The Design of a Combinatorial Data Market

Dmitry Moor dmoor@ifi.uzh.ch
Sven Seuken seuken@ifi.uzh.ch
Tobias Grubenmann grubenmann@ifi.uzh.ch
Abraham Bernstein bernstein@ifi.uzh.ch
University of Zurich, Binzmuhlestrasse 14,
Zurich, 8050 Switzerland

Abstract

In this paper, we propose a market design solution for a data market. We focus on four specific challenges: (1) different providers have the capability to produce different sets of databases; (2) to answer typical queries from buyers, two or more databases must be joined; (3) data providers have high fixed costs for producing a database; and (4) buyers have combinatorial values over which databases are produced and thereby become available in the marketplace. The key idea of our solution is to use a reverse auction for the sellers, a posted-price mechanism for the buyers, and a fixed-point iteration algorithm for finding an outcome that balances the two sides of the market. Via simulations, we show how our market distributes the surplus between buyers and sellers. In particular, we demonstrate that our design rewards providers of "unique" data much more than providers of "common data."

1. Introduction

Many datasets on the Web are unstructured, i.e., they can be interpreted by humans but not by machines. There are numerous domains in which we would benefit greatly from data published in a *structured* way, for example, as a database. This allows machines to understand relationships between different pieces of data (Bernstein et al., 2016). Consequently, this significantly reduces the effort for humans to analyze lots of unstructured datasets that are discovered by traditional search engines. Instead, one could delegate this task to automatic query processing algorithms. For example, in the life sciences, researchers submit queries that join data from multiple databases provided by different companies. Each of these databases contains information on chemical compounds, disease data, biological function, and biomarkers. Automatic aggregation and processing of this data leads to faster and more efficient drug discovery (HCLS, 2001). Another example is IBM Watson, a large scale question answering system that defeated the human champions in the well-known TV show *Jeopardy*. This system heavily relies on querying structured data from distributed databases (Ferrucci et al., 2010) and has numerous applications in cancer treatment and clinical research, financial advisory, and retail.

Technology that enables automatic aggregation and processing of structured data already exists, for example, the *Web of Data* (WoD) (W3C, 2014). This technology does all the work of joining and processing the data and returns a precise answer to the user's query. However, despite the apparent power of the WoD approach, the technology has not yet seen widespread adoption. One of the reasons for this underutilization is of economic

nature: most of the data produced for the WoD so far was either subsidized by governments or produced at a loss (Buil-Aranda et al., 2013). This suggests that one of the most important reasons preventing wide adoption of the WoD is a lack of financial incentives for data providers to publish their data in a structured way. Indeed, data providers may incur high costs for producing their databases, i.e., for structuring their data and linking it against databases of other data providers. Naturally, data providers hope to recoup these costs. However, advertisement, the main source of income for many data publishers as well as traditional search engines, does not work in the Web of Data because in the WoD, data is processed by machines rather than by humans and the machine can simply ignore any ad. Therefore, new sources of revenue for data providers are needed. One possible way to achieve this is via a market in which providers sell data to users and trade is mediated by a market platform. In this paper, we propose the design of such a market.

1.1 Call for Data Markets

The need for data markets was recognized by both business and academic communities. In a recent McKinsey report (2016), for example, the authors explained the need for data markets by referring to the inefficient use of constantly increasing amounts of data produced by businesses adopting IoT technologies.

Schomm et al. (2013) provide a good overview of existing data markets. One prominent practical data market was operated by Microsoft with the Microsoft Azure Data Marketplace platform, but seized operation in 2016 due to a "lack of sustained customer interest" (Ramel, 2016). This lack of interest, however, does not imply a lack of demand for data. A more likely explanation is an inadequate business model. This explanation is supported by the fact that companies like Thomson Reuters, LexisNexis and Bloomberg still make large profits by selling access to their proprietary databases (Thomson-Reuters, 2015; Greg, 2011). However, it is not possible to easily combine and process their data with data from databases produced by other data providers.

There are numerous challenges when designing markets for *information goods* such as data. Already more than 20 years ago, Varian (1995, 1997) highlighted the problem of high sunk and low marginal production costs for these goods. Bakos and Brynjolfsson (1999) studied the problem that buyers may have high uncertainty regarding their valuations for information goods. More recently, Moor et al. (2015) argued that, in data markets, the combinatorial preferences of buyers should be taken into account when the buyers are able to join multiple databases.

1.2 Overview of our Approach

In this paper, we propose a new market design solution for a data market. We focus on three challenges: (1) databases are distributed (produced by different data providers) and can be joined to produce an answer to a query; (2) data providers have high fixed costs for producing a database; and (3) buyers have combinatorial values over which databases are available (i.e., different combinations of databases lead to more or less valuable answers to a buyer's query).

Goldberg et al. (2001), Goldberg and Hartline (2001, 2003) also studied markets for information goods. They proposed an auction for selling goods in unlimited supply (such

as data) in a setting with a single seller. However, they did not consider a setting with multiple distributed sellers or buyers with combinatorial values.

In recent papers by Balazinska et al. (2013), Koutris et al. (2013, 2015) and Deep and Koutris (2016), the authors aimed at designing a data market with quoted prices. The basic idea was to charge a different price for different *views* of the database in a way that would guarantee a *no-arbitrage* property. However, their approach does not allow joining data from multiple different data providers and ignores the costs of production.

While many authors (e.g., Varian (1995), Goldberg et al. (2001)) have previously studied the economic problem of how to sell an information good (such as music files or videos), their approaches do not translate to data markets. The main reason is the combinatorial structure that arises in a data market once we allow databases to be joined: a buyer's value for receiving answers based on one database may be zero while it may be very large once two databases are joined. This combinatorial structure is not present with music or video files which is why data markets require a new design.

The general approach we adopt for designing such a two-sided market is as follows. First, we design an auction to elicit the data providers' production costs. The objective of the auction is to allocate data providers (and respective databases) in a way that maximizes the total utility of buyers. This needs to be done subject to the constraint that the production costs of data providers are recouped. Second, we suggest a uniform posted pricing scheme that is presented to buyers submitting their queries. Apart from its simplicity, the use of uniform posted prices prevents complex strategic behavior of buyers who are assumed to be price-takers. Finally, we propose a mechanism that makes the overall market budget balanced, i.e., that guarantees that the total expected amount of money collected from buyers is equal to the total payment that needs to be accrued to data providers. While there are many possible equilibria that can arise in such a market (for example, a trivial equilibrium where nobody is allocated), our mechanism targets at finding the one with the highest surplus. This balancing mechanism ties together with the reverse auction on the one side and the posted price mechanism on the other side, and we consider this balancing mechanism to be the main contribution of this paper.

Scope of this work. To keep our model simple, we focus only on the most essential features of the domain such as high fixed costs and low marginal costs of production of data, the distributed nature of production and the combinatorial aspect of users' preferences. We argue that these features have the most significant implications on market design. The list of all possible features may include *privacy* and *quality* of data, endogenous demand or dynamic arrival of sellers to the market. Accounting for privacy and quality of data can make the model unreasonably complicated and are therefore outside the scope of the current paper. Endogenous demand and dynamic arrival of sellers are possible future extensions.

2. Preliminaries

2.1 Formal Model

Figure 1 provides a schematic overview of the market we will design. We assume that there are N sellers (i.e., data providers), $s_1, ..., s_N$; and L buyers (or users), $b_1, ..., b_L$, who submit their queries; $N, L > 0$ are given exogenously. To keep the model simple, we assume that

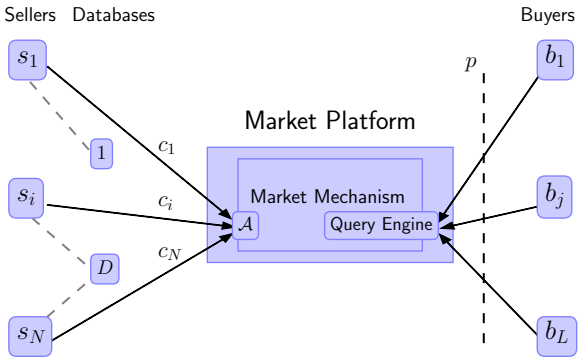

Figure 1: Market structure. Buyers consume rows of database tables corresponding to their queries for a posted price p per row. The reverse auction $\mathcal{A}$ elicits fixed costs of sellers. The market platform balances both sides of the market, i.e., the total payment collected from buyers needs to be equal to the total payment accrued to sellers.

each buyer submits a single query. The answer to a buyer's query consists of multiple rows of a database table (possibly joined over multiple databases) that satisfy the buyer's query. Thus, the buyer can buy *some* of these rows in exchange for money. This means that there are two goods involved in the exchange: *money* and *rows* of database tables corresponding to buyers' queries.

Market Structure. The market is mediated by a *market platform*. This market platform operates a so-called *query engine*, which executes buyers' queries and produces answers for these queries. Given a query from buyer b_j, the query engine takes databases of different sellers as inputs and then outputs $R_j \in \mathbb{N}$ rows by joining those databases.

In our market design, the market platform is a *neutral* (and, in particular, non-profit maximizing) entity for two primary reasons. First, observe that there are two levels of production: the sellers produce their databases, and the query engine then takes/joins those databases to produce answers to buyers' queries. Thus, it is highly convenient to consider the query engine a separate entity residing at the market platform. Second, one can show that in a domain with zero marginal costs of production (such as our domain), a (non-trivial) competitive equilibrium (i.e., where the seller maximizes her profits and the buyer maximizes his utility) is not guaranteed to exist (Mas-Colell et al., 1995).[1] Similar problems occur in markets with natural monopolies (Tirole & Laffont, 1993). In practice, such markets require a *regulator*, and the regulator is usually a governmental organization that induces prices based on its own analysis of production costs and market demand. This fact comprises our second argument for the use of the neutral (non-strategic) market platform that acts as a "regulator."

In our case, it is the market platform "sets prices" based on two key factors: the sellers' production costs and the market demand for the databases. To elicit the production costs of the sellers we argue for the use of a reverse auction $\mathcal{A}$ (defined formally later in this

1. This follows from the fact that the profit maximizing seller will produce the maximum possible number of rows at zero marginal cost if the price per row is positive. The buyer, however, may not be willing to pay for all these rows unless the price is 0.

section). We also argue for the use of a uniform posted price *per row* that is exposed to buyers for estimating the demand for databases (see Figure 1).

Sellers. We assume that every seller can produce a single database and that $c_i \in \mathbb{R}_{\geq 0}$ is the fixed cost of s_i for producing her database.[2][3] Let $c = (c_1, ..., c_N)$ be the cost profile of sellers. Let D be the number of different databases, $D \leq N$. For every database $k \in \{1, ..., D\}$ we let $i(k) = (i_1(k), ..., i_q(k))$ denote the indices of sellers that can produce the database k.

We assume that the c_i are independent random variables distributed according to cumulative probability distributions F_i, $i = 1, ..., N$. We let f_i be the corresponding probability density. We assume that $f_i(c_i)$ has full support on some interval $[\alpha_i, \beta_i]$. Then, the joint probability density is $f(c) = \prod_{i=1}^{N} f_i(c_i)$. Similarly, $f_{-i}(c_{-i}) = \prod_{j \neq i} f_j(c_j)$ is the joint probability density of all sellers except s_i.

Let $t = (t_1, ..., t_N)$ denote transfers (payments) received by s_i and $a = (a_1, ..., a_N)$ denote an allocation decision of the reverse auction $\mathcal{A}$, i.e., $a_i \in [0, 1]$ is the probability that s_i is allocated. [4] The utility of seller s_i is assumed to be quasi-linear, i.e, $u_i(a, c_i, t) = t_i - a_i c_i$.

Sellers are strategic and can thus misreport their costs. Let $\hat{c}_i$ denote the reported cost of s_i. Then, $(\hat{c}_1, ..., \hat{c}_N)$ is a reported cost profile of all sellers. Similarly, $\hat{c}_{-i} = (\hat{c}_1, ..., \hat{c}_{i-1}, \hat{c}_{i+1}, ..., \hat{c}_N)$ denotes a reported cost profile of all sellers except s_i.

Buyers. We assume that every buyer b_j is equipped with an initial *endowment* of money $e \in \mathbb{R}_{\geq 0}$. A buyer can use his endowment to acquire rows of the database table corresponding to his query and keep the rest of this money, $m_j \in \mathbb{R}_{\geq 0}$, in his wallet. Let $r_j \in \mathbb{N}$ denote the number of rows acquired by b_j.

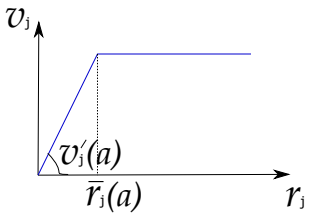

Figure 2: Value function of the buyer b_j when the allocation is a.

We assume that the buyers in our market are *institutional* agents. They could be pharmaceutical companies, operators of cloud applications or some other kind of intermediary. Importantly, we assume that the buyers in our domain can estimate their value for rows of the database tables. We assume that buyers are risk-neutral the b_j's preferences are described by a quasi-linear utility function $u_j(m_j, r_j, a) = v_j(r_j, a) + m_j$. Here, v_j is the value function of b_j. Notice that the value function depends on the allocation decision a of $\mathcal{A}$ regarding data providers. This follows from the assumption that the larger the number of allocated sellers, the more "informative" (and thus valuable) an answer for the buyer's query becomes. For simplicity, we assume that for any a, $v_j(r_j, a)$ is linear and non-decreasing in r_j up to a certain threshold $\bar{r}_j(a) \geq 0$, and exhibits zero marginal increase for every additional row $r_j > \bar{r}_j(a)$ (see Figure 2). [5] Formally, $v_j(r_j, a) = v'_j(a) \min\{r_j, \bar{r}_j(a)\}$. Here, $v'_j(a)$ is the marginal value for rows if the allocation is a. We let $F_{v'(a)}$ and $F_{\bar{r}(a)}$ denote the cumulative distribution functions of $v'_j(a)$ and $\bar{r}_j(a)$, respectively.

2. We assume marginal costs, i.e., costs of maintaining a database and answering queries, to be zero.
3. Throughout the paper we use "she" for sellers and "he" for buyers.
4. For technical reasons and simplicity of some proofs we assume that the allocation is probabilistic. The resulting mechanism that we will present in Section 3 however, will be deterministic.
5. Generally speaking, buyers could have a decreasing marginal value for additional rows. However, the assumption of a constant marginal value and a threshold is not too restrictive as it still captures convex preferences of buyers while providing us with a relatively simple model.

We make two assumptions regarding how buyers' preferences change when additional sellers (and thus, additional databases) are allocated. Assume that $a = (a_1, ..., a_i, ..., a_N)$ is an allocation and let us define $\delta a_i = (0, ..., \delta, ..., 0)$ with the ith element equal to δ, $0 \leq \delta \leq 1 - a_i$.

Assumption 1 (**Monotonicity of the Marginal Value**). *For every $j = 1, ..., L$, $\forall i = 1, ..., N$ the following inequality holds $v'_j(a + \delta a_i) \geq v'_j(a)$.*

The intuition behind this assumption is as follows: As more sellers are allocated, the marginal value of every buyer b_j for his query answer cannot decrease. This assumption is justified by the fact that the more information there is available, the more precise the answer to the query will be.

Assumption 2 (**Monotonicity of the Maximum Buyer's Value**). *For every $j = 1, ..., L$, $\forall i = 1, ..., N$ it holds $v'_j(a + \delta a_i)\overline{r}_j(a + \delta a_i) \geq v'_j(a)\overline{r}_j(a)$.*

This assumption can be interpreted as follows. Buyers submit their queries in order to answer a particular question. As answers for queries become more valuable (due to Assumption 1), each buyer can decide to buy more or fewer rows. However, this doesn't change the value of answering the particular question they had in mind: more information simply permits a better approximation of the question because it makes use of more databases. This makes the total value of a query answer, $v'_j(a)\overline{r}_j(a)$, non-decreasing.

Finally, we assume that buyers are indifferent between identities of sellers who produce a database k.

Allocation and Pricing. Let p denote the posted price per row in an answer to a query. The price p will be set independent of the query to prevent buyers from engaging in complex strategic behavior when deciding which queries to submit (e.g., to get cheaper answers to their questions). While this may seem counterintuitive at first sight, remember that producing any answer has zero marginal costs for the seller. Further, we assume the number of buyers L to be large, such that buyers are price takers and thus cannot manipulate the market price.

Given price p and allocation probabilities a, every buyer b_j solves his *consumption problem* of maximizing his utility subject to the budget constraint (Mas-Colell et al., 1995):

$$\begin{aligned} &\max_{m_j, r_j} u_j(m_j, r_j, a) \\ \text{s.t. } &p \cdot r_j + m_j \leq e, \\ &m_j \geq 0, r_j \leq R_j. \end{aligned} \tag{1}$$

Let $(m^*_j(p, a), r^*_j(p, a))$ be a solution to the consumption problem when the posted price is p and the allocation is a. Here, $r^*_j(p, a)$ is the *(Marshalian) demand of* b_j for rows when the posted price is p and the allocation is a. Similarly, $m^*_j(p, a)$ is the demand of the buyer for money (i.e., how much money the buyer wants to keep). Observe that $m^*_j(p, a)$ and $r^*_j(p, a)$ need not be functions. In fact, they are correspondences (Mas-Colell et al., 1995).

Let $\mathcal{A} = \langle g, h \rangle$ be the reverse auction adopted by the market platform. Here, $g : \mathbb{R}^N \to [0, 1]^N$ denotes an allocation rule that maps the cost profile $c = (c_1, ..., c_N)$ to the allocation decision $(a_1, ..., a_N)$; $a_i = g_i(c_i, c_{-i})$, where $g_i(c_i, c_{-i}) : \mathbb{R}^N \to [0, 1]$ computes the probability

that s_i is allocated. If a is an allocation, we let $\lfloor a \rfloor_k$ be the corresponding allocation in which sellers producing database k are not allocated. We let h denote the payment rule that maps the cost profile $c = (c_1, ..., c_N)$ to the vector of payments $(t_1, ..., t_N)$ to be paid to the sellers. As we shall see in Section 3, these payments need to be made "in expectation" due to the random nature of buyers' values and sellers' costs. As we discuss in Section 3, when $\mathcal{A}$ computes the allocation and payments it takes into account F_i for all $i = 1, ..., N$ as well as both $F_{v'(a)}$ and $F_{\overline{r}(a)}$ for all possible deterministic allocations $a \in \{0, 1\}^N$ of sellers.[6]

Given all of the above information, the market platform can compute the price per row $p \geq 0$, which is exposed to buyers, and payments $t_i \geq 0$ to be paid to sellers.

Remark 2.1. *In practice, the market platform will need to do some market research to learn the distributions $F_i, F_{v'(a)}, F_{\overline{r}(a)}$. Such learning can be done by building an appropriate regression model that captures the connectivity of different databases, their topics, the validity of the data, etc. The design of an appropriate learning procedure, however, is outside of the scope of this paper.*

Finally, we define the social welfare as the total utility of buyers and sellers obtained in the market given allocation a, payments t and the posted price p, i.e., $SW(a, t, p) = \sum_{i=1}^{N} u_i(a, c_i, t) + \sum_{j=1}^{L} u_j(m_j^*(p, a), r_j^*(p, a), a)$.

2.2 Market Properties

We now discuss a number of properties we would like the reverse auction $\mathcal{A}$ and the overall market mechanism to satisfy.

Auction Properties. We begin with the properties we would like the reverse auction $\mathcal{A}$ to satisfy.

Definition 1. *The reverse auction $\mathcal{A} = \langle g, h \rangle$ is **Bayes-Nash incentive compatible** (BNIC), if $\forall i = 1, ..., N$ $\forall c_i$ $\forall \hat{c}_i$ $\forall c_{-i}$*

$$\mathbb{E}_{f_{-i}}[u_i(g(c_i, c_{-i}), c_i, h(c_i, c_{-i}))] \geq \mathbb{E}_{f_{-i}}[u_i(g(\hat{c}_i, c_{-i}), c_i, h(\hat{c}_i, c_{-i}))]. \tag{2}$$

In our work, we look for a reverse auction $\mathcal{A}$ that satisfies BNIC.

The following property guarantees participation of sellers in the reverse auction:

Definition 2. *The reverse auction $\mathcal{A} = \langle g, h \rangle$ is **individually rational** (IR) for sellers, if $\forall i = 1, ..., N$, $\forall c_i, \forall c_{-i}$*

$$\mathbb{E}_{f_{-i}}[u_i(g(c_i, c_{-i}), c_i, h(c_i, c_{-i}))] \geq 0. \tag{3}$$

Market Mechanism Properties. Now, we switch to a discussion of the properties that the overall market mechanism should have.

First, observe that individual rationality is satisfied for buyers automatically. This follows from the fact that when solving their consumption problem (1), buyers always have the option not to consume rows and to keep their whole endowment e.

Additionally, we would like the market mechanism to be budget balanced. Formally:

6. Observe that as the number of databases increases, the number of possible deterministic allocations increases exponentially. In practice, the valuations of buyers for different allocations may be very similar. This would allow the market platform to considerably reduce the amount of information it needs to collect and this would also simplify pricing. In this paper however, we do not discuss such optimizations.

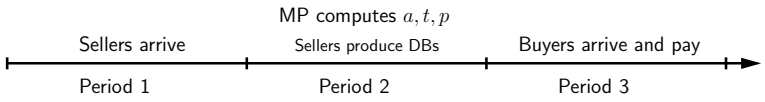

Figure 3: Temporal model of the market. During the first time period, sellers arrive and report their costs. In the second time period, the market platform computes an allocation and payments to sellers, as well as the posted price p. During the third time period, buyers arrive and consume the desired number of rows of database tables corresponding to their queries.

Definition 3. *The market mechanism is* ***budget balanced*** *(BB) if* $\forall c$, $\forall F_{v'(a)}, F_{\overline{r}(a)}$, $\forall F_i$ $(i = 1, ..., N)$*, the price* p*, the allocation and payments computed by* $\mathcal{A} = \langle g, h \rangle$ *satisfy*

$$\sum_{i=1}^{N} t_i = \sum_{j=1}^{L} (e - m_j^*(p, a)), \tag{4}$$

where $(t_1, ..., t_N) = h(c)$ *and* $a = g(c)$*.*

In words, the total payment to sellers computed by $\mathcal{A}$ should be equal to the total amount of money collected from buyers. As $\mathcal{A}$ uses $F_{v'(a)}, F_{\overline{r}(a)}$ and F_i, $i = 1, ..., N$ to compute the allocation and payments (see Section 2.1), this property should hold for arbitrary distributions.

2.3 Temporal Structure

Figure 3 illustrates the temporal structure of the data market we propose. First, for simplicity, we present the model with only three time periods and then elaborate on how it can be generalized for an arbitrary time horizon T.

In time period $\tau = 1$, all sellers s_i, $i = 1, ..., N$ arrive to the market and report their costs to the market platform. Then, in time period $\tau = 2$, the market platform computes allocation $a = (a_1, ..., a_N)$ and payments $t = (t_1, ..., t_N)$ as well as the posted price per row, p, based on reported costs received from sellers and the value model of buyers (i.e., $F_{v'(a)}$ and $F_{\overline{r}(a)}$). Once the allocation is computed, allocated sellers can produce their databases (this happens during the same time period $\tau = 2$). Finally, in time period $\tau = 3$, buyers arrive to the market, submit their queries and pay a price p per row of answers to their queries.

This simple model can be generalized straightforwardly to a setting where buyers do not arrive to the market at the same time, but over a certain time horizon T. In this setting, we assume that each database remains fully relevant (i.e., the value of the buyers remains the same) for T time periods, but that the data in the database becomes obsolete and thus has zero value after T time periods.[7] This means that we assume that T is the *timeliness* of data (i.e., a period of time during which the data is still up to date).

7. It is also possible to use a discounting factor for the value of data. We leave this direction for future work.

The market platform promises to the sellers that over the time horizon T, every allocated seller s_i will receive payments that are expected to add up to t_i.[8] This means that the market platform does not pay t_i to the seller s_i immediately at time $\tau = 1$. Instead, it will accrue money paid by buyers coming to the market over the time horizon T. We assume that L buyers arrive to the market over the time horizon T, each paying $e - m_j^*(p, a)$ to the market platform to get $r_j^*(p, a)$ rows of database tables (possibly joined) corresponding to their queries. These payments go directly to sellers until all promises are fulfilled (i.e., each seller s_i receives t_i).

3. Market Design

In this section, we present our main contribution: a market design solution for selling distributed data. We demonstrate how the reverse auction $\mathcal{A}$ should be designed as well as how the posted price p must be computed to achieve our design goals.

3.1 Market Design Objective and Constraints

Social Choice Function. The result of Myerson and Satterthwaite (1983) in domains when buyers and sellers are both strategic implies that there does not exist a social welfare maximizing mechanism that is BNIC, IR and budget balanced. Thus, optimizing social welfare is not feasible in this domain. Instead, we find it acceptable to sacrifice a bit of social welfare as long as the resulting market is BNIC and guarantees IR and BB. Given this, we can consider either optimizing the total utility of buyers or the revenues of sellers, subject to the aforementioned constraints.

As discussed in Section 1.1, non-negative profits for sellers is a crucial constraint for the viability of markets for distributed data. This is why maximizing the revenue of sellers could be a potential objective. This objective, however, does not seem very attractive, as we envision the market for distributed data to give rise to many novel AI applications coming from the buyers' side. Distributing all the surplus in favor of the sellers (who are often "monopolists" of their data) can make the market uninteresting for many potential buyers.

This is why we focus on optimizing the total utility of buyers subject to the constraint that the fixed costs of the allocated sellers can be recouped and the market is overall budget balanced.

3.2 Deriving a Value for Databases

To compute the allocation and payments of the sellers, we now derive the *induced* values of the buyers for the databases of the sellers. To this end, we first define the aggregate value of buyers for rows of database tables corresponding to their queries. Based on this aggregate value function, we can compute the positive externalities that different databases impose on buyers. Finally, we use these externalities to define buyers' values for different databases.

8. In our model, we assume that exactly L buyers arrive. In practice, more or fewer buyers would arrive. Our model extends straightforwardly to these cases by including an additional expectation over the number of buyers.

Given an allocation $a \in [0,1]^N$ and price per row p, the *market demand* for rows is $r^*(p,a) = \sum_{j=1}^{L} r_j^*(p,a)$. The market demand for money $m^*(p,a)$ is defined analogously, i.e., $m^*(p,a) = \sum_{j=1}^{L} m_j^*(p,a)$. We begin with the following definition:

Definition 4. *An **aggregate buyer** is a fictional buyer with an endowment $E = L \cdot e$ and the utility $U(m,r,a) = V(r,a) + m$. Here, $V(r,a)$ is an **aggregate value function**, i.e., a function that makes the solution of the consumption problem* (1) *for the aggregate buyer equal to the market demand $m^*(p,a), r^*(p,a)$.*

This means that the aggregate buyer is a fictional agent that acts in the same way as all buyers would act together when responding to the price p and the allocation a. The following proposition provides a way to compute the aggregate value function.

Proposition 3.1. *Given allocation $a \in [0,1]^N$, the aggregate value function is*

$$V(r,a) = \int_0^r \pi(z,a)dz, \tag{5}$$

where $\pi(z,a) = \max_{r^*(p',a)=z} p'$.[9]

Proof. From the Kuhn-Tucker conditions (Mas-Colell et al., 1995) for the aggregate consumer's problem (see Equation (1)), we have:

$$\frac{dV}{dr}(r^*,a) = p. \tag{6}$$

If $r^*(p,a)$ was a function, then the right hand side in Equation (6) would be the inverse demand function $r^{*-1}(p,a)$. However, $r^*(p,a)$ is a correspondence and, therefore, there can be many different prices p that support the solution $r^* = r^*(p,a)$. To resolve ambiguity, we let

$$\frac{dV}{dr}(r^*,a) = \max_{r^*(p',a)=r^*} p'. \tag{7}$$

Integrating Equation (7) and replacing $\max_{r^*(p',a)=z} p'$ with $\pi(z,a)$ we get $V(r,a) = \int_0^r \pi(z,a)dz$. Thus, solving Equation (6) with this aggregate value function gives us a solution $r^*(p,a)$, which is a market demand for rows. The demand of the aggregate buyer for money is then equal to $E - p \cdot r^*(p,a) = \sum_{j=1}^{L} e - p\sum_{j=1}^{L} r_j^*(p,a) = \sum_{j=1}^{L} m_j^*(p,a) = m^*(p,a)$, which is exactly the market demand for money. Thus, $V(r,a)$ is the aggregate value function. □

Now that we know how to compute the aggregate value function we can analyze some of its properties. We begin by showing that the aggregate value of buyers can only increase when more databases are allocated:

Proposition 3.2. *For all $r \geq 0$, $\forall a \in [0,1]^N$, $\forall i = 1,...,N$ the following holds:*

$$\frac{\partial V}{\partial a_i}(r,a) \geq 0.$$

9. This corresponds to the maximal *inverse demand function*.

Proof. Consider a single buyer b_j. Let $\delta a = (0, ..., \delta a_i, ..., 0)$, with $\delta a_i \geq 0$. Then $\forall r$, $v_j(r, a + \delta a) = v_j(r, a) + \delta a_i \frac{\partial v_j}{\partial a_i}(r, a)$. Consequently, $\frac{\partial v_j}{\partial a_i}(r, a) = \frac{1}{\delta a_i}(v_j(r, a + \delta a) - v_j(r, a))$. From Assumptions 1 and 2 it follows that $v_j(r, a + \delta a) \geq v_j(r, a)$ for all $r \geq 0$, $\forall a, \forall j = 1, ..., L$. Therefore, $\frac{\partial v_j}{\partial a_i}(r, a) \geq 0$. This means that every buyer's value for r rows can only increase with the increase of a_i. Consequently, the aggregate value can also only increase. □

Example 1. *Assume that $L = N = 2$ and that $e = 10$; each seller produces a single database. Consider a setting where s_1 is allocated while s_2 is not, i.e., $a = (1, 0)$. Assume b_j $(j = 1, 2)$ submits a query against the database of s_1 and has the following value function for her data: $v_1(r_1, a) = 4 \cdot \min\{r_1, 1\}$, $v_2(r_2, a) = 1 \cdot \min\{r_2, 2\}$. Buyers solve their consumption problems, see Equation (1). From the Kuhn-Tucker conditions we obtain the buyers' demands when the allocation is a:*[10]

$$r_1^*(p, a) = \begin{cases} 1, & \text{if } p < 4 \\ [0, 1], & \text{if } p = 4 \\ 0, & \text{otherwise} \end{cases} \qquad r_2^*(p, a) = \begin{cases} 2, & \text{if } p < 1 \\ [0, 2], & \text{if } p = 1 \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, the market demand is

$$r^*(p, a) = \begin{cases} 3, & \text{if } 0 < p < 1 \\ [1, 3], & \text{if } p = 1 \\ 1, & \text{if } 1 < p < 4 \\ [0, 1], & \text{if } p = 4 \\ 0, & \text{if } 4 < p. \end{cases} \tag{8}$$

It is easy to check that this market demand is equal to the demand of the aggregate buyer with an endowment of $E = 2e$ and utility $U(m, r, a) = V(r, a) + m$, where

$$V(r, a) = \begin{cases} 4r, & \text{if } 0 \leq r \leq 1 \\ 3 + r, & \text{if } 1 \leq r \leq 3 \\ 6, & \text{if } 3 \leq r \end{cases}$$

is the aggregate value function. For example, here the aggregate value for the first row $(0 \leq r \leq 1)$ is equal to 4. Therefore, if $p > 4$, the aggregate buyer's demand must be 0. Equation (8) confirms this as $r^*(p, a) = 0$ for $p > 4$. Other cases are analogous.

Remember, that if a is an allocation, then $\lfloor a \rfloor_k$ stands for a similar allocation in which the database k is not allocated. We now define the *externality* imposed by a database k on all buyers as follows:

10. The result is quite intuitive. Indeed, b_j is not willing to buy query answers as long as the price p per answer is larger than his marginal value for the answer (which is equal to 4 for b_1 and 1 for b_2). As soon as the price is smaller than the marginal value, b_1 and b_2 are willing to buy up to one and two query answers respectively.

Definition 5. *For a given allocation a and a posted price p, the externality imposed by the database k is*

$$ext_k(a,p) = V(r^*(p,a),a) - V(r^*(p,\lfloor a \rfloor_k),\lfloor a \rfloor_k). \tag{9}$$

The externality reflects how much additional value the database k brings to all buyers. Note that this quantity could be zero if the database k is not allocated in a. To define the value of buyers for a database k in a consistent way, we split the aggregate value achieved by all buyers proportionally to $ext_k(a,p)$. Formally:

Definition 6. *Given allocation $a \in \{0,1\}^N$ and price p, the* ***induced value $W_k(a,p)$ of the aggregate buyer for the database k*** *is the share of the aggregate value that is proportional to the externality that database k imposes on the aggregate buyer, i.e.,*

$$W_k(a,p) = \frac{ext_k(a,p)}{\sum_{\ell=1}^{D} ext_\ell(a,p)} V(r^*(p,a),a). \tag{10}$$

Observe that $W_k(a,p)$ depends on the posted price p and on the whole allocation a. Thus, indirectly it depends on allocations of all other databases. The dependency on allocation a reflects the important fact that buyers may have combinatorial valuations, i.e., they can value database k higher if a complementary database is also available.

The fact that $W_k(a,p)$ depends on price p has the following intuition: If price p is too high such that no one can afford to buy a single row, the externality imposed by any database is zero. If prices are low, externalities become positive.

We impose an additional assumption: all allocated databases are (weak) complements for the aggregate buyer. Thus, his valuation of databases is supermodular (Chambers & Echenique, 2009). Intuitively, supermodularity can be described as follows. Consider any two databases ℓ, k. Supermodularity says that the induced value of database k can only increase as the allocation probability of the other database ℓ increases. Formally:

Assumption 3 (**Weak Complementarity**)**.** *For any two databases ℓ and k, for any $a \in \{0,1\}^N$, $\forall p \geq 0$ the following inequality holds:*

$$W_k(a,p) \geq W_k(\lfloor a \rfloor_\ell, p). \tag{11}$$

Remark 3.1. *This assumption is quite intuitive: If the database ℓ complements another database k, an increase in allocation probability of ℓ leads to an increase in the induced value of database k for the aggregate buyer. Imposing this assumption on the aggregate buyer is natural: given that this buyer can be considered as the whole population, it makes sense to allocate those databases that are complementary. Note that, for individual buyers different databases can still be either complements or substitutes.*

While Assumption 3 is intuitive, it does not follow directly from Assumption 1 and Assumption 2. Indeed, one could construct an example where Assumptions 1 and 2 are satisfied but Assumption 3 is not. To eliminate these corner cases we state the Assumption 3 explicitly.

Example 2. *We follow the setup of Example 1. The externality that s_1 imposes on buyers when the allocation is $a = (1, 0)$ and the posted price is p is $ext_1(a, p) = V(r^*(p, a), a) - V(r^*(p, \lfloor a \rfloor_1), \lfloor a \rfloor_1)$, where $\lfloor a \rfloor_1 = (0, 0)$. Here, the aggregate value when the allocation is $\lfloor a \rfloor_1$ is $V(r, \lfloor a \rfloor_1) = 0$ as no queries can be answered. Similarly, the externality that s_2 imposes on buyers under allocation a is 0 (as s_2 is not allocated). Thus, for example, if $p = 1 - \epsilon$, then the market demand $r^*(p, a) = 3$ and the induced value of the database of s_1 is $W_1(p, a) = \frac{6}{6+0} \cdot 6 = 6$. Example 3 extends the current example for the case when all databases are allocated.*

3.3 Designing the Reverse Auction

Now that we have defined the induced values for different databases, $W_k(a, p)$, we can define an appropriate auction $\mathcal{A}$ that maximizes the total utility of buyers, subject to the constraint that the fixed costs of allocated sellers are recouped. In this auction, the market platform computes the allocation a and payments t based on the costs reported by sellers. We design this auction in a way that is similar to Myerson (1981) optimal auction.

First, observe that the expected total utility of buyers is equal to the difference between the expected value that the aggregate buyer can achieve under the allocation $g(c)$ and the total payment that the buyers must make to the sellers. Formally,

$$\begin{aligned}\mathbb{E}_f[U(g(c), h)] =& \mathbb{E}_f\Big[V(r^*(p, g(c)), g(c)) - \sum_{i=1}^{N} h_i(c)\Big] \\ =& \mathbb{E}_f\Big[\sum_{k=1}^{D} \mathbb{E}_{g(c)}\Big[W_k(a, p)\Big] - \sum_{i=1}^{N} h_i(c)\Big]. \end{aligned} \tag{12}$$

Here, $\mathbb{E}_{g(c)}\Big[W_k(a, p)\Big]$ is the expected induced value of the database k with the probabilistic allocation of sellers $g(c)$ (see the formal definition of this term in Appendix B). The market design problem now is to find an auction $\mathcal{A} = \langle g, h \rangle$ that maximizes $\mathbb{E}_f[U(g(c), h)]$ subject to BNIC, IR and the following constraints:

$$\sum_{i \in i(k)} g_i(c) \leq 1 \qquad \forall k = 1, ..., D \;\; \forall c \in [0, 1]^N, \tag{13}$$

$$g_i(c) \geq 0 \qquad \forall i = 1, ..., N \quad \forall c \in [0, 1]^N. \tag{14}$$

Constraints (13) and (14) ensure that each database is allocated at most once and that the allocation probabilities are non-negative.

We assume that the distributions of costs of sellers f_i are regular, i.e., monotone and strictly increasing (Myerson, 1981). We also let $\phi_i(c_i) = c_i + \frac{F_i(c_i)}{f_i(c_i)}$ denote the *virtual cost* of seller s_i. Let us also define the *virtual surplus* as $\widetilde{S}(a) = \sum_{k=1}^{D} W_k(a, p) - \sum_{i=1}^{N} \phi_i(c_i) a_i$, $a \in \{0, 1\}^N$. Now, we are ready to present the optimal reverse auction.

Buyer-Optimal Reverse Auction (BORA)

Allocation rule: $a^* \in \arg\max_{a \in \{0,1\}^N} \widetilde{S}(a)$; use random tie breaking in the case of ties.

Payment rule: For each seller s_i:
If $a_i = 1$, then

$$t_i = \phi_i^{-1}\Big(\phi_i(c_i) + \widetilde{S}(a^*) - \widetilde{S}(\lfloor a \rfloor_i^*)\Big). \tag{15}$$

If $a_i = 0$, then $t_i = 0$.

In words, the allocation rule says that the auction allocates sellers in a way that maximizes the virtual surplus. We break ties randomly. Informally, the payment of the allocated agent is computed in a similar way as VCG payments, where agents report their virtual costs instead of their true costs. To better understand the intuition behind the payment rule, let us consider several special cases.

One database, one seller. Let us first consider the setting with a single seller, i.e., $N = 1$, and consequently, $D = 1$. In this case, the seller is allocated whenever her virtual cost is smaller than the induced value of buyers for her database, i.e., $\phi_1(c_1) \leq W_1(a, p)$. From Equation (15) it follows that the payment to the seller must be equal to $t_1 = \phi_1^{-1}(W_1(a, p))$. This payment is similar to the Myerson (1981) *reserve payment*. In words, the reserve payment is equal to the cost that the seller should have had for her virtual cost to be equal to the induced value of the aggregate buyer for the respective database.

One database, multiple sellers. Let us now assume that there are multiple sellers that can produce the same database, i.e., $D = 1$ and $N > 1$. In this case, the seller with the smallest virtual cost is allocated as long as her virtual cost is smaller than the induced value of buyers for her database. W.l.o.g., let us assume that $\phi_1(c_1)$ is the smallest virtual cost and $\phi_2(c_2)$ is the second smallest virtual cost. Then, the payment of the allocated seller is equal to the minimum of $\phi_1^{-1}(\phi_2(c_2))$ and $\phi_1^{-1}(W_1(a, p))$, i.e., $t_1 = \min\{\phi_1^{-1}(\phi_2(c_2)), \phi_1^{-1}(W_1(a, p))\}$. In other words, the payment of the allocated agent is computed as the minimum of the reserve payment and the *critical value*. Here, the critical value is defined similarly to (Nisan et al., 2007), i.e., it is equal to the largest cost that the seller could have reported while still being allocated, i.e., $\phi_i^{-1}\Big(\min_{j \in i(k) \setminus i} \phi_j(c_j)\Big)$.

Two databases, two sellers. Consider the setting with two distinct databases, each produced by a single seller. Assume further that $\phi_1(c_1) > W_1(a, p)$ for $a = (1, 1)$ and for all p. In contrast to the setting with a single database and a single seller discussed above, in this case, one can happen that the database 1 is allocated (i.e., despite of the fact that its virtual cost is larger than the induced value of the respective database). Such a situation is possible, for example, when the database 1 has a very strong complementary effect on the database 2. Thus, the presence of the database 1 can increase the induced value of the second database, $W_2(a, p)$, as this induced value depends on the *whole* allocation a. As a result this may lead to a higher virtual surplus. Therefore, it may be optimal to allocate both databases. Example 4 in Appendix A illustrates this case.

Multiple databases, multiple sellers. In this most general case, the intuition behind the payment rule (15) mimics the intuition of the standard VCG mechanism. Indeed, the first two summand in the Equation (15), $(\phi_i(c_i) + \widetilde{S}(a^*))$, correspond to the total virtual

surplus achieved by all sellers apart of s_i at the optimal allocation a^*. The last summand, $\widetilde{S}(\lfloor a \rfloor_i^*)$, corresponds to the optimal virtual surplus achieved in a similar setting but without the seller s_i being present. Thus, the argument of the inverse virtual cost function can be interpreted as a *virtual externality* imposed by the seller s_i.

Observe also that in the general case, the objective of maximizing the virtual surplus in the allocation rule of the BORA auction is non-linear. This follows from the fact that the induced values of databases need not depend linearly on different allocations. This poses a number of computational challenges that we will address in Section 3.5. Appendix A presents a number of worked examples that illustrate the BORA auction.

Theorem 3.3. *If the distributions f_i are regular for all $i = 1, ..., N$, and the databases are complementary for the aggregate buyer, then the BORA auction maximizes buyers utilities and satisfies constraints* (13)*,* (14)*, BNIC and IR.*

Proof. First, let $\mathbb{E}_{f_{-i}}[g_i(c_i, c_{-i})]$ be an *ex-interim* allocation probability of s_i. As Myerson (1981) showed, BNIC, IR and constraints (13) and (14) imply monotonicity of ex-interim allocation (see Lemma B.1 in Appendix B). We use this result to prove the following lemma:

Lemma 3.4. *Consider the allocation rule $g : \mathbb{R}^N_{\geq 0} \to [0, 1]^N$ that maximizes*

$$\mathbb{E}_f\Big[\sum_{k=1}^{D}\Big(\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)}\Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i(c)\Big)\Big] \tag{16}$$

subject to monotonicity of the ex-interim allocation and constraints (13) *and* (14)*. Further, consider the payment rule $h_i(c) = g_i(c)c_i + \int_{c_i}^{\beta_i} g_i(\hat{c}_i, c_{-i})d\hat{c}_i$ for every $i = 1, ..., N$, $\forall c$. Then, $\mathcal{A} = \langle g, h\rangle$ maximizes buyers' utilities under the constraints* (13) *and* (14)*, BNIC and IR.*

Proof. The proof is presented in Appendix B. □

Now we would like to show that if a database ℓ is allocated with a positive probability, then this probability must be equal to 1. To achieve this, we first present the following proposition that shows that constraints (13) are binding when databases are complements:

Lemma 3.5. *Let $g^*(c)$ be a solution of* (16)*. Then, if for a database ℓ there exists a seller s_i with $i \in i(\ell)$ such that $g_i^*(c) > 0$, then $\sum_{i\in i(\ell)} g_i^*(c) = 1$.*

Proof. See Appendix B for the proof. □

From Lemma 3.5 and Lemma B.2 (see Appendix B), it follows that, there must exist a deterministic allocation $g^*(c)$ that maximizes Equation (16). Consider a mechanism that for any reported cost profile c maximizes the virtual surplus $\widetilde{S}(a) = \sum_{k=1}^{D} W_k(a, p) - \sum_{k=1}^{D}\sum_{i\in i(k)} \phi_i(c_i)a_i$, where $a \in \{0, 1\}^N$. This allocation also maximizes Equation (16). Remember that the distributions f_i are regular.[11] Thus, $\phi_i(c_i)$ must be monotone. Consequently, the ex-interim allocation is also monotone.

11. For irregular distributions we could use *ironing* in a similar way as in (Myerson, 1981).

From Lemma 3.4 it follows that in order guarantee BNIC, the payments of sellers must satisfy

$$h_i(c) = g_i^*(c)c_i + \int_{c_i}^{\beta_i} g_i^*(\hat{c}_i, c_{-i})d\hat{c}_i \tag{17}$$

for every $i = 1, ..., N$, $\forall c$. If a seller s_j is not allocated (i.e., $a_j = 0$), then from monotonicity of the ex-interim allocation (see Lemma B.1) it follows that s_j is not allocated for any other cost $q_j \geq c_j$. Consequently, $h_j(c) = 0$. If a seller s_j is allocated (i.e., $a_j = 1$), then Equation (17) can be simplified as follows

$$h_j(c) = c_j + \int_{c_j}^{\zeta_j} d\hat{c}_j = c_j + \zeta_j - c_j = \zeta_j, \tag{18}$$

where

$$\zeta_j = \sup\{c | \phi_j(c) \leq \phi_i(c_i) \quad \forall i \in i(k) \backslash j \ \texttt{and} \ \phi_j(c) \leq \phi_j(c_j) + \widetilde{S}(a^*) - \widetilde{S}(\lfloor a \rfloor_j^*)\}. \tag{19}$$

Now, let us exclude the seller s_j who produces a database k from the mechanism and consider two scenarios. In the first scenario, the resulting allocation of databases stays the same, i.e., the database k is still produced but perhaps by a different seller $i \in i(k)$. In this case, it must hold that $\phi_i(c_i) = \min\{\phi_q(c_q), q \in i(k)\}$. Thus, the second inequality in (19) implies the first one. In the second scenario, the database k is not allocated anymore. Thus, it must be that $\phi_i(c_i) - \phi_j(c_j) \geq \tilde{S}(a^*) - \tilde{S}(\lfloor a \rfloor_j^*)$, where $\phi_i(c_i) = \min\{\phi_q(c_q), q \in i(k)\}$. Equivalently, $\phi_i(c_i) \geq \phi_j(c_j) + \tilde{S}(a^*) - \tilde{S}(\lfloor a \rfloor_j^*)$ and therefore, the first inequality in (19) is again implied by the second one. Therefore, from the monotonicity of $\phi_j(c_j)$ it follows that the payment of any seller s_j can now be rewritten as follows:

$$h_j(c) = \phi_j^{-1}\Big(\phi_j(c_j) + \widetilde{S}(a^*) - \widetilde{S}(\lfloor a \rfloor_j^*)\Big). \tag{20}$$

□

3.4 The Overall Market Mechanism

The reverse auction designed in the previous section does not guarantee that the market mechanism is budget balanced. Instead, it assumes that the market platform can always pay the sellers. To guarantee that the market is budget balanced, the market mechanism needs to set a posted price p, such that the total amount of money collected from the buyers, $\sum_{j=1}^{L}(e - m_j^*(p, a))$, is equal to the total payment $\sum_{i=1}^{N} t_i$ received by the sellers.

Consider the *budget surplus*[12] achieved when the allocation is a and the price is p:

$$B(p, a) = \sum_{j=1}^{L}(e - m_j^*(p, a)) - \sum_{i=1}^{N} t_i. \tag{21}$$

12. In microeconomic literature, this quantity is also often called the *excess demand* for money (Mas-Colell et al., 1995).

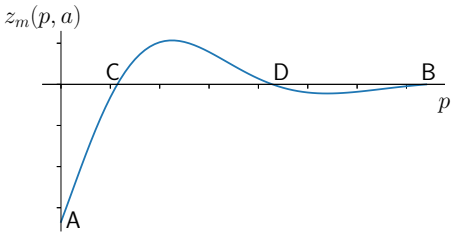

Figure 4: Example of the dependency of the budget surplus $B(p,a)$ on posted price p. Point C corresponds to the minimal p that satisfies overall budget balance.

Observe that the total payment to be accrued to sellers depends on allocation a and on price p. To see this, notice that if $p = 0$, buyers can gain a lot of value from submitting queries against all databases (for free). Thus, the induced value of any database k, $W_k(a,p)$, is large, and it is likely that the database is allocated. As a result, the second term in (21) is positive. At the same time, buyers do not pay anything and, consequently, the first term in Equation (21) is zero. This means that if $p = 0$, then $B(p,a) < 0$. This case is illustrated by point A in Figure 4. A similar argument works for a situation in which $p = \infty$. In this case, both terms in $B(p,a)$ are zero, which corresponds to the trivial equilibrium when no sellers are allocated. Point B in Figure 4 illustrates this scenario. As can be seen in Figure 4, the non-trivial equilibrium prices that satisfy the budget balance constraint correspond to points C and D. In general, there may be multiple solutions for which $B(p,a) = 0$ (such as points B, C and D in Figure 4). Consequently, there may be many different posted prices that guarantee budget balance. However, we aim to find the smallest such price, as it would deliver the largest total utility to buyers.

Data: $F_i(c_i)$, $i = 1, ..., N$; $F_{v'(a)}$, $F_{\overline{r}(a)}$ for all $a \in \{0,1\}^N$
Result: Allocation a, payments t, posted price p

$\delta \leftarrow 0.01$ // Step size
$\iota \leftarrow 0$
$p(\iota) = 0$
$a(\iota) \leftarrow (a_1, ..., a_N)$, s.t., $\forall k \leq D$ holds $\sum_{i \in i(k)} a_i = 1$
ask sellers to report $\hat{c} = (\hat{c}_1, ..., \hat{c}_N)$
repeat
  compute $W_k(a(\iota), p(\iota))$ for all $k \leq D$ // See Definition 6
  set up $\mathcal{A} = \langle g, h \rangle$ parametrized by $W_k(a(\iota), p(\iota))$ // See BORA
  solve $\mathcal{A}$, i.e., compute $a \leftarrow g(\hat{c})$ and $t \leftarrow h(\hat{c})$ // See BORA
  compute market demand for money $m_j^*(p(\iota), a(\iota))$
  $B(p(\iota), a(\iota)) \leftarrow \sum_{j=1}^{L} \left(e - m_j^*(p(\iota), a(\iota))\right) - \sum_{i=1}^{N} t_i$
  $p(\iota + 1) \leftarrow p(\iota) - B(p(\iota), a(\iota)) \cdot \delta$ // Price update
  $\iota \leftarrow \iota + 1$
until $|B(p(\iota), a(\iota))| \leq \epsilon$;
$p \leftarrow p(\iota)$
return a, t, p

Algorithm 1: Fixed-point iteration for computation of the allocation and the price.

To find a solution, we adopt an idea similar to the Tatonnement process (Cheng & Wellman, 1998). More concretely, we design an iterative algorithm that updates the price, allocation and payments of sellers at every iteration ι. We begin from an initial price $p_0 = 0$ that corresponds to a situation when rows of database tables corresponding to buyers' queries are free. We then perform a fixed-point iteration by increasing the posted price $p(\iota)$ as a function of the iteration ι, as well as adjusting allocation probabilities $a(\iota)$. At every iteration ι, we evaluate $W_k(a(\iota), p(\iota))$ and compute the tentative allocation a and payments t of the auction $\mathcal{A}$. Knowing the allocation and payments, we can compute the excess demand for money, $B(p(\iota), a(\iota))$, at this iteration. As long as $B(p(\iota), a(\iota))$ is negative, we increase the price $p(\iota)$. We stop the algorithm when the change in price is smaller than the chosen tolerance threshold. Algorithm 1 summarizes the whole market mechanism.

Remember that the sellers are asked to report their costs only once. Precisely, this happens at time $\tau = 1$ of our temporal model (see Figure 3). We assume that the sellers understand Algorithm 1 and the rules of the BORA auction. Thus, they can make a truthful report $\hat{c} = c$ (see line 5 of Algorithm 1).

Note also that Algorithm 1 does not require any interaction with the buyers. Instead, it is considered a heuristic procedure for computing an equilibrium price and allocation. Thus, this algorithm must be executed at time period $\tau = 2$ of our temporal model (see Figure 3), in other words, *before* actual buyers arrive to the market. This implies that the iterative nature of the algorithm does not change the incentives of the buyers to behave truthfully and the truthfulness of the overall market follows immediately from the truthfulness of the BORA auction.

Second, observe that a non-trivial equilibrium does not always exist. Consequently, our algorithm may return a null allocation and zero payments. Consider, for example, about a domain with a single seller with a high fixed cost and assume that there is a single buyer with a very small marginal value and a small value threshold. In this case, it is not possible to compensate the seller for producing her database. This result, however, does not constitute a failure of our market design: Indeed, if the society does not value the data highly enough, then the data should not be produced in the first place. In other words, we are aiming at designing a market that incentivizes data providers to produce *useful* data rather than *any* data. [13]

Finally, note that, even though we designed our market with the goal of optimizing the buyers' surplus, it is not possible to provide any meaningful lower bound on the share of the surplus obtained by buyers in general. The following proposition states this result formally:

Proposition 3.6. *The share of the buyers' surplus achieved by Algorithm 1 auction is lower bounded by zero.*

Proof. To prove the statement we construct a corner case where all sellers have zero costs and face no competition for producing their databases. At the same time, only joining *all* databases brings value to every buyer, while joining any other combination of databases

13. Remember, that our choice of the initial price $p_0 = 0$ follows the idea that we want to find an equilibrium with the largest surplus for buyers. Clearly, if the initial price was too high, then the trivial equilibrium in which nobody is allocated could be reached immediately. Technically, we could also launch Algorithm 1 from several starting points. However, our experiments show that even starting with $p = 0$, we can obtain high levels of surplus. Example 5 in Appendix A illustrates our approach.

has zero value. In this case, we can show that the buyers' surplus is zero (and it can't be negative as buyers can always decide not to participate in the trade). The full proof is provided in Appendix B. □

The corner case used to prove Proposition 3.6 is obviously pathological, and we would not expect such cases in practice. To study how much surplus buyers can get in more realistic settings, we have performed a number of computational experiments (see Section 4).

3.5 Winner Determination via Mixed-Integer Allocation Programming

In this section, we discuss computational challenges that arise in practical implementation of our proposed BORA auction. First, remember that buyers are indifferent about the identities of sellers. Thus, it follows that the induced values of databases $W_k(a,p)$ are constant for any allocation of sellers as long as the allocation of the respective databases stay the same. As the number of databases D is typically much lower than the number of sellers N, we can now compute the induced values of databases for every possible deterministic allocation $\alpha \in \{0,1\}^D$ of databases rather than for every possible deterministic allocation $a \in \{0,1\}^N$ of sellers. With a slight abuse of notation we let $W_k(\alpha,p)$ be the induced value of the database k when the *allocation of databases* is α.

Further, remember that the winner determination problem in the BORA auction is non-linear. In order to linearize it, we can pre-compute the induced values of databases for every possible deterministic allocation of databases. We then include these pre-computed values into the objective function with auxiliary binary optimization variables indicating whether a particular deterministic allocation of databases is chosen. This idea is illustrated with the following linearized mixed integer program:

$$\max_{\substack{a_i, i=1,\dots,N \\ z_\alpha, \alpha \in \{0,1\}^D}} \sum_{\alpha \in \{0,1\}^D} \left[z_\alpha \cdot \sum_{k=1}^{D} W_k(\alpha,p) \right] - \sum_{i=1}^{N} \phi_i(c_i) a_i \tag{22}$$

$$\texttt{s.t.} \quad z_\alpha \leq \sum_{j \in i(k)} a_j \quad \forall \alpha \in \{0,1\}^D \ \ \forall k = 1,\dots,D \ \texttt{s.t.} \quad \alpha_k = 1 \tag{23}$$

$$z_\alpha \leq \sum_{j \in i(k)} (1 - a_j) \quad \forall \alpha \in \{0,1\}^D \ \ \forall k = 1,\dots,D \ \texttt{s.t.} \quad \alpha_k = 0 \tag{24}$$

$$a_i \in \{0,1\} \quad \forall i = 1,\dots,N \tag{25}$$

$$z_\alpha \in \{0,1\} \quad \forall \alpha \in \{0,1\}^D. \tag{26}$$

Here, binary optimization variables a_i represent allocation decisions of the BORA auction regarding sellers s_i, $i = 1,\dots,N$. Further, the synthetic optimization variable z_α is equal to 1 if the deterministic allocation of databases is $\alpha \in \{0,1\}^D$. Here, constraints (23) and (24) build a bridge between allocation decisions regarding different sellers and the chosen deterministic allocation of databases produced by these sellers. In particular, constraints (23) guarantee that the deterministic allocation α in which the database k is allocated ($\alpha_k = 1$) is not feasible ($z_\alpha = 0$) if none of the sellers producing database k are allocated. Similarly, the constraint (24) sets $z_\alpha = 0$ if at least one seller producing the database k is allocated even though the database k should not be allocated ($\alpha_k = 0$).

Looking into the objective function (22), we see that constraints (23) and (24) guarantee that there is only a single term of the total induced value of databases that gets activated for a specific allocation of databases and sellers. In particular, if the allocation of databases is α and the allocation of sellers a satisfies constraints (23) and (24), then the objective function value is $\sum_{k=1}^{D} W_k(\alpha, p) - \sum_{i=1}^{N} \phi_i(c_i)a_i$.

Observe, that such a linearization of the allocation problem of the BORA auction is achieved by a high cost. Indeed, in order to achieve the linear formulation, we have introduced a number of synthetic optimization variables z_α, that grows exponentially in the number of databases D. However, the high complexity of our approach seems to be unavoidable and follows directly from the combinatorial preferences of buyers for different allocations of databases. Example 4 in Appendix A illustrates the advantages of using the combinatorial design of the auction to achieve higher buyers' surplus.

4. Experiments

To study the economic properties of our market, we carry out a set of computational experiments. To do this, we first implement a simulation set up to generate buyers and sellers according to the model described in Section 2.1. We then run our market mechanism based on the Algorithm 1 and measure social welfare, as well as the share of the social welfare obtained by buyers and sellers, respectively. Note that, at first, we perform all of our experiments in a small "stylized" setting to gain a detailed understanding of the behavior of our market mechanism. To study the scalability of our approach, we then perform a number of computational experiments in a medium-sized domain, in which we increase the numbers of databases, data providers and buyers. More realistic large-scale market simulations may require a thorough examination of buyers' preferences for data of a particular domain (e.g., preferences of doctors or drug developers using life sciences databases (Hall et al., 2013) or of marketers in the digital marketing domain (Blue Kai, Inc., 2011)). We defer these simulations to future work.

4.1 Experiment Set-up

Small Set-up. We simulate a setting with $N = 3$ sellers and $D = 2$ databases. We assume that seller s_1 can produce database 1, and both sellers s_2 and s_3 can produce database 2. We assume that costs c_i of all sellers are i.i.d., $c_i \sim U[0, 20]$, $i = 1, 2, 3$. This models a scenario in which seller s_1 is a unique producer of database 1 (i.e., s_1 has a monopolistic ownership of her data), while sellers s_2 and s_3 are competing to produce database 2. Because the BORA auction is BNIC (Theorem 3.3) we assume that sellers report their costs truthfully.

We vary the number of buyers L from 1 to 128 while keeping the number of sellers and databases fixed. Each buyer has an initial endowment of money $e = 10$. For simplicity, we assume that the marginal value v'_j and the threshold $\overline{r}_j$ of any buyer b_j only depends on the number of allocated databases but not on the identities of these databases. Thus, if there are no databases allocated, the marginal value and the threshold of each buyer are equal to zero. If there is exactly one allocated database, then the marginal value of each buyer is drawn from $U[0, 2]$. The threshold, in this case, is drawn from a discrete uniform distribution $U\{0, 5\}$. Finally, if there are exactly two databases allocated then the marginal

value v_j is incremented by a random variable drawn from $U[0, 2]$, while $\overline{r}_j$ is incremented by a random variable drawn from $U\{0, 5\}$.

Medium Set-up. To study the scalability of our approach, we perform a number of computational experiments with more realistic numbers of data providers, databases and buyers. Generally speaking, the exact numbers can be domain specific. In practice, we expect them to be in the order of dozens to hundreds for data providers and thousands for buyers. Some evidence for these numbers is derived from an examination of existing data marketplaces. For example, a recent report on the Oracle proprietary digital data marketplace for selling marketing data (Blue Kai, Inc., 2011) presents a domain with more than 200 data providers and with more than 200 customers across multiple industries. Assuming that a typical query against real-world databases joins only two to four of these databases, we can split buyers into different groups based on the databases they are most often interested in. In this case, buyers within a group are assumed to submit most of their queries against 5 to 15 databases and very few queries against the remaining databases. In other words, we assume that the overall market can be partitioned into several pieces with loose pairwise connections between these pieces. Such partitioning reduces the number of databases and buyers in each part, which therefore leads to a lower computational hurdle. Therefore, for our experiments, we assume that such a partitioning can be carried out effectively using a certain clustering technique, and we can run our market mechanism for each piece of the partitioning. The design of the exact clustering procedure is beyond the scope of this work.

Thus, in our medium-sized experiments, we vary the number of databases D, from 2 to 10. While some databases can be produced by unique data providers (monopolistic data ownership), others can be produced by many different data providers. This allows us to vary the number of data providers N from 2 to 100. Similarly to the small set-up, we generates sellers' costs from a uniform distribution $U[0, 20]$.

In this set-up, we vary the number of buyers L from 8 to 1024. We use the same simple value model for buyers as in the small set-up. Specifically, we assume that allocation of an additional database has two effects on buyers. First, it leads to a growth of the marginal value v_j of a buyer by a random amount drawn from $U[0, 2]$. Second, it causes an increase in the threshold $\overline{r}_j$ by a random number drawn from $U\{0, 5\}$. The main complication of the medium-sized set-up compared to the small set-up is of a computational nature: Due to the combinatorial structure of the problem, we now must specify the value function of every buyer for all 2^D possible deterministic allocations of databases. As previously discussed, in practice, many buyers submit their queries against only a certain subset of available databases. This means that the marginal change of a buyer's value caused by allocation of an additional database gets smaller as the number of allocated databases grows. As a result, the computational problem of estimating the typical value function of a buyer may be somewhat *simpler* than in our experiments. However, modeling more realistic value functions of buyers can be domain specific as it depends on the typical queries that buyers submit. Modeling such realistic value functions is one of the directions of our future work.

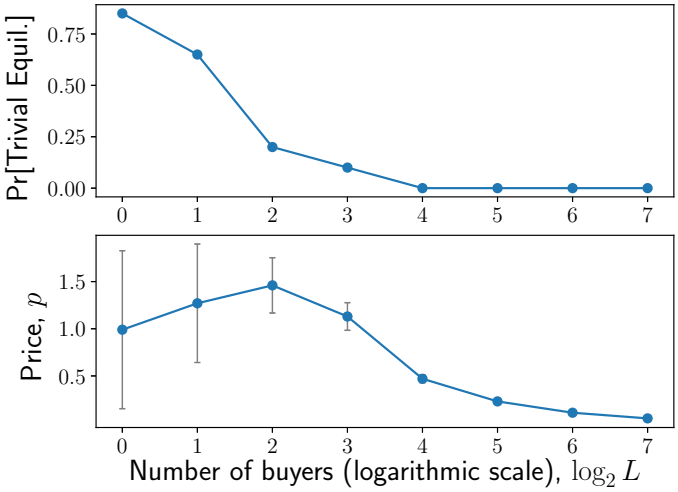

Figure 5: The top graph shows the probability that the market is in a trivial equilibrium. We vary the number of buyers from 1 to 128 (or from 0 to 7 in the logarithmic scale). The bottom graph shows the dependency of price p on the number of buyers, L. Error bars indicate confidence intervals at 0.05 significance level.

4.2 Studying the Small Set-up.

We call the auction instances with the same number of buyers a *setting*. For each setting (with 1, 2, 4, ... buyers) we generate 10 random instances as described above. This allows us to estimate the mean values and confidence intervals for every setting.

4.2.1 Probability of "No Trade"

Consider Figure 5 (top), which shows the probability that the market mechanism only finds the *trivial equilibrium* ("no trade"). We see that, when L is small (≤ 8), then it is likely that there are not enough buyers to cover the fixed costs of sellers, whatever the price p. If $L \geq 8$, then both databases are always allocated. In this case, seller s_1 produces database 1 and either s_2 or s_3 produces database 2.

4.2.2 Posted Price

Consider Figure 5 (bottom), which shows the dependency of the posted price (in a non-trivial equilibrium) on the number of buyers L. As L grows, the amount of money that must be collected from each buyer to achieve BB decreases. Consequently, the price p also decreases. As the number of buyers increases, the marginal effect of every additional buyer on the price decreases (the price curve becomes less steep). This is expected, as the impact of an individual buyer on the aggregate demand decreases as L increases, which leads to more and more"price-taking behavior".

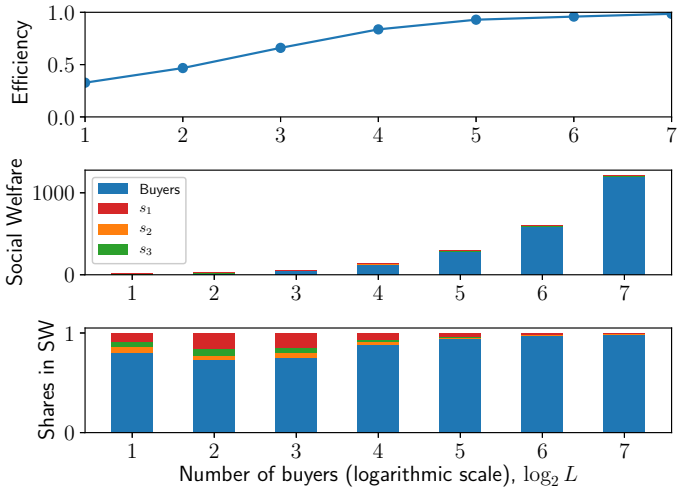

Figure 6: The top graph shows efficiency reached in a non-trivial equilibrium. The middle graph shows social welfare separated by buyers and allocated sellers. The bottom graph shows the relative distribution of the achieved welfare. Small differences in the mean surplus of sellers s_2 and s_3 are not statistically significant at the 0.05 significance level.

4.2.3 Efficiency and Social Welfare

We now study the most important question: How *efficient* is our market mechanism? To this end, consider Figure 6. In the graph in the center, we observe that the absolute value of social welfare grows linearly as the number of buyers increases (an exponential trend in the logarithmic scale). This is what we would expect. Now, consider Figure 6 (top), which shows the efficiency of our market mechanism.[14] For the illustration reason we omit the efficiency measurement that corresponds to $L = 1$. The reason for that is that in this case our market reaches almost 100% efficiency. This is due to the fact that when the number of buyers is small, the market stays in the "no trade" equilibrium most of the time (see Figure 5 (top)). Even when there is trade, only one database gets allocated, which makes achieving high efficiency easier. If both databases are allocated ($L > 32$), the efficiency is 95%. As L increases further, the efficiency stabilizes. This is due to the fact that the posted price p becomes more flat (see Figure 5 (bottom)), such that there is a constant fraction of buyers with a marginal value per row smaller than the posted price p. These buyers do not buy anything, causing the efficiency loss. Thus, as p becomes constant, the efficiency also becomes nearly constant.

4.2.4 Shares of Social Welfare

Consider Figure 6 (bottom), which shows how social welfare is distributed between buyers and sellers. Observe that seller s_1's share increases as the number of buyers increases (as long as $L \leq 8$), while shares of other sellers decrease (see also Table 1 for the absolute values and standard errors). To understand this result, remember that seller s_1 is a monopolist, i.e., she faces no competition for her database. Thus, her payment is solely determined by

14. Efficiency is defined in the standard way, as the fraction of the social welfare achieved by our mechanism and the social welfare of an *optimal* (omniscient) mechanism which can disregard incentive constraints.

Surplus	Number of buyers						
	2	4	8	16	32	64	128
Buyers	7.20 (1.28)	17.66 (2.25)	45.00 (6.17)	128.02 (8.23)	280.82 (6.18)	583 (7.97)	1198.23 (15.65)
Seller s_1	1.17 (0.62)	4.39 (1.46)	8.61 (1.40)	9.57 (1.26)	9.57 (1.26)	9.57 (1.26)	9.57 (1.26)
Seller s_2	0.54 (0.35)	0.80 (0.62)	3.40 (1.16)	4.06 (1.33)	4.06 (1.33)	4.06 (1.33)	4.06 (1.33)
Seller s_3	0.59 (0.41)	1.74 (0.67)	3.26 (0.87)	3.43 (0.97)	3.43 (0.97)	3.43 (0.97)	3.43 (0.97)

Table 1: Mean values and standard errors for the surplus of buyers and sellers for the small setup reached in a non-trivial equilibrium.

the reserve price set by the auction, which only depends on the value that database 1 is expected to generate for *all buyers*. As L increases, this value naturally increases, which means that the seller receives a larger payment. However, this payment is bounded by the upper bound of the support of the distribution of sellers' costs (which is 20 in this set up). This is why as soon as $L \geq 16$, the share of s_1 can only decrease. In contrast, s_3's payment is constant for $L \geq 20$ and depends only on the virtual cost of her competitor s_2.[15] Finally, there is also no significant evidence that sellers s_2 and s_3 get different surplus in the market at 0.05 significance level.

4.3 Studying the Medium Set-up.

4.3.1 Aggregate Demand

To illustrate combinatorial preferences of buyers, we compute aggregate demand curves that correspond to different deterministic allocations of databases. Remember that the value model we adopted for our simulations assumes, for simplicity, that marginal values v'_j and thresholds $\overline{r}_j$ of buyers depend only on the number of allocated databases but not on identities of those databases. Thus, we can restrict our attention to considering only 10 different deterministic allocations that correspond to different numbers of allocated databases.

Figure 7 illustrates how the demand curve changes as the number of allocated databases increases. There are two effects happening in parallel. First, as the number of allocated databases grows, the threshold $\overline{r}_j$ also increases (see Section 4.1). This shifts the demand curve to the right for larger numbers of allocated databases.

Second, as the number of allocated databases grows, the marginal values of buyers also increase (see Assumption 1). As a result, for large price levels, the elasticity of demand increases, as now there are more buyers reacting to changes in the posted price. To see this more clearly, compare the leftmost curve that corresponds to the case with a single allocated database to the second curve that corresponds to the case with two allocated databases.

15. When $L \leq 8$ then the reserve payment is smaller than the virtual cost of s_2. Then s_3's payment sometimes depends on the reserve payment.

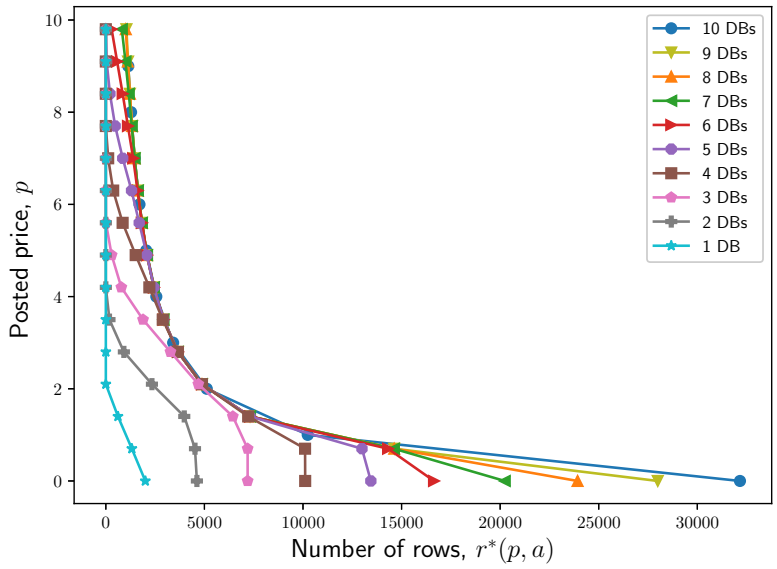

Figure 7: Aggregate demand curves in the domain with 1024 buyers and 10 databases. Different curves correspond to different numbers of allocated databases. As the number of allocated databases increases, the respective demand curve shifts to the right. The difference between two demand curves gets smaller as the number of allocated databases grows.

Clearly, if the price $p > 2$, no rows are consumed in the first scenario. This happens because marginal values of all buyers are smaller than this price (remember that marginal values in this case are drawn from $U[0, 2]$). By contrast, the marginal values of buyers in case of two allocated databases are drawn from a distribution with a larger support,i.e., interval $[0, 4]$. Thus, more buyers are now reacting to price changes at the price level $p = 2$. Consequently, for large price levels, the aggregate demand becomes more elastic as the number of allocated databases increase.

Remember that as the number of allocated databases increases, there are fewer buyers with very small marginal values. Therefore, for small price levels (e.g., $p \approx 0.5$), there are more buyers with marginal values larger than p. These buyers keep buying even if the price changes slightly. This is why, for small price levels, the aggregate demand becomes less elastic as the number of allocated databases increase.

In Figure 7, we also see that the difference between demand curves gets smaller as the market grows. In practice, such "convergence" of demand curves may allow us to reduce the computational hurdle arising from combinatorial preferences of buyers by considering smaller domains. We conjecture that, in this case, we could bound the efficiency loss caused by such an approximation. However, we leave this direction to future work.

4.3.2 Expected Profit

Now, we study how the expected profits of sellers responds to the level of competition between sellers for producing their databases. To do this, we fix the number of databases $D = 10$ and the number of buyers $L = 1024$. We further assume that database 1 can

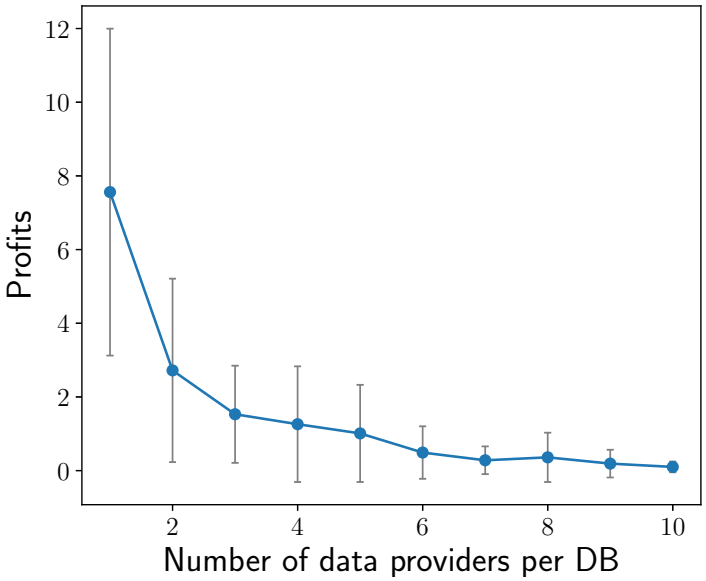

Figure 8: Expected profits of sellers in a market with 10 DBs and 1024 buyers. All sellers extract a positive expected surplus. The larger the number of sellers involved in the production of a DB, the smaller the surplus these sellers can expect.

be produced only by a single data provider s_1, who has a monopolistic ownership for the respective data. Similarly, database $k \leq D$ can be produced by k different data providers who compete with each other in the BORA auction to produce the database. Thus, the total number of data providers in this scenario is

$$\sum_{k=1}^{D} k = 1 + 2 + ... + 10 = 55.$$

Figure 8 demonstrates that the seller s_1 enjoys the largest expected surplus. To explain this result, remember that s_1 faces no competition in the BORA auction, as she is a unique data provider for database 1. This means that the externality imposed by s_1 is potentially larger that the externality exposed by any other seller facing a stronger competition. Consequently, the payment must also be larger.

One insight provided by this simulation is that, despite that all allocated data providers can recoup their fixed costs, our market rewards data providers who *innovate* (produce original data) substantially more than those who produce databases containing *common knowledge*.

4.3.3 Shares of Social Welfare

Figure 9 (left) presents the distribution of the social welfare achieved in the market as the number of buyers increases from 16 to 1024. As in Section 4.3.2, we set the number of databases $D = 10$ and we assume that database 1 can be produced by only a single data provider s_1, while database k can be produced by k different data providers.

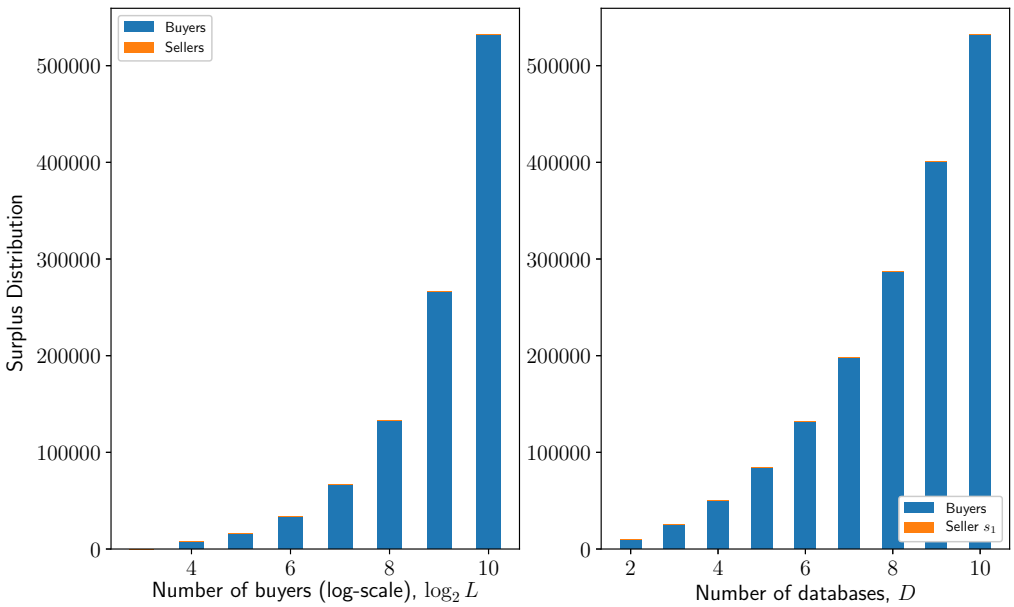

Figure 9: Distribution of the total welfare achieved in the market with 1024 buyers and different numbers of DBs. Buyers get the largest share of it, while the seller that produces the unique DB gets the second largest portion.

The figure confirms that buyers receive the largest share of total welfare. It is clear as the BORA auction is designed to maximize the buyers' surplus. The figure also suggests that the surplus of sellers stays constant as the number of buyers increases. This result is expected: indeed, the payment of s_1 approaches its maximum value of $\phi^{-1}(20)$ (in this experiment, $c_1 \sim U[0, 20]$) as the number of buyers increases. At the same time, payments of other sellers are essentially *second price* payments. These payments depend on costs distributions F_i of sellers rather than on the number of buyers. Consequently, the total payment to be accrued to sellers does not depend much on the number of buyers. Finally, the total social welfare achieved in the market grows linearly as the number of buyers increases (exponential trend in the logarithmic scale).

Figure 9 (right) presents the distribution of social welfare achieved in the market as the number of databases increases from 2 to 10. Here, we set the number of buyers to 1024. As before, we can see that buyers always get the largest share in the overall surplus. The figure also demonstrates the exponential growth of the social welfare. This trend however, may be caused by the strong complementarity of all databases implied by our experimental setup. In practice, buyers may not join all possible databases and thus the trend can be less steep.

4.3.4 Varying the Level of Unique Data Providers

Next, we study how the extent of monopolistic ownership of data affects the social welfare and its distribution in the market. To do this, we consider a setting with 10 databases and 1024 buyers. We assume that k out of 10 databases are produced by a single data provider

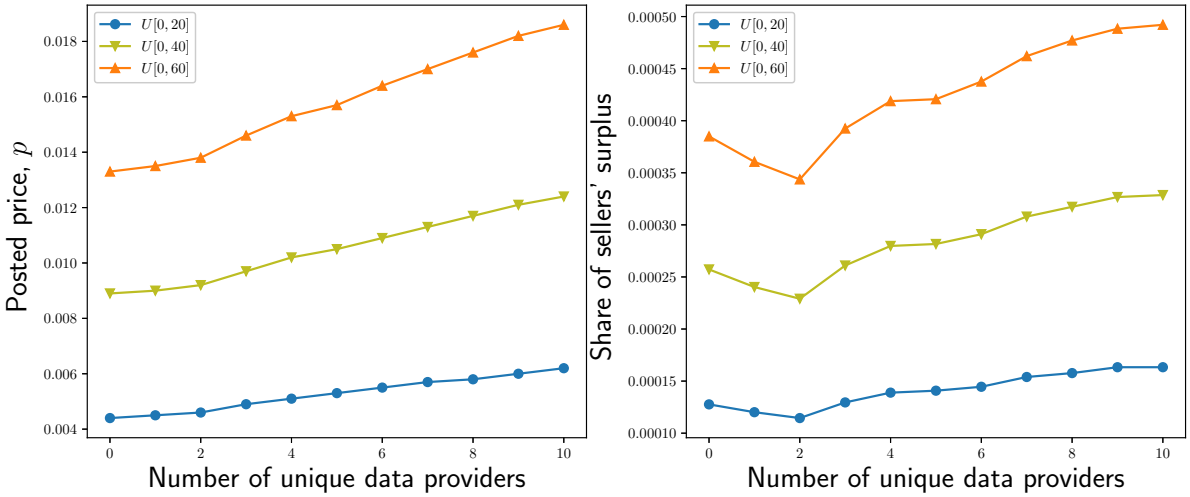

Figure 10: Dependency of the price (left) and share of sellers' surplus (right) on the number of sellers with monopolistic ownership of data. The experiment is carried out for different costs distributions of sellers, F_i.

and the rest $(10-k)$ can be produced by two data providers. In this case, we say that there are k data providers *with monopolistic ownership of data*. In our experiments, we vary k from 0 to 10. We argue that restricting the competition for the rest $(10-k)$ of databases to two sellers is enough. Indeed, the competition in the respective BORA auction drives profits of competing sellers down to a constant level (see Section 4.3.2). As the number of buyers increases, even the competition between only two sellers that produce the same database makes their profits small compared to the profit of the seller s_1 with monopolistic ownership of data (see Figure 8).

Figure 10 illustrates our findings. Here, the horizontal axis corresponds to the fraction of databases produced by a data provider with monopolistic ownership of her data, i.e., $\frac{k}{D}$, $k \leq D$. As Figure 10 (left) demonstrates, the posted price p grows linearly as the number of data providers with monopolistic ownership of data increases. This trend is clear, as the number of allocated databases increases, buyers' willingness to pay also grows (following from Assumptions 1 and 2). At the same time, the market platform now needs to collect more money from buyers to compensate the newly allocated data providers. Naturally, the market platform needs to increase the posted price to keep the overall market balanced.

We carry out this experiment for three different costs distributions. In the first case, costs of sellers c_i are drawn uniformly from $[0, 20]$ as before. In the second and the third cases, these costs are drawn from $U[0, 40]$ and $U[0, 60]$ respectively. This approach enables to model different levels of uncertainty of the market platform regarding costs of sellers. As Figure 10 (left) demonstrates, the higher this uncertainty, the larger is the posted price the market platform needs to set. The explanation of this follows form the fact that higher level of uncertainty enables sellers to receive larger payments in the BORA auction. Consequently, the market platform must set higher prices to make the market budget balanced. As a result, the profits of sellers must also grow (see Figure 10 (right)).

4.4 Discussion

The results of our computational experiments raise a number of interesting points regarding designing a market for distributed data.

First, the design we proposed solves the original problem we posed. Specifically, the market brings high surpluses to buyers and compensates allocated sellers. The market enables allocated data providers to recoup the high fixed costs they experienced when producing their databases and linking them against databases of other data providers. As we showed, such a market gives stronger incentives to data providers to innovate, i.e., to produce unique data sets instead of transforming common knowledge into a structured form. The rising inequality in profits distribution arises from the fact that the production of a unique database reduces competition in the BORA auction. This allows data providers with monopolistic ownership of data to receive payments that are typically larger than the "second price" payment. However, even those allocated data providers who do not have access to the unique data can expect to run positive profits.

Second, our market awards a large portion of the achieved welfare to buyers. In fact, even in the most extreme cases, when all data providers enjoy high profits from monopolistic ownership over their data, buyers can still expect to get a substantial portion of the achieved welfare. In practice, providing a high share of surplus to buyers can be crucial when designing such a market. It allows for a shift in the current paradigm meaning that the linked data must be free, and thus makes buyers less resistant to entering the market (some discussion on this topic can be found, for example in Grubenmann et al. (2018), Grubenmann et al. (2017)).

Finally, our experiments give us a reason to think that the complex combinatorial preferences of buyers do not constitute an unsolvable issue. In fact, we think that the combinatorial structure can be tackled efficiently when we aggregate buyers and setup the posted price based on their aggregated preferences. In particular, we showed that as the number of databases grows, the aggregate demand curve does not change significantly. This opens up an opportunity to approximate the aggregate demand curve for a large domain by considering only a smaller part of it. We conjecture that this can reduce the complexity of our approach with a bounded loss in efficiency. However, we leave this direction to future work.

Limitations. In our computational experiments, we considered buyers coming from the same value model. This means that, despite the fact that different buyers in our experiments have different marginal values v_j and thresholds $\overline{r}_j$, these values are still drawn from the same distribution. We also assumed that all buyers are endowed with the same initial amount of money and that the preferences of all buyers depend on the number of allocated databases, not on the identities of these databases. All these simplifying assumptions were made for better clarity of the experimental results, rather than to circumvent any complications arising in computations of the equilibrium allocation and prices. Thus, any of these assumptions can be easily relaxed.

5. Conclusion

In this paper, we have proposed a combinatorial market for distributed data. Our research is motivated by the increasing value of data, while the design of good mechanisms for buying

and selling data has proved to be elusive. We have argued that data is different from other information goods such as music or videos because databases produced by different data providers can be joined and buyers have combinatorial values over which databases are available. The key idea behind our solution is to use two different mechanisms for the two sides of the market and to employ a fixed-point iteration algorithm for finding an outcome that balances the entire market. Our experimental results are consistent with our theoretical predictions, with a small number of buyers, it is likely that no trade happens because the buyers' values are not large enough to warrant the high fixed costs of the data providers. But, as more and more buyers arrive on the market, the probability of trade approaches one, and the posted price buyers face quickly stabilizes. We have also shown that, as the number of allocated databases increases, the marginal change in the aggregate demand gets smaller. This opens the door for future opportunities to design an approximation algorithm that would efficiently tackle the computational hardness of equilibrium price computation procedure for larger domains. Another important discovery of our model highlights the fact that data providers who innovate by producing unique data sets can expect to receive larger rewards than those who structure the common knowledge data. However, even in the most extreme cases, where every data provider enjoys monopolistic ownership of her data, buyers can still expect to receive at least half of the total welfare generated by the trade. Future work can build on our model and consider various extensions, such as the dynamic arrival of sellers to the market or endogenous demand (i.e., where the number of buyers varies depending on the price). One particularly important subject of future work is the development of a realistic domain generator and large-scale simulations to study the behavior of our market mechanism under real-world conditions.

Appendix A. Examples

Example 3. *We use the setting of Example 2 but assume now that both sellers s_1 and s_2 are allocated, i.e., $a'' = (1,1)$. As there is now more data available to buyers, their preferences change. Assume that value functions of buyers for the new allocation are $v_1(r_1, a'') = 6\min\{r_1, 1\}$ and $v_2(r_2, a'') = \min\{r_2, 4\}$. Now, the aggregate buyer has the following aggregate value $V(r, a'')$ and demand $r^*(p, a'')$:*

$$V(r,a'') = \begin{cases} 6r, & \text{if } r \in [0,1] \\ 5+r, & \text{if } r \in [1,5] \\ 10, & \text{if } 5 \le r \end{cases} \quad r^*(p,a'') = \begin{cases} 5, & \text{if } p \in (0,1) \\ [1,5], & \text{if } p = 1 \\ 1, & \text{if } p \in (1,6) \\ [0,1], & \text{if } p = 6 \\ 0, & \text{if } 6 < p. \end{cases}$$

In this case, the positive externality imposed by s_2 is $ext_2(a'', p) = V(r^*(p, a''), a'') - V(r^*(p, a), a)$. Again, if $p = 1 - \epsilon$, then the market demand $r^*(p, a'') = 5$ and the aggregate value of having both databases is $V(r^*(p, a''), a'') = V(5, a'') = 10$. Then, the externality imposed by s_2 is $ext_2(a'', p) = V(r^*(p, a''), a'') - V(r^*(p, a), a) = 10 - 6 = 4$. If we now assume that for allocation $a''' = (0,1)$ agents have same preferences as for allocation $a = (1,0)$, then $ext_1(a'', p) = V(r^*(p, a''), a'') - V(r^*(p, a'''), a''') = 10 - 6 = 4$. Thus, the induced value of the databases are $W_1(p, a'') = \frac{4 \cdot 10}{8} = 5$, $W_2(p, a'') = \frac{4 \cdot 10}{8} = 5$. Observe, that the presence of s_2 increased the induced value $W_1(p, a'')$ for the database of s_1.

Example 4. *Consider a domain with a single buyer, $L = 1$. Assume that there are $N = 2$ sellers each producing a single database, i.e., $D = 2$. Let $c_1, c_2 \sim U[0, 2]$ and $c_1 = 1.5$, $c_2 = 0.5$. In this case, the virtual cost function for both sellers is $\phi(c) = c + \frac{F(c)}{f(c)} = 2c$; consequently, $\phi_1(c_1) = 3$ and $\phi_2(c_2) = 1$. Assume that the value function of the buyer is $v_1(r_1, a) = 5\min\{r_1, 1\}$ if both databases are allocated (i.e., $a = (1,1)$) and $v_1(r_1, a) = 0$ otherwise. The buyer has an endowment $e = 4$. As there is only a single buyer, the aggregate value function corresponds to the value function of this buyer, i.e., $V(r, a) = v_1(r, a)$. The endowment of the aggregate buyer is $E = e$.*

Let us now compute the induced values of both databases. First, $ext_1(a, p) = ext_2(a, p) = 5$ for all $p \le 5$ and for $a = (1,1)$. Also $ext_1(a, p) = ext_2(a, p) = 0$ if $a \ne (1,1)$ or if $p > 5$. Thus, $W_1(a, p) = W_2(a, p) = \frac{1}{2} \cdot 5 = 2.5$ for any $p \le 5$ if $a = (1,1)$ and $W_1(a, p) = W_2(a, p) = 0$ for other cases. Obviously, the solution to the allocation problem is $a^* = (a_1^*, a_2^*) = (1,1)$. In this case, the objective is $2.5 + 2.5 - 3 - 1 = 1$ for any $p \le 5$. The payments are computed as follows: $t_1 = \frac{1}{2}(3 + 1 - 0) = 2$, $t_2 = \frac{1}{2}(1 + 1 - 0) = 1$.

Observe that if we run instead two BORA auctions for each of the databases separately, we first would not allocate the first database as its virtual cost $\phi_1(c_1) = 3$ is larger than the induced value of the database $W_1(a, p) = 2.5$. Consequently, the second database would also not be allocated as the buyer has a positive value only for both databases.

Finally, if we set $p = 3$, then the buyer would decide to pay $e - m_1^*(p, a) = 3$ for a single row of answers for his query, $r_1^*(p, a) = 1$. Such a price makes the overall market balanced as the total payment to the sellers must be $t_1 + t_2 = 3$.

Example 5. *In this example, we would like to demonstrate that even if there exist multiple non-trivial equilibria, our mechanism finds the "best" one, i.e., the equilibrium with the largest surplus for buyers.*

Consider a domain with a single buyer, $L = 1$*. Assume that there are* $N = 2$ *sellers each producing a single database. Let* $c_1, c_2 \sim U[0,1]$ *and* $c_1 = c_2 = 0.5$*. In this case, the virtual cost functions are* $\phi_1(c_1) = \phi_2(c_2) = 1$*. Assume that the value function of the buyer is* $v_1(r_1, a) = 2\min\{r_1, 6\}$ *if both databases are allocated (i.e.,* $a = (1,1)$*) and* $v_1(r_1, a) = 2\min\{r_1, 2\}$ *if only one database is allocated. The buyer has an endowment* $e = 10$*. The aggregate value function is* $V(r, a) = v_1(r, a)$ *(the endowment of the aggregate buyer is* $E = e$*).*

Suppose Algorithm 1 starts with $p = 0$*. In this case, both databases must be allocated. However, the market is not budget balanced as the buyer pays 0. Assume now that after several iterations of Algorithm 1, the price increases to* $p = 1$*. In this case, an allocation* $a = (1,0)$ *makes the market budget balanced. Indeed, in this case,* $W_1(a, p) = 4$ *while* $W_2(a, p) = 0$*. Consequently, the virtual surplus is* $\tilde{S} = 4 - 1 = 3$ *and the payments are* $t_1 = 2$ *and* $t_2 = 0$*. Given this price, the buyer decides to buy two rows and thus pays the total amount of* 2 *which implies budget balancedness. In this equilibrium, the buyer gets a surplus of 2.*

Observe however, that the allocation $a = (1,0)$ *does not maximize the virtual surplus given the price* $p = 1$*. Instead, the BORA auction would allocate both databases. This would lead to a virtual surplus of* $\tilde{S} = 10$ *and payments* $t_1 = 4$*,* $t_2 = 4$*. This, makes the market unbalanced as buyers can pay only 6. Consequently, the price must increase up to* $p = 4/3$ *for the market to become budget balanced. In the new equilibrium, the buyer gets a surplus of 4 and pays the total amount of* $t_1 + t_2 = 8$ *to sellers.*

Appendix B. Proofs

Let $G_i(g, c_i) = \int_{c_{-i}} g_i(c_i, c_{-i}) f_{-i}(c_{-i}) dc_{-i}$ denote the ex-interim allocation of s_i.

Definition 7. *A mechanism $\mathcal{A} = \langle g, h \rangle$ is **feasible** if it satisfies BNIC, IR, and $\sum_{i \in i(k)} g_i(c) \leq 1$, $\forall k = 1, ..., D$, $g_i(c) \geq 0$, $\forall i = 1, ..., N$.*

Lemma B.1. *A mechanism $\mathcal{A} = \langle g, h \rangle$ is feasible if and only if the following conditions hold:*

1. *$c_i \leq q_i$ implies $G_i(g, q_i) \leq G_i(g, c_i)$ for any q_i, $\forall c_i \in [\alpha_i, \beta_i]$, $i = 1, ..., N$,*
2. *$\mathbb{E}_{f_{-i}}[u_i(g, c_i, h)] = \mathbb{E}_{f_{-i}}[u_i(g, \beta_i, h)] + \int_{c_i}^{\beta_i} G_i(g, q_i) dq_i$,*
3. *$\mathbb{E}_{f_{-i}}[u_i(g, \beta_i, h)] \geq 0$ for all $i = 1, ..., N$,*

and $\sum_{i \in i(k)} g_i(c) \leq 1$, $g_i(c) \geq 0$ for all $k = 1, ..., D$, $\forall c \in \prod_{i=1}^{N} [\alpha_i, \beta_i]$.

Proof. The proof repeats the respective proof provided in (Myerson, 1981) for a reverse auction setting.

□

In words, the first condition of the previous Lemma means *monotonicity of ex-interim allocation* while the second and the third conditions are more technical and will be used in derivation of the surplus optimal mechanism.

Now, remember that the probabilistic allocation of sellers $g(c)$ induces a probabilistic allocation of databases. We can define the expected induced value of a database k as follows:

Definition 8. *The **expected induced value** of the database k given the probabilistic allocation $g(c)$ of sellers is*

$$\mathbb{E}_{g(c)}\Big[W_k(a, p)\Big] = \sum_{a \in \{0,1\}^N} \prod_{i=1}^{N} g_i^{a_i}(c)(1 - g_i(c))^{1 - a_i} W_k(a, p).$$

In a setting when multiple sellers compete for producing a database ℓ, an assignment of the full allocation probability $\gamma \geq 0$ to only one of them leads to weakly higher expected induced values of all databases than any other assignment of γ. The following lemma shows this fact formally:

Lemma B.2. *Let $g(c)$ be a probabilistic allocation of sellers such that $\sum_{i \in i(\ell)} g_i(c) = \gamma$ for some $\ell \leq D$, $0 \leq \gamma \leq |i(\ell)|$. Then, for any allocation $g'(c)$ such that $g'_q = \min\{1, \gamma\}$ for some $q \in i(\ell)$, $g'_s = 0$ $\forall s \in i(\ell) \backslash q$, and $g'_j = g_j$ $\forall j \notin i(\ell)$ we have*

$$\mathbb{E}_{g'(c)}\Big[W_k(a, p)\Big] \geq \mathbb{E}_{g(c)}\Big[W_k(a, p)\Big], \quad \forall k \leq D.$$

Proof. W. l. o. g. let $s_1, ..., s_d$ be sellers producing the database ℓ (here, $d = |i(\ell)|$). We first introduce some helpful notation. Specifically, let $a_{1:d} = (a_1, ..., a_d) \in \{0, 1\}^d$ and $a_{-1:d} = (a_{d+1}, ..., a_N) \in \{0, 1\}^{N-d}$ be the allocation of the first d sellers and of the rest of the sellers respectively. Thus, we can rewrite $a = (a_{1:d}, a_{-1:d})$.

In this case, from Lemma B.3 it follows that $\forall a_{-1:d}$, $\forall p$, for any two $a_{1:d}, a'_{1:d} : ||a_{1:d}|| > 0, ||a'_{1:d}|| > 0$, we have $W_k((a_{1:d}, a_{-1:d}), p) = W_k((a'_{1:d}, a_{-1:d}), p)$ for any $k \leq D$.

Consider now the expected induced value of a database k under the probabilistic allocation $g(c)$:

$$\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] = \sum_{a\in\{0,1\}^N} \prod_{i=1}^{N} g_i^{a_i}(c)(1-g_i(c))^{1-a_i} W_k(a,p) =$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1)\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((0,...,0,a_{-1:d}),p) +$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} g_1\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((1,...,0,a_{-1:d}),p) +$$

$$...$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} g_1\cdot ...\cdot g_d \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((1,...,1,a_{-1:d}),p) =$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1)\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((0,...,0,a_{-1:d}),p) +$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-(1-g_1)\cdot ...\cdot (1-g_d)) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((a_{1:d},a_{-1:d}),p).$$

Here, $||a_{1:d}|| > 0$ implies that there exists $q \leq d$, s.t., $a_{1:d}(q) = 1$. We now can rewrite the expression above as

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1)\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((0,...,0,a_{-1:d}),p) +$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((a_{1:d},a_{-1:d}),p) -$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1)\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((a_{1:d},a_{-1:d}),p) =$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} W_k((a_{1:d},a_{-1:d}),p) +$$

$$\sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1)\cdot ...\cdot (1-g_d) \prod_{i=1}^{N-d} g_{d+i}^{a_{-1:d}(i)}(1-g_{d+i})^{1-a_{-1:d}(i)} \times$$

$$(\underbrace{W_k((0,...,0,a_{-1:d}),p) - W_k((a_{1:d},a_{-1:d}),p)}_{\leq 0}).$$

Notice that the last term in the expression above is non-positive due to Assumption 3. At the same time, the first summand does not depend on $g_1, ..., g_d$. Thus, the problem now is to find such an assignment of $g_1, ..., g_d$ that is feasible (i.e., $0 \leq g_i \leq 1$, $\forall i \in i(\ell)$ and $\sum_{i \in i(\ell)} g_i = \gamma$) and that minimizes $(1 - g_1) \cdot ... \cdot (1 - g_d)$. We claim that $g_q = \min\{\gamma, 1\}$ for some $q \in i(\ell)$ and $g_s = 0$ $\forall s \in i(\ell) \backslash q$ is such an assignment.

To see this, let's consider only the case when $q = 1$ (all other cases are symmetric). We proceed by induction in d. If $d = 1$, the statement is trivial. Indeed, in this case, $g_1 = \min\{1, \gamma\}$ minimizes $(1 - g_1)$. Now consider the case $d = 2$. In this case, we are solving the following problem:

$$\begin{aligned} & \min_{g_1, g_2} (1 - g_1)(1 - g_2) \\ \texttt{s.t.} \quad & g_1 + g_2 = \gamma \\ & g_1, g_2 \in [0, 1]. \end{aligned}$$

If $\gamma \geq 1$, let us rewrite the objective function as $\min_{g_2}(1 - \gamma + g_2)(1 - g_2) = \min_{g_2} 1 - g_2^2 - \gamma + \gamma g_2$. In this case, the concave objective function is minimized at the boundary of the $[0, 1]$ interval, namely when $g_2 = 0$ (respectively, $g_1 = \min\{1, \gamma\} = 1$). The optimal objective value in this case is 0.

If $\gamma < 1$, let us rewrite the objective as $\min_{g_1}(1 - \gamma + g_1)(1 - g_1) = \min_{g_1} 1 - g_1^2 - \gamma + \gamma g_1$. In this case, $g_1 = \gamma$ minimizes the objective function. The optimal objective value in this case is $1 - \gamma$.

Assume now that the statement is true for some r, $1 \leq r \leq d$, i.e., $(1 - g_1) \cdot ... \cdot (1 - g_r)$ is minimized by setting $g_1 = \min\{1, \gamma\}$. Consider the following problem for $r + 1$:

$$\begin{aligned} & \min_{g_1, ..., g_{r+1}} (1 - g_1) \cdot ... \cdot (1 - g_{r+1}) \\ \texttt{s.t.} \quad & \sum_{i=1}^{r+1} g_i = \gamma \\ & g_i \in [0, 1] \ \ \forall i = 1, ..., r + 1. \end{aligned}$$

We can rewrite it as

$$\begin{aligned} & \min_{g_1, ..., g_{r+1}} (1 - g_1)\Big((1 - g_2) ... \cdot (1 - g_{r+1})\Big) \\ \texttt{s.t.} \quad & \sum_{i=2}^{r+1} g_i = \gamma - g_1 \\ & g_i \in [0, 1] \ \ \forall i = 1, ..., r + 1. \end{aligned}$$

From the induction hypothesis, it follows that for any $0 \leq g_1 \leq \gamma$ setting $g_2 = \gamma - g_1$, $g_3 = ... = g_{r+1} = 0$ minimizes $(1 - g_2) \cdot ... \cdot (1 - g_{r+1})$. Therefore, the problem is to find such g_1, g_2 that solve

$$\begin{aligned} & \min_{g_1, g_2} (1 - g_1)(1 - g_2) \\ \texttt{s.t.} \quad & g_2 = \gamma - g_1 \\ & g_1, g_2 \in [0, 1]. \end{aligned}$$

As we have shown above, the solution to this problem is $g_1 = \min\{1, \gamma\}$, $g_2 = 0$. Q.E.D.

□

Lemma 3.4. *Let* $g : \mathbb{R}^N_{\geq 0} \to [0,1]^N$ *maximizes*

$$\mathbb{E}_f\Big[\sum_{k=1}^{D}\Big(\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)}\Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i(c)\Big)\Big]$$

subject to monotonicity of ex-interim allocation and constraints (13), (14). *Let also* $h_i(c) = g_i(c)c_i + \int_{c_i}^{\beta_i} g_i(q_i, c_{-i})dq_i$ *for every* $i = 1, ..., N$, $\forall c$. *Then* $\mathcal{A} = \langle g, h\rangle$ *is an optimal surplus maximizing reverse auction.*

Proof. The proof is similar to the one presented in (Myerson, 1981). Consider the surplus of the auctioneer.

$$\mathbb{E}_f[U(g,h)] = \mathbb{E}_f\Big[\sum_{k=1}^{D}\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i=1}^{N}h_i(c)\Big] = \int_c\Big(\sum_{k=1}^{D}\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] -$$
$$\sum_{i=1}^{N}h_i(c)\Big)f(c)dc = \int_c\sum_{k=1}^{D}\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big]f(c)dc - \int_c\sum_{i=1}^{N}h_i(c)f(c)dc =$$
$$\sum_{k=1}^{D}\int_c\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big]f(c)dc - \sum_{i=1}^{N}\int_c c_ig_i(c)f(c)dc + \sum_{i=1}^{N}\int_c c_ig_i(c)f(c)dc - \int_c\sum_{i=1}^{N}h_i(c)f(c)dc =$$
$$\sum_{k=1}^{D}\int_c\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big]f(c)dc - \sum_{k=1}^{D}\sum_{i\in i(k)}\int_c c_ig_i(c)f(c)dc + \sum_{i=1}^{N}\int_c c_ig_i(c)f(c)dc - \sum_{i=1}^{N}\int_c h_i(c)f(c)dc =$$
$$\underbrace{\sum_{k=1}^{D}\int_c\Big(\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)}c_ig_i(c)\Big)f(c)dc}_{A} + \sum_{i=1}^{N}\int_c\Big(c_ig_i(c) - h_i(c)\Big)f(c)dc =$$
$$A + \sum_{i=1}^{N}\int_{c_i}\int_{c_{-i}}(c_ig_i(c) - h_i(c))f_i(c_i)f_{-i}(c_{-i})dc_idc_{-i} = A - \sum_{i=1}^{N}\int_{c_i}\mathbb{E}_{f_{-i}}[u_i(g,c_i,h)]f_i(c_i)dc_i.$$

Given that we are looking for a feasible mechanism, we can rewrite $\mathbb{E}_{f_{-i}}[u_i(g, c_i, h)] = \mathbb{E}_{f_{-i}}[u_i(g, \beta_i, h)] + \int_{c_i}^{\beta_i} G_i(g, q_i)dq_i$ (see condition 2 of Lemma B.1). Therefore,

$$\mathbb{E}_f[U(g,h)] = A - \sum_{i=1}^{N} \int_{c_i} \mathbb{E}_{f_{-i}}[u_i(g,\beta_i,h)]f_i(c_i)dc_i - \sum_{i=1}^{N} \int_{c_i}\int_{c_i}^{\beta_i} G_i(g,q_i)dq_i f_i(c_i)dc_i =$$

$$A - \underbrace{\sum_{i=1}^{N} \mathbb{E}_{f_{-i}}[u_i(g,\beta_i,h)]}_{B} - \sum_{i=1}^{N} \int_{c_i}\int_{c_i}^{\beta_i} G_i(g,q_i)dq_i f_i(c_i)dc_i = A - B - \sum_{i=1}^{N} \int_{c_i}\int_{c_i}^{\beta_i} G_i(g,q_i)dq_i dF_i(c_i) =$$

$$A - B - \sum_{i=1}^{N} \Big[F_i(c_i)\int_{c_i}^{\beta_i} G_i(g,q_i)dq_i\Big|_{\alpha_i}^{\beta_i} - \int_{\alpha_i}^{\beta_i} F_i(c_i)d\int_{c_i}^{\beta_i} G_i(g,q_i)dq_i\Big] =$$

$$A - B - \sum_{i=1}^{N} \Big[0 + \int_{\alpha_i}^{\beta_i} F_i(c_i)G_i(g,c_i)dc_i\Big] = A - B - \sum_{i=1}^{N} \int_{\alpha_i}^{\beta_i}\int_{c_{-i}} g_i(c_i,c_{-i})f_{-i}(c_{-i})dc_{-i}F_i(c_i)dc_i =$$

$$A - B - \sum_{i=1}^{N} \int_{c} g_i(c_i,c_{-i})\frac{F_i(c_i)}{f_i(c_i)}f(c)dc = \sum_{k=1}^{D} \int_{c} \Big(\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)} \Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i(c)\Big)f(c)dc - B.$$

From Lemma B.1 it follows that

$$B = \sum_{i=1}^{N} \mathbb{E}_{f_{-i}}[u_i(g,\beta_i,h)] = \sum_{i=1}^{N} \Big(\mathbb{E}_{f_{-i}}[u_i(g,c_i,h)] - \int_{c_i}^{\beta_i} G_i(g,q_i)dq_i\Big) =$$

$$\sum_{i=1}^{N} \Big(\int_{c_{-i}} (h_i(c) - g_i(c)c_i)f_{-i}(c_{-i})dc_{-i} - \int_{c_i}^{\beta_i}\int_{c_{-i}} g_i(q_i,c_{-i})f_{-i}(c_{-i})dc_{-i}dq_i\Big) =$$

$$\sum_{i=1}^{N} \Big(\int_{c_{-i}} (h_i(c) - g_i(c)c_i - \int_{c_i}^{\beta_i} g_i(q_i,c_{-i})dq_i)f_{-i}(c_{-i})dc_{-i}\Big).$$

From the third condition of Lemma B.1 it follows that $h_i(c) - g_i(c)c_i - \int_{c_i}^{\beta_i} g_i(q_i,c_{-i})dq_i \geq 0$. This means that the payment rule that maximizes the expected surplus of the auctioneer, $\mathbb{E}_f[U(g,h)]$, must satisfy $h_i(c) - g_i(c)c_i - \int_{c_i}^{\beta_i} g_i(q_i,c_{-i})dq_i = 0$. Consequently, $h_i(c) = g_i(c)c_i + \int_{c_i}^{\beta_i} g_i(q_i,c_{-i})dq_i$.

This means that the problem now is reduced to finding such an allocation function g that maximizes

$$\mathbb{E}_f\Big[\sum_{k=1}^{D} \Big(\mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)} \Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i(c)\Big)\Big].$$

Q.E.D. □

Lemma 3.5. *Let $g^*(c)$ be a solution of (16). Then, if for a database ℓ there exists a seller s_i with $i \in i(\ell)$ such that $g_i^*(c) > 0$, then $\sum_{i\in i(\ell)} g_i^*(c) = 1$.*

Proof. Assume that there are d sellers $s_1, ..., s_d$ who produce the database ℓ, i.e. $1, ..., d \in i(\ell)$. Assume that $g_1^*(c) > 0$ but $\sum_{i\in i(\ell)} g_i^*(c) < 1$. Consequently, $g_1^*(c) < 1$. Consider the first term under the expectation in Equation 16:

$$\sum_{k=1}^{D} \mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] = \sum_{k=1}^{D} \sum_{a_{-1:d}\in\{0,1\}^{N-d}} g_1 \cdot ... \cdot g_d \prod_{i=1}^{N-d} g_i^{a_{-1:d}(i)}(1-g_i)^{1-a_i} W_k((1,...,1,a_{-1:d}),p) +$$

$$\sum_{k=1}^{D} \sum_{a_{-1:d}\in\{0,1\}^{N-d}} g_1 \cdot ... \cdot (1-g_d) \prod_{i=1}^{N-d} g_i^{a_{-1:d}(i)}(1-g_i)^{1-a_i} W_k((1,...,0,a_{-1:d}),p) +$$

$$...$$

$$\sum_{k=1}^{D} \sum_{a_{-1:d}\in\{0,1\}^{N-d}} (1-g_1) \cdot ... \cdot (1-g_d) \prod_{i=1}^{N-d} g_i^{a_{-1:d}(i)}(1-g_i)^{1-a_i} W_k((0,...,0,a_{-1:d}),p).$$

Observe, that the sum above is linear in g_1, i.e., we can rewrite it as

$$\sum_{k=1}^{D} \mathbb{E}_{g(c)}\Big[W_k(a,p)\Big] = g_1\lambda(g_{-1}) + \gamma(g_{-1}),$$

where $\lambda(g_{-1})$ and $\gamma(g_{-1})$ are independent of g_1. Now, we can rewrite Equation 16 as follows

$$\mathbb{E}_f\Big[\sum_{k=1}^{D}\Big(\mathbb{E}_{g^*(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)}\Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i^*(c)\Big)\Big] =$$

$$\mathbb{E}_f\Big[g_1^*(c)\lambda(g_{-1}^*) - \phi_1(c_1)g_1^*(c) - ... - \phi_d(c_d)g_d^*(c) - \sum_{k=2}^{D}\sum_{i\in i(k)} \phi_i(c_i)g_i^*(c) + \gamma(g_{-1}^*)\Big] =$$

$$\mathbb{E}_f\Big[g_1^*(c)\Big(\lambda(g_{-1}^*) - \phi_1(c_1)\Big) \underbrace{- \phi_2(c_2)g_2^*(c) ... - \phi_d(c_d)g_d^*(c) - \sum_{k=2}^{D}\sum_{i\in i(k)} \phi_i(c_i)g_i^*(c) + \gamma(g_{-1}^*)}_{\texttt{does not depend on } g_1^*(c)}\Big].$$

Here, $g_1^*(c) > 0$ implies that $\lambda(g_{-1}^*) \geq \phi_1(c_1)$. Thus, if for some positive $\epsilon \leq 1 - g_1^*$ we take any $g'(c)$ such that $g_1'(c) = g_1^*(c) + \epsilon$ and $g_i'(c) = g_i^*(c)$ for all $i \neq 1$, then

$$\mathbb{E}_f\Big[\sum_{k=1}^{D}\Big(\mathbb{E}_{g'(c)}\Big[W_k(a,p)\Big] - \sum_{i\in i(k)}\Big(c_i + \frac{F_i(c_i)}{f_i(c_i)}\Big)g_i'(c)\Big] =$$

$$\mathbb{E}_f\Big[(g_1^*(c)+\epsilon)\Big(\lambda(g_{-1}^*) - \phi_1(c_1)\Big) - ... - \phi_d(c_d)g_d^*(c) - \sum_{k=2}^{D}\sum_{i\in i(k)} \phi_i(c_i)g_i^*(c) + \gamma(g_{-1}^*)\Big] \geq$$

$$\mathbb{E}_f\Big[g_1^*(c)\Big(\lambda(g_{-1}^*) - \phi_1(c_1)\Big) - ... - \phi_d(c_d)g_d^*(c) - \sum_{k=2}^{D}\sum_{i\in i(k)} \phi_i(c_i)g_i^*(c) + \gamma(g_{-1}^*)\Big].$$

This means that $g^*(c)$ cannot be an optimal solution. Contradiction. Q.E.D. □

Lemma B.3. *For any database k and any price p, for any $a, a' \in \{0,1\}^N$ such that $\forall \ell \notin i(k)$: $a_\ell = a'_\ell$ and $\exists s, q \in i(k) : a_q = 1, a'_s = 1$ it follows*

$$W_k(a,p) = W_k(a',p).$$

Proof. From our assumption that buyers are indifferent about the identities of sellers it follows that $\forall k$, $\forall b_j$, $\forall p$ and for any a and a' satisfying the conditions above, we have $r_j^*(p,a) = r_j^*(p,a')$ and $m_j^*(p,a) = m_j^*(p,a')$. Consequently, $r^*(p,a) = \sum_{j=1}^L r_j^*(p,a) = \sum_{j=1}^L r_j^*(p,a') = r^*(p,a')$; similarly, $m^*(p,a) = m^*(p,a')$.

It follows that $\pi(z,a) = \pi(z,a')$. This results in $V(r,a) = V(r,a')$ and consequently in $ext_k(a,p) = ext_k(a',p)$. From this it immediately follows that $W_k(a,p) = W_k(a',p)$. Q.E.D. □

Proposition 3.6. *The share of the buyers' surplus achieved in Algorithm 1 auction is lower bounded by zero.*

Proof. We prove this statement by providing an example of the domain in which buyers reach zero surplus. Consider a domain with a single buyer, $L = 1$. Assume that there are $N = 2$ sellers each producing a single database, i.e., $D = 2$. Let $c_1, c_2 \sim U[0,3]$ and $c_1 = c_2 = 0$. The virtual cost function for both sellers is $\phi(c) = c + \frac{F(c)}{f(c)} = 2c$; thus, $\phi_1(c_1) = \phi_2(c_2) = 0$.

Assume that the value function of the buyer is $v_1(r_1, a) = 5\min\{r_1, 1\}$ if both databases are allocated (i.e., $a = (1,1)$) and $v_1(r_1, a) = 0$ otherwise. The buyer's endowment is $e = 5$. With a single buyer, the aggregate value function $V(r,a) = v_1(r,a)$ and the aggregate endowment $E = e$.

Let us now compute the induced values. First, $ext_1(a,p) = ext_2(a,p) = 5$ for all $p \leq 5$ and $a = (1,1)$. Also $ext_1(a,p) = ext_2(a,p) = 0$ if $a \neq (1,1)$ or if $p > 5$. Thus, $W_1(a,p) = W_2(a,p) = 2.5$ for any $p \leq 5$ if $a = (1,1)$ and $W_1(a,p) = W_2(a,p) = 0$ otherwise.

Given price p, the allocation problem in this case is $\max_{a_1,a_2} \Big\{ W_1((a_1,a_2),p) + W_2((a_1,a_2),p) - \phi_1(c_1)a_1 - \phi_2(c_2)a_2 \Big\}$. The solution to this problem is $(a_1^*, a_2^*) = (1,1)$. In this case, the objective value is $2.5 + 2.5 - 0 \cdot 1 - 0 \cdot 1 = 5$ for any $p \leq 5$. Payments are computed as follows: $t_1 = \frac{1}{2}(0 + 5 - 0) = 2.5$, $t_2 = \frac{1}{2}(0 + 5 - 0) = 2.5$. Setting the price $p = 5$, the market becomes balanced. In this case, the buyer pays $5 = (t_1 + t_2)$ for a single row of answers for his query, $r_1^*(p,a) = 1$. However, the buyer's surplus is 0. Q.E.D. □

References

Bakos, Y., & Brynjolfsson, E. (1999). Bundling information goods: Pricing, profits, and efficiency. *Management Science*, *45*(12), 1613–1630.

Balazinska, M., Howe, B., Koutris, P., Suciu, D., & Upadhyaya, P. (2013). A discussion on pricing relational data. *In Search of Elegance in the Theory and Practice of Computation*, *8000*, 167–173.

Bernstein, A., Hendler, J., & Noy, N. (2016). A new look at the semantic web. *Commun. ACM*, *59*(9), 35–37.

Blue Kai, Inc. (2011). Whitepaper: Data management platforms demystified. Tech. rep..

Buil-Aranda, C., Hogan, A., Umbrich, J., & Vandenbussche, P.-Y. (2013). Sparql web-querying infrastructure: Ready for action?. In *Proceedings of the 12th International Semantic Web Conference - Part II*, ISWC '13, pp. 277–293, New York, NY, USA. Springer-Verlag New York, Inc.

Chambers, C. P., & Echenique, F. (2009). Supermodularity and preferences. *Journal of Economic Theory*, *144*(3), 1004 – 1014.

Cheng, J. Q., & Wellman, M. P. (1998). The walras algorithm: A convergent distributed implementation of general equilibrium outcomes. *Comput. Econ.*, *12*(1), 1–24.

Deep, S., & Koutris, P. (2016). The design of arbitrage-free data pricing schemes. Tech. rep., University of Wisconsin-Madison, Working Paper.

Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., Schlaefer, N., & Welty, C. (2010). Building Watson: An overview of the DeepQA project.. *AI Magazine*, *31(3)*, 59–79.

Goldberg, A., & Hartline, J. (2001). Competitive auctions for multiple digital goods. *Springer Berlin Heidelberg, Berlin, Heidelberg*, 416–427.

Goldberg, A. V., & Hartline, J. D. (2003). Envy-free auctions for digital goods. In *Proceedings of the 4th ACM Conference on Electronic Commerce*, EC '03, pp. 29–35, New York, NY, USA. ACM.

Goldberg, A. V., Hartline, J. D., & Wright, A. (2001). Competitive auctions and digital goods. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '01, pp. 735–744, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

Greg, M. (2011). Inside the bloomberg machine. WallStreet and Technology, http://www.wallstreetandtech.com/trading-technology/inside-the-bloomberg-machine/d/d-id/1264634?

Grubenmann, T., Bernstein, A., Moor, D., & Seuken, S. (2018). Financing the web of data with delayed-answer auctions. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pp. 1033–1042, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Grubenmann, T., Dell'Aglio, D., Bernstein, A., Moor, D., & Seuken, S. (2017). Decentralizing the Semantic Web: who will pay to realize it?. In *Proceedings of the Workshop on Decentralizing the Semantic Web (DeSemWeb)*.

Hall, A. S., Shan, Y., Lushington, G., & Visvanathan, M. (2013). An overview of computational life science databases and exchange formats of relevance to chemical biology research. *Combinatorial Chemistry and High Throughput Screening*, *16*(3), 189–198.

HCLS (2001). Semantic Web Health Care and Life Sciences (HCLS) Interest Group. https://www.w3.org/2001/sw/hcls/.

Koutris, P., Upadhyaya, P., Balazinska, M., Howe, B., & Suciu, D. (2013). Toward practical query pricing with querymarket. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 613–624.

Koutris, P., Upadhyaya, P., Balazinska, M., Howe, B., & Suciu, D. (2015). Query-based data pricing. In *Journal of the ACM (JACM)*, Vol. 62.

Mas-Colell, A., Whinston, M. D., & Green, J. R. (1995). *Microeconomic Theory*. Oxford University Press, New York.

McKinsey report (2016). Creating a successful Internet of Things Data Marketplace. http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/creating-a-successful-internet-of-things-data-marketplace?cid=soc-web.

Microsoft Azure Data Marketplace https://datamarket.azure.com/home.

Moor, D., Grubenmann, T., Seuken, S., & Bernstein, A. (2015). A double auction for querying the web of data. In *The Third Conference on Auctions, Market Mechanisms and Their Applications*.

Myerson, R. B. (1981). Optimal auction design. *Math. Oper. Res.*, *6*(1), 58–73.

Myerson, R. B., & Satterthwaite, M. A. (1983). Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, *29*(2), 265–281.

Nisan, N., Roughgarten, T., Tardos, E., & Vazirani, V. (2007). *Algorithmic Game Theory*. Cambridge University Press.

Ramel (2016). Microsoft closing azure datamarket. Application Development Trends Magazine, https://adtmag.com/articles/2016/11/18/azure-datamarket-shutdown.aspx.

Schomm, F., Stahl, F., & Vossen, G. (2013). Marketplaces for data: An initial survey. *ACM SIGMOD*, *42*(1), 15–26.

Thomson-Reuters (2015). Thomson reuters annual report. Tech. rep..

Tirole, J., & Laffont, J.-J. (1993). *A Theory of Incentives in Procurement and Regulation*. MIT Press.

Varian, H. R. (1995). Pricing information goods. Tech. rep., University of Michigan.

Varian, H. R. (1997). Versioning information goods. Tech. rep., University of California.

W3C (2014). Linked Data. https://www.w3.org/standards/semanticweb/data.