = \min\{val(v), 2\}$; everything else remains unchanged
 - 2 Run Algorithm 1 from Subsection 4.1 on instance I' , and let R_* be the revenue obtained
 - 3 Compute the optimal single-price solution without discontinuities, on the original instance I ,
as described in Section 3, with revenue R_{SP}
 - 4 Return the solution that achieves $\max\{R_*, R_{SP}\}$
-

Proof. The proof is by induction on v_{\max} . For $v_{\max} = 2$ the result follows from Theorem 4 since $0.8 = \frac{1}{H_2 - 0.25}$.

Now assume we have an instance I where $v_{\max} = j > 2$. As usual, let OPT denote the optimal revenue for I and ALG the revenue returned by Algorithm 2. Also, let R_j be the revenue extracted by setting price j at every node, and $V_j = \{v \in V : val(v) = j\}$. We consider two cases.

Case (i): $|V_j| \geq \frac{1}{(H_j - 0.25)j} \cdot \text{OPT}$. Then, $\frac{\text{ALG}}{\text{OPT}} \geq \frac{R_j}{\text{OPT}} = \frac{j \cdot |V_j|}{\text{OPT}} \geq \frac{\frac{1}{H_j - 0.25} \cdot \text{OPT}}{\text{OPT}} = \frac{1}{H_j - 0.25}$.

Case (ii): $|V_j| < \frac{1}{(H_j - 0.25)j} \cdot \text{OPT}$. Let I^* be an instance derived from I by setting $val^*(v) = \min\{val(v), j-1\}$, i.e., we only reduce the valuation of the nodes with $val(v) = v_{\max}$ by 1. Let OPT^* denote the optimal revenue for I^* , and ALG^* the revenue returned by Algorithm 2. By the inductive hypothesis we have $\text{ALG}^* \geq \frac{1}{H_{j-1} - 0.25} \cdot \text{OPT}^*$.

Furthermore, notice that the set of vertices with valuation greater than 1 is the same in both instances. So, Algorithm 2 on input I^* considers exactly the same price vectors as it does on input I , with the exception of the single-price solution that universally uses j . We conclude that $\text{ALG}^* \leq \text{ALG}$. Next, we prove the following useful claim.

► **Claim 7.** $\text{OPT}^* \geq \text{OPT} - |V_j|$.

Proof of Claim 7. Let \mathbf{p} be an optimal price vector for I . Construct the price vector \mathbf{p}^* by decreasing any price that is at least j to $j-1$. It is straightforward to see that in instance I we have $R(\mathbf{p}^*) \geq R(\mathbf{p}) - |V_j| = \text{OPT} - |V_j|$, while in both instances $R(\mathbf{p}^*)$ is the same. What is left to show is that \mathbf{p}^* is feasible for I^* . Observe, however, that the two instances have exactly the same edge restrictions and that, by its definition, \mathbf{p}^* did not increase the price difference between any two vertices compared to \mathbf{p} . Thus, $\text{OPT}^* \geq R(\mathbf{p}^*) \geq \text{OPT} - |V_j|$. ◀

Now, we can write

$$\begin{aligned} \frac{\text{ALG}}{\text{OPT}} &\geq \frac{\text{ALG}^*}{\text{OPT}} \geq \frac{\frac{1}{H_{j-1} - 0.25} \cdot \text{OPT}^*}{\text{OPT}} \geq \frac{\frac{1}{H_{j-1} - 0.25} \cdot (\text{OPT} - |V_j|)}{\text{OPT}} \\ &\geq \frac{1}{H_{j-1} - 0.25} \left(1 - \frac{\frac{1}{j(H_j - 0.25)} \cdot \text{OPT}}{\text{OPT}} \right) = \frac{1}{H_{j-1} - 0.25} \cdot \frac{jH_j - 0.25j - 1}{j(H_j - 0.25)} \\ &= \frac{1}{H_{j-1} - 0.25} \cdot \frac{j(H_{j-1} - 0.25)}{j(H_j - 0.25)} = \frac{1}{H_j - 0.25}, \end{aligned}$$

which concludes the proof. ◀

4.3 Approximation Algorithms for General Price Sets

We end Section 4 by extending our results when P is an arbitrary set of k positive integers, i.e., $P = \{p_1, p_2, \dots, p_k\}$. This can be seen as a more realistic model, especially for small values of k . In such a case, one could try to directly apply Theorems 2, 4, or 6 for $P' = \{1, 2, 3, \dots, p_k\}$.

■ **Table 1** Examples of obtained approximation ratios.

P	$\{1, 2\}$	$\{1, 2, 3\}$	$\{1, \dots, 100\}$	$\{10, 20, 25\}$	$\{3, 6, 10, 11\}$
$1/H_k$	0.667	0.545	0.193	0.545	0.48
Alg. 2, general α	0.8	0.631	0.202	–	–
Thm. 10, general $\alpha \parallel \alpha = 0$	0.8	0.631	0.202	0.597 \parallel 0.689	0.524 \parallel 0.574

However, this may produce a very poor approximation when k is small but p_k is large, and feasibility is not guaranteed either. In what follows, P_j denotes $\sum_{i=1}^j \frac{p_i - p_{i-1}}{p_i}$, where $p_0 = 0$.

We begin with a generalization of Theorem 2.

► **Theorem 8.** *For any number n of agents and possible prices $p_1 < p_2 < \dots < p_k$ the optimal single-price algorithm achieves a ρ -approximation, where $\rho = 1/\min\{H_n, P_k\}$, i.e.,*

$$R_{\text{SP}} \geq \frac{\text{MAX}}{\min\{H_n, P_k\}} \geq \frac{\text{OPT}}{\min\{H_n, P_k\}},$$

and this approximation guarantee is tight.

For $k = 2$, Theorem 8 yields an approximation ratio of $\frac{p_2}{2p_2 - p_1}$. We can still use the ideas of Theorem 4 to improve this factor. Notice, however, that although all of our results so far are independent of $\alpha(\cdot, \cdot)$, now the improvement will depend on the edge constraints. As in Algorithm 1, we can define a bipartite graph by using a restricted subset of the edges of G . In analogy to the set E' in section 4.1, we let $E' = \{(u, v) \in E : \text{val}(u) = p_2, \text{val}(v) = p_1, \text{ and } \alpha(u, v) < p_2 - p_1\}$, and $\alpha = \max_{(v_1, v_2) \in E'} \alpha(v_2, v_1)$. We have the following.

► **Theorem 9.** *When $P = \{p_1, p_2\}$ there is a polynomial time ρ -approximation algorithm for the Inequity Aversion Pricing problem, where $\rho = \frac{p_2^2}{2p_2^2 - p_1 p_2 - (p_2 - p_1) \min(p_1, p_2 - p_1 - \alpha)}$. Furthermore, this ratio is tight.*

Notice that Theorem 9 yields a 0.8-approximation when $P = \{1, 2\}$. Finally, based on the improved approximation for two prices, we can get an analog of Theorem 6 for any number of distinct prices. Given an instance I , let I' be the new instance where for every $v \in V$, $\text{val}'(v) = \min\{\text{val}(v), p_2\}$, while the constraints remain the same.

► **Theorem 10.** *Let $P = \{p_1, p_2, \dots, p_k\}$, and suppose that on instance I' (described above) the algorithm implied by Theorem 9 gives a $\frac{1}{P_2 - x}$ -approximate solution. Then, we can get a $\frac{1}{P_k - x}$ -approximate solution for the original instance of the Inequity Aversion Pricing problem in polynomial time.*

The proofs of all results in this subsection are deferred to the full version of the paper. We note however, that the algorithms and the proofs for Theorems 9 and 10 are similar to the corresponding algorithms and proofs for Theorems 4 and 6 respectively.

Table 1 summarizes approximation ratios obtained by the optimal single price solution, Algorithm 2, as well as the algorithm implied by Theorem 10 for different sets of prices.

5 Hardness for Single Value Revenue Functions

In [2] there is an $n^{1-\epsilon}$ inapproximability result for Inequity Aversion Pricing, but for general revenue functions and $\alpha(u, v) = 1$ for every edge. An NP-hardness proof is also given for these edge constraints when single value and constant revenue functions are allowed. The NP-hardness of Inequity Aversion Pricing as we study it here, i.e., allowing only single value

revenue functions, was left as an open question. We resolve this question by proving that the problem remains NP-complete even if we restrict the revenue functions to be single value. Our reduction implies that the result holds even when the price differences are allowed to be close to the maximum possible price k . Further, when $\alpha(u, v) = 0$ for every edge, we are able to show APX-hardness.

The reduction, below, is from the decision version of *3-Terminal Node Cut*: Given a graph $G(V, E)$, a set $S = \{v_1, v_2, v_3\} \subseteq V$, and an integer q , is there a subset of q vertices that can be deleted, so that any two vertices of S are in different connected components of the resulting graph? The NP-completeness of the weighted version of 3-Terminal Node Cut is discussed in [6], while the APX-hardness of the unweighted version we use here is discussed in [10]; in either case no explicit proof is given. The NP-completeness result we need follows from Theorem 15 as well.

► **Theorem 11.** *Let $\epsilon > 0$ be any small constant. The decision version of Inequity Aversion Pricing (for single value revenue functions) is NP-complete even when $\alpha(u, v)$ is as large as $k^{1-\epsilon}$ for all $(u, v) \in E(G)$, where k is the maximum possible price.*

Proof. It is immediate that the problem is in NP. To facilitate the presentation, we prove the NP-hardness when $\alpha(\cdot, \cdot)$ is upper bounded by $k^{1/3}/3$. As discussed at the end of the proof, the reduction can be easily adjusted when the upper bound of $\alpha(\cdot, \cdot)$ is $k^{1-\epsilon}$, for constant ϵ .

Let us consider an instance of 3-Terminal Node Cut, i.e., a graph $G(V, E)$, with $|V(G)| = n$, a set $S = \{v_1, v_2, v_3\}$ of non adjacent vertices of G , and an integer q . We may assume that $q \leq n - 3$, otherwise the question is trivial. Next we give a construction of an appropriate instance for Inequity Aversion Pricing.

Let H be the graph obtained from G as follows. We replace every vertex $v \in S$ by n^3 vertices, where each such vertex has the same neighbors as v , i.e., if u_v is a vertex in the bundle of vertices replacing v , then for every edge $(v, x) \in E(G)$ we add the edge (u_v, x) to $E(H)$. For any $v \in S$, we call such a set of vertices in H a v -bundle. The set of prices is $\{\perp, 1, 2, \dots, k\}$, where $k = n^3 + n^2$. Finally, for any $(u, v) \in E(H)$ we set $\alpha(u, v)$ and $\alpha(v, u)$ arbitrarily, as long as they are at most $k^{1/3}/3$. Note that $|V(H)| = n - 3 + 3n^3$, and $|E(H)| \leq |E(G)| + 3(n - 1)n^3 \leq 3n^4$.

Next we define the single value revenue functions for the vertices of H . For every $v \in V(G) \setminus S$, let $val(v) = n^3 + n^2$, and for every $v_i \in S$, let $val(u_{v_i}) = n^3 + \frac{i-1}{2}n^2$ for all u_{v_i} in the v_i -bundle. We show below that G has a subset of at most q vertices that separate all the vertices of S , if and only if there is a feasible choice of prices for the vertices of H that gives revenue at least R_q , where $R_q = (n - 3 - q)n^3 + \sum_{i=1}^3 n^3 (n^3 + \frac{i-1}{2}n^2)$.

One direction is easy. Let A be a subset of at most q vertices of G that separate the three vertices of S . For all $v \in A$ we put a discontinuity on the corresponding v in H . If we think of these vertices as removed from H , this creates several connected components. For any other vertex $u \in V(H)$, if u is in the same component as some v_i -bundle (or itself is one of the vertices of the v_i -bundle), set its price to $n^3 + \frac{i-1}{2}n^2$, otherwise set its price to $n^3 + n^2$. Notice that any vertex without a discontinuity produces revenue at least n^3 , while any vertex u_{v_i} in a v_i -bundle with $v_i \in S$ produces revenue exactly $n^3 + \frac{i-1}{2}n^2$. Now, it is straightforward to check that this price vector \mathbf{p} is feasible and gives enough revenue: $R(\mathbf{p}) = \sum_{u \in V(H)} R(u) \geq (n - 3 - q)n^3 + \sum_{i=1}^3 n^3 (n^3 + \frac{i-1}{2}n^2) = R_q$.

For the opposite direction we begin with a couple of observations. Assume that there is a price vector \mathbf{p}_* that gives revenue at least R_q . We claim that \mathbf{p}_* can have only a few discontinuities.

► **Claim 12.** *There is no feasible price vector \mathbf{p} with $R(\mathbf{p}) \geq R_q$ and more than q discontinuities.*

One immediate implication of Claim 12 is that for any $v \in S$ not every vertex in the v -bundle has price \perp . This holds because the v -bundle has n^3 vertices and only $q \leq n - 3$ of them can get \perp . This is crucial, because if we think of the vertices with price \perp as removed from H , then no two vertices are separated because of discontinuities in the v -bundles. In particular, we can completely ignore those discontinuities with respect to connectivity.

Let $D_{\mathbf{p}} = \{v \in V(G) \setminus S \mid \mathbf{p}_v = \perp\}$, i.e., $D_{\mathbf{p}}$ is the set of non terminal vertices in G that their corresponding vertices in H have discontinuities in \mathbf{p} . So far, by Claim 12, we have that $|D_{\mathbf{p}}| \leq q$. What is left to be shown is that these discontinuities separate the v -bundles, for any $v \in S$.

► **Claim 13.** *There is no feasible price vector \mathbf{p} such that $R(\mathbf{p}) \geq R_q$, and for some $v_i, v_j \in S$ vertices from both the v_i -bundle and the v_j -bundle are in the same connected component of the graph $H' = H - \{v \in V(H) \mid v \text{ is not in a bundle and } \mathbf{p}_v = \perp\}$.*

We conclude that $D_{\mathbf{p}}$ is a set of at most q vertices of G that separate all the vertices of S . This completes the proof for the case where $\alpha(\cdot, \cdot)$ is upper bounded by $k^{1/3}/3$.

► **Claim 14.** *The above reduction generalizes for $\alpha(\cdot, \cdot)$ upper bounded by $k^{1-\epsilon}$ for any positive constant ϵ .*

The proofs of Claims 12, 13, and 14 are deferred to the full version of the paper. ◀

For the special case where all the differences are 0, we show that the problem is APX-hard. In doing so, we prove that 3-Terminal Node Cut is MAX SNP-hard, and thus APX-hard. As noted already, MAX SNP-hardness of 3-Terminal Node Cut is discussed—but not explicitly proved—in [10]. Here, having this reduction is crucial, and we have therefore obtained an explicit construction, since eventually we need to show that 3-Terminal Node Cut restricted in a specific set of instances is MAX SNP-hard.

► **Theorem 15.** *Multi-Terminal Node Cut is MAX SNP-hard even for 3 terminals and all the weights equal to 1.*

► **Theorem 16.** *Inequity Aversion Pricing (for single value revenue functions) is APX-hard when $\alpha(e) = 0$ for all $e \in E(G)$.*

The proofs of Theorems 15 and 16 are deferred to the full version of the paper.

► **Remark.** The maximum price k in the instance constructed in the proof of Theorem 16 does not depend on the size of the problem. Given that there is some constant ρ beyond which it is hard to approximate 3-Terminal Node Cut, this means that there exists some constant k^* for which Inequity Aversion Pricing does not have a PTAS. Note that for such a k^* we do have a constant factor approximation, with factor $H_{k^*}^{-1}$.

6 Concluding remarks

We studied a revenue maximization problem under inequity aversion for the natural class of single-value revenue functions. Apart from establishing the first hardness results for this class, we also derived approximation algorithms based on combinatorial and graph-theoretic tools, which improve the state of the art when the set of available prices is small. We find this to be a realistic setting as special price offers are usually derived by specific discount and

promotion policies. Several questions still remain open. Even for $k = 2$, it is not known if the problem is NP-hard, or whether we can have better approximation ratios. Clearly, it would also be interesting to resolve the approximability for general k , i.e., can we have a better than $O(1/H_k)$ -approximation for large k ? Exploring further models of negative externalities is another attractive direction that is not as well studied as the case of positive externalities.

References

- 1 H. Akhlaghpour, M. Ghodsi, N. Haghpanah, H. Mahini, V. S. Mirrokni, and A. Nikzad. Optimal iterative pricing over social networks (extended abstract). In *Proceedings of the 6th Workshop on Internet and Network Economics, WINE 2010*, pages 415–423, 2010.
- 2 N. Alon, Y. Mansour, and M. Tennenholtz. Differential pricing with inequity aversion in social networks. In *ACM Conference on Economics and Computation, EC 2013*, pages 9–24, 2013.
- 3 S. Bhattacharya, J. Kulkarni, K. Munagala, and X. Xu. On allocations with negative externalities. In *Proceedings of the 7th Workshop on Internet and Network Economics, WINE 2011*, pages 25–36, 2011.
- 4 G. E. Bolton and A. Ockenfels. A theory of equity, reciprocity and competition. *American Economic Review*, 100:166–193, 2000.
- 5 Z. Cao, X. Chen, X. Hu, and C. Wang. Pricing in social networks with negative externalities. In *Proceedings of the 4th International Conference on Computational Social Networks, CSoNet 2015*, pages 14–25, 2015.
- 6 W. H. Cunningham. The optimal multiterminal cut problem. In *Reliability Of Computer And Communication Networks, Proceedings of a DIMACS Workshop, New Brunswick, New Jersey, USA, December 2-4, 1989*, pages 105–120, 1989.
- 7 P. Domingos and M. Richardson. Mining the network value of customers. In *Proceedings of the 7th ACM International Conference on Knowledge Discovery and Data Mining, KDD 2001*, pages 57–66, 2001.
- 8 E. Fehr and K. M. Schmidt. A theory of fairness, competition and co-operation. *Quarterly Journal of Economics*, 114:817–868, 1999.
- 9 D. Fotakis and P. Siminelakis. On the efficiency of influence-and-exploit strategies for revenue maximization under positive externalities. In *Proceedings of the 8th Workshop on Internet and Network Economics, WINE 2012*, pages 270–283, 2012.
- 10 N. Garg, V. V. Vazirani, and M. Yannakakis. Multiway cuts in node weighted graphs. *J. Algorithms*, 50(1):49–61, 2004.
- 11 A. Goldberg, J. Hartline, A. Karlin, M. Saks, and A. Wright. Competitive auctions. *Games and Economic Behavior*, 55(2):242–269, 2006.
- 12 N. Haghpanah, N. Immorlica, V. S. Mirrokni, and K. Munagala. Optimal auctions with positive network externalities. In *Proceedings of the 12th ACM Conference on Electronic Commerce, EC 2011*, pages 11–20, 2011.
- 13 J. Hartline, V. S. Mirrokni, and M. Sundararajan. Optimal marketing strategies over social networks. In *Proceedings of the 17th international conference on World Wide Web*, pages 189–198, 2008.
- 14 D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.
- 15 J. Kleinberg. Cascading behavior in networks: Algorithmic and economic issues. In N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, editors, *Algorithmic Game Theory*, chapter 24, pages 613–632. Cambridge University Press, Cambridge, 2007.
- 16 M. D. Plummer and L. Lovász. *Matching Theory*. North-Holland Mathematics Studies. Elsevier Science, 1986.