Share: Stackelberg-Nash based Data Markets

Anonymous Author(s)

ABSTRACT

With the prevalence of data-driven intelligence, data markets with various data products are gaining considerable interests as a promising paradigm for commoditizing data and facilitating data flow. In this paper, we present **S**tackelberg-Nas**h** based D**a**ta Ma**r**k**e**ts (*Share*), which is the first work to introduce both Stackelberg game and Nash game into buyer-leading multi-seller data markets to realize absolute pricing for data. We propose a three-stage Stackelberg-Nash game to model trading dynamics which not only balances the profits of all participants who are considered selfish and want to maximize their own profit but also ensures buyer' priority and solves the seller selection problem based on sellers' inner competition. We define Stackelberg-Nash Equilibrium and use backward induction to solve the equilibrium. For inner Nash equilibrium, we propose both conventional direct derivation and a novel mean-field based method for complicated cases along with provable approximation guarantees. Experiments on real datasets verify the effectiveness and efficiency of *Share*.

ACM Reference Format:

Anonymous Author(s). 2018. Share: Stackelberg-Nash based Data Markets. In *SIGMOD '23.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Data products (e.g., machine learning models, aggregate statistics, and query services) have paved the way for a variety of data-driven tasks in many different industries. High-performance data products require a large amount of high-quality data. While there are a wealth of data generated from different sources, they are highly dispersed, which brings significant challenges to data aggregation. Besides, there is a gap between data supply and demand, and data suppliers or demanders usually lack the necessary resources and techniques to survey the vast data sources and turn data into data products. Thus, despite the increasingly available and enriched data, the wealth of data is far from being fully exploited. Recent studies and practices have demonstrated data markets as a promising paradigm to commoditize data and connect data suppliers and demanders.

A typical data market consists of three parties, buyers, brokers, and sellers. Buyers propose demands for data products and pay for them; brokers help facilitate the transactions between buyers and sellers as well as take charge of manufacturing data products out of data; sellers offer data with different quality and sell data to brokers in exchange for compensations. While many recent works have addressed different aspects of the data market, there are several challenges that remain to be addressed. We use a motivating example below to describe the desired properties of a data market and the associated challenges.

Medical Data Market Example. Consider a drug company as a buyer who has demands for a data model (one kind of data products) based on medical data to study the effectiveness of a new drug. A broker buys data from multiple hospitals, uses the data to train the required model and sells the model to the company. Hospitals as sellers own data generated from medical services or medical research and can sell data to the broker with different quality depending on the privacy preserving mechanisms which may add noise to the data (since the data are privacy sensitive) and data price offered by the broker.

Desired Properties. In a data market such as above, all three parties are *selfish*, i.e. have their own *revenue* and *cost* and aim to maximize their own profit. Moreover, how they act in the market affects each other. If the drug company sets a low price for buying the desired model for profit, the broker may pay little to buy the training data to guarantee profit after recovering costs (e.g., for training or operating), and therefore the hospitals offer poor-quality data, which causes low-performance models and in turn harms the profits of the company. Therefore, it's necessary to design data markets which consider the three selfish but interdependent parties and maximize profits of buyers, brokers, and sellers simultaneously to motivate their participation and boost market vitality ($\mathcal{P}_1$).

While all three parties need to maximize their profit, they take different roles in the market flow. Data sellers (owners) such as the hospitals typically do not regard data selling as the main business. Rather, data transactions are often initiated by data buyers (demanders) such as the drug company. Therefore, it's necessary to construct data markets with the buyer-leading (demand-leading) property ($\mathcal{P}_2$) which takes full account of buyers' priority against others to encourage data trading.

Pricing rules play a significant role in the market mechanism. In order to mirror real markets, prices should be absolute, instead of being pegged to a certain benchmark (relative prices), to reflect actual values of data. Besides, incentive mechanisms should involve all three parties in the pricing process rather than forcing them to passively receive certain prices which may discourage their participation. Thus, it's necessary to formulate data markets in which absolute prices can be directly decided by market participants based on their mutual interactions ($\mathcal{P}_3$).

Considering such buyer-leading and multi-seller market and the desired three properties, an important research question is: *how to build a well-functioning data market with an absolute pricing mechanism where both the profit needs for all selfish participants and the leading position for buyers are considered.*

Challenges. Recent works have proposed data markets with different products including machine learning models [1, 8, 27, 35, 36], data aggregation statistics [2, 22, 28, 42], or query services [29, 30,

SIGMOD '23, June 18–23, 2023, Seattle, WA

ACM ISBN 978-1-4503-XXXX-X/23/06…$15.00
https://doi.org/10.1145/1122445.1122456

32]. These works vary in design goals, and typically address one aspect or the need for one party, such as model quality optimization for buyers [22], revenue maximization for sellers [8], fair revenue allocation [27], social welfare maximization [28], or market protection from strategic participants [20]. A few recent works [32, 35, 36] proposed three-sided data markets which aim to simultaneously satisfy the needs of all three parties, however, they assume the brokers are neutral and do not consider profit maximization for them. In summary, existing works do not work well in data markets with three selfish but interdependent parties that all want to realize profit optimization ($\mathcal{P}_1$).

A reasonable pricing mechanism plays an important role in balancing the profits of all participants. Absolute data pricing ($\mathcal{P}_3$), however, is far from trivial. Due to the special properties of data (e.g., freely replicable and inherently combinatorial due to correlations with signals in other data) summarized in [1], pricing mechanisms for physical goods cannot be directly applied to data. Most of the existing works [1, 8, 29, 42, 56] employed relative prices for data or data products, which fail to capture the value of the data and motivate participants. Therefore, the first challenge is (C_1): *How to design an incentive mechanism including absolute pricing rules for data markets that can balance the profits of all three selfish and interdependent participants.*

Moreover, little progress is made in existing works on buyer-leading data markets ($\mathcal{P}_2$). Though efforts have been made to satisfy buyers' needs (e.g., demands for product utility and purchase budget) in [1, 36], data markets where transactions are initiated by buyers are still understudied. Furthermore, the decisive influence buyers can have on the price was ignored. Therefore, the second challenge is (C_2): *How to embody the advantages of buyers against other parties in the buyer-leading data markets.*

To give priority to buyers ($\mathcal{P}_2$), it's important to ensure they gain the *best* data (with the highest data quality). Typically there are a large number of data sellers with usable data, so it's critical to select the *best* data from the sellers, which we call *seller selection problem*. Many existing works made the broker responsible for seller selection [2, 36], which not only requires the broker's capability of learning the data quality but also ignores the sellers' ability of choosing their provided data quality. We argue that there should be inner competition among sellers, which can make the winners as the selected sellers without the assistance of brokers or others. Hence, the third challenge is (C_3): *How to model the inner competition among data sellers to select the best set of data for data transaction.*

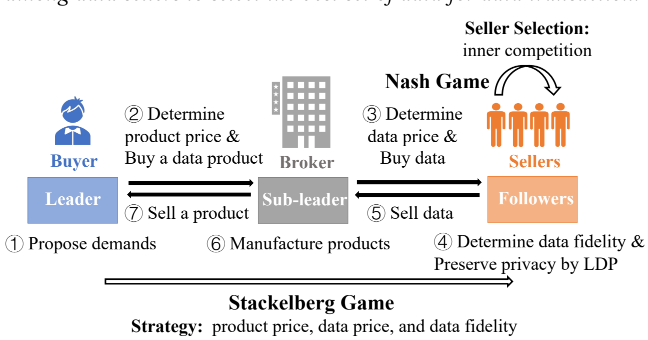


Figure 1: A data market framework with Stackelberg-Nash game.

Contributions. In this paper, we address the identified three challenges by proposing **S**tackelberg-Nas**h** based D**a**ta Ma**r**k**e**ts (*Share*) utilizing game theory, as in Figure 1.

Game theory is widely used [2, 3, 51] to design multi-objective incentive mechanisms and provide optimal strategies for participants. We adopt game theory to solve the problem of profit maximization for all the parties in data markets (C_1). The interactions of the three entities are modeled as Stackelberg game, in which each participant can achieve her profit-maximization goal by making her optimal strategy. Moreover, absolute prices of data are modeled as strategies of participants and are directly determined in the game process.

Considering the buyer priority, we formalize the interactions among three parties as a three-stage Stackelberg game which can deal with the different status of participants by regarding buyers as leaders, brokers as sub-leaders, and sellers as followers. As shown in Figure 1, the buyer first announces what data product she demands for and determines the product price based on her profit-maximization goal; the broker then tries to buy data from sellers; each seller then chooses what data quality to provide. Buyers are thus endowed with a dominant position in two aspects: the priority of initiating transactions and the intensive influence on prices (C_2).

For the problem of seller selection, we model the inter-seller competition as a Nash game because of its advantage in modeling sellers' equal positions (C_3), which is the first work that applies Nash game for the seller selection problem. In our Nash model, the buyer needs data which can be from multiple sellers, and the data quantity that each seller can sell is determined through the competition based on data quality (data fidelity). Data sellers decide on the data quality which is correlated with the privacy preserving level. We assume sellers use local differential privacy [15] to perturb their data, hence lower privacy corresponds to less noise and higher data quality. Nash equilibrium among sellers is preferred, which can guarantee that no seller can change her strategy individually and the seller selection result is stable. However, it's challenging to find the equilibrium point especially when the number of sellers is large since the analytic solutions may be complicated and hard to derive. In *Share*, we apply the direct derivation and propose a mean-field based approximation for complex cases to derive Nash equilibrium.

Our goal in this paper is not to cover all critical issues in buyer-leading data markets, but rather to propose a reasonable and feasible mechanism to satisfy several essential desiderata. Overall, the major contributions are summarized as follows.

- We present *Share*, a buyer-leading and multi-seller data market framework based on a three-stage Stackelberg-Nash game, which satisfies the properties of buyer-leading, for-all profit maximization, and absolute pricing.
- We apply Nash game for the seller selection problem, which formulates sellers' inner competition and incorporates seller selection into the game process among three parties.
- We define Stackelberg-Nash Equilibrium in data markets and derive it by backward induction. To solve inner Nash game, we apply the direct derivation as well as design a novel mean-field method for complex cases, for which error analysis is conducted.
- We conduct experiments on real datasets to verify the effectiveness and efficiency of *Share*.

Organization. The rest of the paper is organized as follows. Section 2 presents the related work. Section 3 provides the preliminaries. We provide a data market framework with Stackelberg-Nash game and design the market mechanism for three parties in Section 4. Approaches to deriving the market equilibrium and the detailed market dynamics are presented in Section 5. We report the experimental results and findings in Section 6. Section 7 draws a conclusion and discusses future work.

2 RELATED WORK

In this section, we discuss related work on the data market and game theory.

2.1 Data Market

Data markets trade data either in direct or indirect forms (derived data products). Bloomberg [5], SafeGraph [46], Dawex [13] and et al. implemented data markets where buyers directly purchase data. Koutris et al. [29, 30] proposed query-based data markets which allow buyers to obtain information through querying the database and pay for the query. Recently, model-based data markets [1, 8, 27, 35, 36] have been proposed. Agarwal et al. [1] applied algorithmic game theory for two-sided data-driven machine learning model markets. Dealer [35, 36], an end-to-end model marketplace with differential privacy, more comprehensively considered the needs of sellers and buyers. It also regulated the broker's role as model pricing and model training.

In Dealer, however, it is assumed that the broker to be neutral without her own profit consideration and determines model prices only for single goal optimization, i.e., revenue maximization for sellers. An et al. [2], who studied transactions for crowdsensing data, were devoted to the multiple goal optimization. Nevertheless, they oversimplified the markets in terms of transaction objects, market participants, and profit function formulations. In this paper, we combine multiple game mechanisms to model for-all profit-maximization data markets with unrestricted data and data products, privacy consideration for data sellers, as well as exquisitely designed profit functions based on economics theories.

As for data pricing, there are several surveys [10, 43, 44] claiming fundamental principles and reviewing the development and evolution of pricing models. Ghorbani and Zou [23] introduced Shapley value to data valuation. Federated learning based data pricing was developed by Wang et al. [55] while reinforcement learning was adopted by Yoon et al. [59] to price data. In terms of absolute pricing mechanism, Dealer [35, 36] provided absolute prices for data models, which, however, highly rely on the survey results and can't be adjusted dynamically. Agarwal et al. [1] applied Myerson's payment function rule to determine absolute model payment but allocated relative compensations to sellers in proportion to their contributions. Therefore, we propose an absolute pricing mechanism to bridge the gap.

Many works looked at the seller selection problem. In Dealer [35, 36], brokers choose datasets to achieve Shapley coverage maximum of the trained model. In [2], combinatorial multi-armed bandit mechanism was used for seller selection by brokers. However, the selection results directly affect the profits of sellers, and therefore the seller selection problem is closely correlated to the profit maximization problem for sellers and should not be considered separately. In fact, the selection problem can be seen as the spontaneous process of the inner competition among sellers, not conducted by the broker or others. In our work, seller selection problem is formalized as the inner Nash game among sellers, which is a part of the incentive mechanism for profit optimization of all participants.

2.2 Game Theory

Game theory provides a decision-making and analysis tool for individual behaviors with conflicting objectives. Early game theory originated from the study of competition in a duopoly in microeconomics. Stackelberg game was originally proposed by von Stackelberg [52] to model the asymmetric competition among oligopoly firms. Morgenstern et al. [38] first defined the basic concepts of game, marking the preliminary formation of game theory. Nash accurately described Nash equilibrium [41] and proved its existence in N-player finite non-cooperative game with mixed strategies [40]. Later, Selten et al. [47] considered the dynamic game and proposed Subgame Perfect Nash Equilibrium. Harsanyi et al. [24] introduced incomplete information into game theory and proposed Bayesian Nash equilibrium. In terms of cooperative game, Lloyd Shapely [45] laid the foundation [21] and provided the formula of Shapley Value [48] which regulated how to allocate revenue among collaborators.

Game theory has been widely used in various situations. Many researchers used Nash game as a powerful tool to formulate and solve problems where there is competition [26, 39]. Stackelberg game was first used to formulate the determining process for oligopoly firms producing homogeneous products [52], and has been further applied to many other practical situations with hierarchical organizations [49, 54, 60]. Besides the original Stackelberg game composed of one leader and one follower, many variants [3, 34, 51] were proposed and investigated. For example, Bansal et al. [3] used a two-stage Stackelberg game to determine prices for Unmanned Aerial Vehicles. Some studies [34, 51] combined Stackelberg game and Nash game together to deal with the problems where both hierarchy and simultaneity exist, but their issues including participant roles, major actions, and optimizing goals in traditional scenarios are quite different from those in data markets.

Since Nash proposed his theory, many researchers have sought algorithms for finding Nash equilibrium. Conitzer et al. [11] showed complexity results of deriving Nash equilibrium and Daskalakis et al. [12] further studied the complexity of computing a mixed Nash equilibrium. In terms of solving Stackelberg game, backward induction approach, an iterative technique to derive dynamic game equilibrium, is often used [2, 53, 58]. In fact, establishing Stackelberg equilibrium can be formulated as a bilevel optimization problem [50]. Some studies also combined other techniques into the equilibrium solving problem [3, 33, 51].

In this paper, we design the buyer-leading data markets based on Stackelberg game because Stackelberg game concerns interactions of participants with asymmetric status and can thus realize the desired buyer-leading property while remaining the profit maximization for all parties. Moreover, we first adopt Nash game for the seller selection problem since Nash game models the competition among equals with conflicting profits and can be used for the inner competition among data sellers, which can select sellers based on their optimal strategies.

3 PRELIMINARIES

In this section, we describe local differential privacy, Shapley value and game theory used in *Share*. For reference, Table 1 summarizes the frequently used notations.

Table 1: The summary of frequently used notations.

	Notation	Definition
Buyer $\mathcal{B}$	N	data quantity for production
	p^M	unit price of data product
	v	product performance
	θ_1, θ_2	parameters of concern on each attribute
	ρ_1, ρ_2	parameters of sensitivity to each attribute
	$\mathbf{U}_1(\cdot), \mathbf{U}_2(\cdot)$	utility function of each attribute
	$\mathbf{U}(\cdot)$	utility function of the product
	$\Phi(\cdot)$	profit function of the buyer
Broker $\mathcal{A}$	p^D	unit price of data
	$\sigma_k, k = 0, 1, ..., 5$	parameters related to cost
	$\mathbf{C}(\cdot)$	cost function of production
	$\Omega(\cdot)$	profit function of the broker
Seller $\mathcal{S}_i$	i	index of seller
	m	total number of sellers
	τ_i	data fidelity
	ϵ_i	parameter in local differential privacy
	χ_i	sold data quantity
	λ_i	parameter of privacy sensitivity
	$\mathbf{L}_i(\cdot)$	privacy loss function
	$\Psi_i(\cdot)$	profit function
Data	D_i	seller $\mathcal{S}_i$'s raw dataset
	D_i^t	seller $\mathcal{S}_i$'s provided dataset
	D^t	whole dataset for manufacturing
	q_i^D	dataset quality provided by seller $\mathcal{S}_i$
	q^D	total quality of dataset for manufacturing
	q^M	data product quality
	ω_i	weight of seller $\mathcal{S}_i$'s dataset

3.1 Local Differential Privacy

Differential Privacy (DP) [16–18] is a framework for privacy protection by providing perturbation against the discovery of presence or absence of a record in a dataset. As for data markets, DP is adopted in Dealer [35, 36] to protect sellers' privacy, where the broker utilizes a perturbation algorithm on collected data and trains a set of models with DP guarantees. In our setting, the broker is not trusted, the protection for privacy should be conducted by each seller locally. Since each seller needs to provide privacy-protected data to the broker, we apply Local Differential Privacy (LDP) [15] which perturbs each data record before the data being shared with the broker.

Definition 3.1 (Local Differential Privacy). A randomized algorithm $\mathcal{A} : \mathcal{Y} \to \mathcal{Z}$ satisfies ϵ-local differential privacy if and only if for any pairs of input tuples $y, y' \in \mathcal{Y}$, and for any $z \in \mathcal{Z}$, it always holds

$$\mathbb{P}[\mathcal{A}(y) = z] \le e^{\epsilon} \cdot \mathbb{P}\left[\mathcal{A}\left(y'\right) = z\right], \tag{1}$$

where $\epsilon \ge 0$ is the privacy budget and $\mathbb{P}[\cdot]$ is the probability.

Widely used (local) DP mechanisms include Laplace mechanism [16], Index mechanism [37], and Gaussian mechanism [18]. The smaller ϵ is, the less the privacy loss is, and the worse the dataset quality is. In our context, each seller adopts a privacy scheme satisfying LDP to ensure privacy protection for each record and form her for-sale dataset with a distinctive quality corresponding to the privacy level.

3.2 Shapley Value

Shapley value [48] is an approach to fairly evaluate data importance. Proposed by Shapley [45], it satisfies the four fundamental requirements of fairness in markets, i.e., balance, symmetry, zero element, and additivity. In *Share*, Shapley value is used to measure the contribution of each seller's provided data to the data product by measuring its marginal utility improvement, e.g., the accuracy increase for a classification model or the explained variance score raise for a regression model.

Definition 3.2 (Shapley Value). Consider a set of m data sellers such that each seller $\mathcal{S}_i$ owns a dataset D_i $(i = 1, 2, ..., m)$. A coalition $\mathbb{D}$ is a subset of $\{D_1, D_2, ..., D_m\}$. Denote by $\mathbf{U}(\mathbb{D})$ a utility function that represents the performance of a data product manufactured by coalition $\mathbb{D}$ towards a task, e.g., machine learning model accuracy. Shapley value of seller $\mathcal{S}_i$ is defined as follows.

$$\mathcal{SV}_i = \frac{1}{m} \sum_{\mathbb{D} \subseteq \{D_1, D_2, \ldots, D_m\} \setminus D_i} \frac{\mathbf{U}(\mathbb{D} \cup \{D_i\}) - \mathbf{U}(\mathbb{D})}{\binom{m-1}{|\mathbb{D}|}}. \tag{2}$$

3.3 Game theory

Game theory is the study of mathematical models of strategic interactions among rational agents. An equilibrium to the game is a stable state in which no player can change her strategy to gain more payoff. Two non-cooperative games, Nash game [41] and Stackelberg game [52], are applied in this work.

Definition 3.3 (Nash Game, Nash Equilibrium). Nash game is a game where two or more players take strategy simultaneously to maximize their own expected payoff. Nash equilibrium is the stable state where no player has anything to gain by changing only one's own strategy. Formally, let T_i be the set of all possible strategies for player i, where $i = 1, 2, ..., n$. Let $\boldsymbol{t}^* = \left(t_i^*, \boldsymbol{t}_{\neg i}^*\right)$ be a strategy profile, a set consisting of one strategy for each player, where $\boldsymbol{t}_{\neg i}^* = (t_1^*, ..., t_{i-1}^*, t_{i+1}^*, ..., t_n^*)$ denotes the $n-1$ strategies of all the players except i. Let $\pi_i\left(t_i, \boldsymbol{t}_{\neg i}\right)$ be player i's payoff as a function of the strategies. The strategy profile $\boldsymbol{t}^*$ is a Nash equilibrium if

$$\pi_i\left(t_i^*, \boldsymbol{t}_{\neg i}^*\right) \ge \pi_i\left(t_i, \boldsymbol{t}_{\neg i}^*\right), \forall t_i \in T_i, i = 1, 2, ..., n. \tag{3}$$

Definition 3.4 (Stackelberg Game, Stackelberg Equilibrium). Stackelberg game is a game where one player (the *leader* L) moves first, and the other player (the *follower* F) can observe the *leader*'s behavior and move sequentially. Stackelberg equilibrium is a refinement of Nash equilibrium used in dynamic games. Similarly, the strategy profile (t_L^*, t_F^*) is a Stackelberg equilibrium if

$$\begin{aligned} \pi_L\left(t_L^*, t_F^*\right) &\ge \pi_L\left(t_L, t_F^*\right), \\ \pi_F\left(t_L^*, t_F^*\right) &\ge \pi_F\left(t_L^*, t_F\right). \end{aligned} \tag{4}$$

4 MARKET FRAMEWORK: PARTICIPANTS, MECHANISM, AND EQUILIBRIUM

In this section, we first describe the market framework from the perspectives of buyers, brokers, and sellers respectively in Section 4.1. Then, we formulate the market mechanism as a three-stage

Stackelberg-Nash game in Section 4.2 and define the market equilibrium, Stackelberg-Nash Equilibrium in Section 4.3.

4.1 Market Participants

Considering the data markets composed of three types of participants, i.e., buyers, brokers, and sellers, we define the role of each party as follows.

- **Buyer.** Buyer $\mathcal{B}$ wants to fulfill her data-driven task through data trading. Buyer $\mathcal{B}$ needs to purchase a data product from broker $\mathcal{A}$ with her claimed demands, i.e., the required product performance indicated as v and the size of the dataset for manufacturing denoted by N.
- **Broker.** Broker (Arbiter) $\mathcal{A}$ wants to make profits by bridging the transactions between buyers and sellers. Broker $\mathcal{A}$ needs to buy N data records from m sellers and make the product from the data to satisfy the product performance demand v which incurs certain manufacturing cost (e.g., computing resources). Then, broker $\mathcal{A}$ sells the product to buyer $\mathcal{B}$ in exchange for profit.
- **Seller.** Each seller $\mathcal{S}_i, i = 1, 2, ..., m$ owns a dataset D_i and wants to sell it for compensation. Seller $\mathcal{S}_i$ needs to preprocess her dataset for privacy protection and sell χ_i protected data records to broker $\mathcal{A}$. Note that χ_i is to be determined by the market mechanism and $\sum_{i=1}^{m} \chi_i = N$, the total number of data records that buyer $\mathcal{B}$ requires. Seller $\mathcal{S}_i$ receives compensation from broker $\mathcal{A}$ while incurring the privacy loss (cost) for the data she sells.

Since there are various types of markets with distinctive characteristics which cannot be all covered, we choose one type of markets with the following reasonable assumptions.

- The market is buyer-leading, and a round of transaction starts when a buyer raises the demand. Buyers orientate the market in turn (coming one at a time) as in work [1] so that only one buyer $\mathcal{B}$ is considered in each round of transactions.
- There exists one broker $\mathcal{A}$ in our market, e.g., the large-scale data trading center in the real world, which means that the competition among brokers for the same data product is beyond our consideration.
- There are a large number of sellers $\{\mathcal{S}_i | i = 1, 2, ..., m\}$, and each seller $\mathcal{S}_i$ has a dataset D_i to participate in the trading which is big enough, i.e., for any required number $\chi_i \in \mathbb{N}^+$ of data records, $|D_i| \geq \chi_i$.
- The market is highly transparent, which means each participant can see the behaviors of other participants, and cooperation is excluded, i.e., no two entities would collude to get a better outcome.

Problem Statement. Each of the participants has the *revenue* (received compensation or gained utility) and the *cost* (payment, manufacturing cost, or privacy loss). We assume that all the participants in the market are profit-driven and want to maximize their own profit, i.e., the difference between the *revenue* and *cost*, and the profit functions are known public. The problem is to find an optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$, where buyer $\mathcal{B}$, broker $\mathcal{A}$, and each seller $\mathcal{S}_i$ determine product price p^{M^*}, data price p^{D^*}, and data fidelity τ_i^* respectively such that the profits of all participants are maximized. The detailed definitions of the profit functions of buyers, brokers, and sellers are given below.

4.1.1 Profit Function of Buyer.

When buyer $\mathcal{B}$ comes to the market and asks for a data product with the required performance, she cares about her *revenue*, the utility she can get from the product and her *cost*, the payment she should give to the broker.

Revenue. The revenue of buyer $\mathcal{B}$ is correlated to the product utility which is quantified as a utility function $\mathbf{U}(\cdot)$. For data products, what buyer $\mathcal{B}$ concerns includes both 1) the product performance v and 2) the quality of the dataset used in the manufacture q^D. Apparently, product performance matters to buyer $\mathcal{B}$. For data models, v can be the explained variance, accuracy, or other indicators measuring model performance. However, it only indicates how the product performs under a certain testing environment (with respect to specific validation datasets). Dataset quality measures how *good* the raw materials are, adding a new dimension to the judgment of product utility. The dataset quality is measured as the total quality of datasets contributed by all sellers $q^D = \sum_{i=1}^{m} q_i^D = \sum_{i=1}^{m} g(\chi_i, \tau_i)$, where q_i^D is dataset quality provided by seller $\mathcal{S}_i$ and defined as $g(\chi_i, \tau_i)$ which is positively correlated with data fidelity τ_i and data quantity χ_i. Data fidelity τ_i is determined by the privacy level of LDP mechanism used by the seller, and data quantity χ_i will be determined based on sellers' inner competition. The detailed formulations for τ_i and χ_i will be elaborated later.

Combining product performance and dataset quality to measure product utility can make the quantification of product utility more comprehensive. According to the utility theory in economics [57], we define the utility of a data product as the weighted sum of the utility of the dataset quality $\mathbf{U}_1(q^D)$ and the product performance $\mathbf{U}_2(v)$.

$$\mathbf{U}\left(\boldsymbol{\chi}, \boldsymbol{\tau}, v\right) = \theta_1 \mathbf{U}_1(q^D) + \theta_2 \mathbf{U}_2(v), \tag{5}$$

where $\boldsymbol{\chi} = (\chi_1, \chi_2, ...\chi_m)$ and $\boldsymbol{\tau} = (\tau_1, \tau_2, ...\tau_m)$. θ_1 and θ_2 satisfy $\theta_1, \theta_2 \in (0, 1), \theta_1 + \theta_2 = 1$, which measure the relative significance of the two for buyer $\mathcal{B}$. In our example, if dataset quality plays a greater role in the decision-making of the drug company, the company may set $\theta_1 = 0.7$ and $\theta_2 = 0.3$. The utility of dataset quality $\mathbf{U}_1(q^D)$ and product performance $\mathbf{U}_2(v)$ are further formulated as the logarithmic functions following the principle of diminishing marginal utility in economics.

$$\begin{aligned} \mathbf{U}_1(q^D) &= \ln\left(1 + \rho_1 q^D\right), \\ \mathbf{U}_2(v) &= \ln\left(1 + \rho_2 v\right), \end{aligned} \tag{6}$$

where $\rho_1 > 0$ and $\rho_2 > 0$ refer to buyer $\mathcal{B}$'s sensitivity to these two attributes respectively. More sensitive, more utility added when the attribute gets better. For example, if higher dataset quality can bring buyer $\mathcal{B}$ much more utility, its ρ_1 would be big, meaning that buyer $\mathcal{B}$ is highly sensitive to the quality of production materials.

Cost. The *cost* of buyer $\mathcal{B}$ is the payment to broker $\mathcal{A}$. $q^M = h(q^D, v)$ is defined to objectively represent the quality of the data product which depends on both dataset quality q^D and product performance v, and $h(\cdot)$ can be instantiated according to the practical situation. Also, p^M is defined as the unit price of q^M (or, simplified as the product price). Therefore, the payment for the product can be formulated as the price times the product quality, i.e., $p^M q^M$, which corresponds to the common sense that the payment for goods is equal to the unit price multiplied by the quantity.

Profit. The profit $\Phi(\cdot)$ of buyer $\mathcal{B}$ is the difference between the quantification of the product utility and the payment to the broker as follows.

$$\Phi\left(p^M, \boldsymbol{\tau}\right) = \mathbf{U}\left(\boldsymbol{\chi}, \boldsymbol{\tau}, \nu\right) - p^M q^M. \tag{7}$$

4.1.2 Profit Function of Broker.

When broker $\mathcal{A}$ receives the data product requirements from buyer $\mathcal{B}$, she cares about her *revenue*, the payment from buyer $\mathcal{B}$ and her *cost* which consists of the compensations to sellers to buy the data and the manufacturing cost in the process of producing the data product.

Revenue. The *revenue* of broker $\mathcal{A}$ is the payment from buyer $\mathcal{B}$, i.e., $p^M q^M$ (the *cost* of buyer $\mathcal{B}$).

Cost. The *cost* of broker $\mathcal{A}$ is the sum of 1) the compensations to sellers and 2) the manufacturing cost. Broker $\mathcal{A}$ needs to pay compensations to sellers according to their provided dataset quality. The total compensation is defined as $p^D q^D$ which is the product of the total quality of the dataset p^D and the unit price p^D (or, simplified as the data price). Broker $\mathcal{A}$ also needs to consume some resources to make the product, e.g., computation resources for training models, which is measured by cost function $\mathbf{C}(\cdot)$ related to data size N and product performance ν, both significantly affecting the manufacturing cost. According to the work [9], we adopt a widely used transcendental logarithmic cost function as follows because of its adaptability to varied economies of scale and manufacturing strategy (e.g., how to allocate computing resources).

$$\begin{aligned}\mathbf{C}(N, \nu) = \exp\Bigg(&\sigma_0 + \sigma_1 \ln(N) + \sigma_2 \ln(\nu) + \frac{1}{2}\sigma_3 \ln^2(N) \\ &+ \frac{1}{2}\sigma_4 \ln^2(\nu) + \sigma_5 \ln(N) \cdot \ln(\nu)\Bigg).\end{aligned} \tag{8}$$

where $\sigma_k, k \in \{0, 1, 2, 3, 4, 5\}$ are the parameters of the translog cost function which can be fitted by broker $\mathcal{A}$ based on the actual manufacturing procedure.

Profit. The profit $\Omega(\cdot)$ of broker $\mathcal{A}$ is defined as the received payment from buyer $\mathcal{B}$ minus the manufacturing cost and the compensations to sellers as follows.

$$\Omega\left(p^M, p^D, \boldsymbol{\tau}\right) = p^M q^M - \mathbf{C}\left(N, \nu\right) - p^D q^D. \tag{9}$$

4.1.3 Profit Function of Seller.

When seller $\mathcal{S}_i$ gets the purchase request for data from broker $\mathcal{A}$, she cares about her *revenue*, the compensation from broker $\mathcal{A}$ and her *cost* coming from her privacy loss.

Revenue. The *revenue* of seller $\mathcal{S}_i$ is the compensation from broker $\mathcal{A}$, i.e., $p^D q_i^D$ (the first part of the *cost* of broker $\mathcal{A}$).

Cost. The *cost* of seller $\mathcal{S}_i$ is the privacy loss incurred based on data fidelity τ_i she provides. As discussed earlier, data fidelity is closely correlated to the data protection process, hence defined as $\tau_i = f(\epsilon_i)$ where ϵ_i is the privacy parameter in LDP as in Section 3.1. Higher ϵ_i, less noise added, leading to better fidelity τ_i. There are many alternative function forms for $f(\cdot)$, as long as it follows the diminishing trend of marginal effect. Based on Inada Conditions [25] in economics which can stipulate the marginal change trend and encourage the market equilibrium, we conclude the following characteristics that $f(\cdot)$ should satisfy.

1. The data record has fidelity $\tau_i = 0$ when $\epsilon_i = 0$ which means random noise has been added to the data.
2. Larger ϵ_i, higher τ_i due to less noise to data.
3. τ_i increases slower as ϵ_i becomes larger because very little noise is being added and it does not make a significant difference to further increase data fidelity. On the other hand, when ϵ_i is very small, i.e., noise is very large, increasing ϵ_i, can significantly increase data fidelity. Besides, τ_i cannot increase perpetually and should be upper bounded.

We choose an inverse trigonometric function form as $f(\cdot)$ and give the following definition of τ_i.

$$\tau_i = \frac{2}{\pi}\operatorname{arcsec}(\epsilon_i + 1),\ \epsilon_i \in [0, \infty), \tag{10}$$

which leads to $\tau_i \in [0, 1)$. Additionally, $\tau_i = 1$ when no noise is added. Therefore, $\tau_i \in [0, 1]$.

Bigger τ_i means better fidelity of data and more privacy loss for $\mathcal{S}_i$. We quantify such loss by function $\mathbf{L}_i(\cdot)$ which is positively related to τ_i. It's intuitive that the loss function should not only increasing but also increasing faster for higher τ_i, which corresponds to the principle of increasing marginal cost in economics. Moreover, the loss should be positively related to data quantity χ_i provided by seller $\mathcal{S}_i$. Various function forms can be applied following these principles. For simplicity in the solving process, we adopt a widely used quadratic function as follows.

$$\mathbf{L}_i\left(\tau_i\right) = \lambda_i(\chi_i\tau_i)^2, \tag{11}$$

where $\lambda_i > 0$ is $\mathcal{S}_i$'s privacy sensitivity. In our example, the privacy loss of the hospital corresponds to the negative impact of data exposure and $\mathbf{L}_i(\cdot)$ quantifies the hospital's economic estimation of the impact.

Profit. The profit of seller $\mathcal{S}_i$ is the difference between the compensation from broker $\mathcal{A}$ and the quantification of the privacy loss as follows.

$$\Psi_i\left(p^D, \tau_i\right) = p^D q_i^D - \mathbf{L}_i\left(\tau_i\right). \tag{12}$$

4.2 Market Mechanism

In our framework, the three entities take strategies in order. We present the market workflow first. Then we specify the strategies of buyer $\mathcal{B}$, broker $\mathcal{A}$, and each seller $\mathcal{S}_i$, respectively. Based on the strategies, the market mechanism is proposed.

Market Workflow. The market workflow is shown in Fig. 1. ① Buyer $\mathcal{B}$ puts forward the demand for a product including the required product performance and the corresponding indicators as well as the size of dataset for manufacturing. ② Buyer $\mathcal{B}$ determines the unit product price to buy the data product from broker $\mathcal{A}$. ③ Broker $\mathcal{A}$, acting as the bridge for the transaction between buyer $\mathcal{B}$ and m sellers, determines the unit data price to buy the data from sellers. ④ Each seller chooses what data, strictly speaking, what data fidelity to sell, and conducts corresponding privacy protection locally. ⑤ Sellers sell the protected datasets to broker $\mathcal{A}$ in exchange of the compensations. ⑥ Using the dataset bought from sellers, broker $\mathcal{A}$ manufactures the product. ⑦ Broker $\mathcal{A}$ sells the product to buyer $\mathcal{B}$. After $\mathcal{B}$ receives the product and gives payment to $\mathcal{A}$, the transaction is finished.

Buyer's Strategy. Buyer $\mathcal{B}$ makes her strategy first, which is to determine the product price p^M, in order to maximize her profit by considering the desired utility of the product and predicting the responses of the broker and sellers to p^M.

Broker's Strategy. Broker $\mathcal{A}$ takes her strategy second, which is to determine data price p^D, in order to maximize her profit by considering the given p^M from buyer $\mathcal{B}$, predicting what data fidelity sellers would provide according to p^D as well as considering the corresponding manufacturing cost.

Seller's Strategy. Sellers make their strategies last. The strategy of each seller $\mathcal{S}_i$ is to determine data fidelity τ_i, to maximize her profit by balancing the revenue of selling data and the cost of the privacy loss.

Meanwhile, inner competition among m sellers should be considered. Given the unit data price p^D, if seller $\mathcal{S}_i$ provides data with higher fidelity τ_i, more quantity would likely be sold. If other sellers provide better fidelity, less data quantity of $\mathcal{S}_i$ could be chosen. Therefore, the data quantity χ_i sold by $\mathcal{S}_i$ can be calculated according to all sellers' $\boldsymbol{\tau}$ as below.

$$\chi_i = N \frac{\omega_i \tau_i}{\sum_{j=1}^{m} \omega_j \tau_j}, \tag{13}$$

where $\omega_1, \omega_2, ..., \omega_m$ refer to the weights of sellers' data, which are maintained by the broker. Such weights reflect the historical performance of each seller's data in past deals and can therefore mirror previous buyers' influence on the current buyer to some extent. The broker would update these weights after each round of transaction. For example, new weights can be generated based on both old weights and sellers' contributions to the data product measured by Shapley value [48] as described in Section 3.2 in the last transaction.

We define such inner competition in sellers as a Nash game. Each seller $\mathcal{S}_i$ determines her strategy τ_i simultaneously to maximize her own profit which is simultaneously affected by other sellers' strategies. Nash equilibrium would be achieved where no seller can increase her profit by unilaterally changing her strategy with all other sellers' strategies fixed.

Three-Stage Stackelberg-Nash Game. Strategies of buyer $\mathcal{B}$, broker $\mathcal{A}$, and sellers $\mathcal{S}_i$ ($i = 1, 2, ..., m$) constitute the strategy profile $\langle p^M, p^D, \boldsymbol{\tau} \rangle$ of data markets. Such a profile determines market trading rules including what data (data fidelity) to sell ($\boldsymbol{\tau}$), selling at what price for both data (p^D) and data product (p^M) as well as how to select sellers (the calculated $\boldsymbol{\chi}$ based on $\boldsymbol{\tau}$). The market mechanism is formulated as a three-stage Stackelberg-Nash game, where buyer $\mathcal{B}$ is the leader, broker $\mathcal{A}$ is the sub-leader, and m sellers act as the followers. Each of them tries to maximize her own profit by determining her optimal strategy variable. The three-stage Stackelberg-Nash game is defined as follows.

Definition 4.1 (Three-Stage Stackelberg-Nash Game). The game consists of three stages for buyer $\mathcal{B}$, broker $\mathcal{A}$, and sellers $\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_m$, respectively.

Stage 1 Buyer $\mathcal{B}$: $p^{M^*} = \arg\max_{p^M} \Phi\left(p^M, \boldsymbol{\tau}\right)$.

Stage 2 Broker $\mathcal{A}$: $p^{D^*} = \arg\max_{p^D} \Omega\left(p^M, p^D, \boldsymbol{\tau}\right)$.

Stage 3 Seller $\mathcal{S}_i$: $\tau_i^* = \arg\max_{\tau_i} \Psi_i\left(p^D, \boldsymbol{\tau}\right), i = 1, 2, ..., m$.

The above three-stage Stackelberg-Nash game involves both hierarchy and simultaneity. Hierarchy indicates that a certain participant, buyer $\mathcal{B}$ in our context, has some advantage over others that enables her to act first, broker $\mathcal{A}$ takes her strategy second, and sellers make their strategies last, while simultaneity indicates the equal positions of m sellers, who take strategy simultaneously in their inner Nash game.

4.3 Market Equilibrium

In the above game, our objective is to find an optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$, by which each participant can maximize her own profit. Meanwhile, the optimal solution must satisfy some equilibrium so that no one is willing to adopt other strategies, which indicates the market stability and sustainability, making our design reasonable. We define a Stackelberg-Nash Equilibrium (SNE) in data markets as follows.

Definition 4.2 (Stackelberg-Nash Equilibrium). An optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$ constitutes a Stackelberg-Nash Equilibrium (SNE) if and only if the following set of inequalities is satisfied.

$$\Phi\left(p^{M^*}, \boldsymbol{\tau}^*\right) \geq \Phi\left(p^M, \boldsymbol{\tau}^*\right), \tag{14}$$

$$\Omega\left(p^{M^*}, p^{D^*}, \boldsymbol{\tau}^*\right) \geq \Omega\left(p^{M^*}, p^D, \boldsymbol{\tau}^*\right), \tag{15}$$

$$\Psi_i\left(p^{D^*}, \boldsymbol{\tau}^*\right) \geq \Psi_i\left(p^{D^*}, \boldsymbol{\tau}^*_{\neg i}, \tau_i\right), i = 1, 2, ..., m, \tag{16}$$

where $\boldsymbol{\tau}_{\neg i}$ means other sellers' strategies except $\mathcal{S}_i$'s, i.e., τ_j, $j \neq i$, $j = 1, 2, ..., m$.

SNE indicates that each participant takes her optimal strategy which maximizes her own profit in a buyer-leading sequence. No one can add her own profit by unilaterally changing her strategy with all other participants' strategies fixed.

5 MARKET CONSTRUCTION: EQUILIBRIUM SOLVING APPROACH AND TRADING DYNAMICS

In this section, we first derive the market equilibrium by backward induction in Section 5.1. We solve Stage 3, i.e., sellers' Nash equilibrium by two approaches, direct derivation and mean-field approximation for complex cases. Error analysis for the mean-field approximation and equilibrium analysis for SNE are also presented. Then, we describe the market dynamics in Section 5.2.

5.1 Solving Equilibrium: Backward Induction Approach

To determine the optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$, we adopt the backward induction approach [4]. We first investigate Stage 3 to solve Nash equilibrium among sellers and derive the expression of each seller's optimal strategy $\tau_i^*, i = 1, 2, ..., m$ (Eq. 20) for any given unit data price p^D in Section 5.1.1. We explore two methods, direct derivation and an approximate method using the mean-field state which can deal with complicated cases. Next, we consider Stage 2 to determine the expression of the optimal strategy p^{D^*} (Eq. 25) of broker $\mathcal{A}$ for any given unit product price p^M in Section 5.1.2. In this process, the expression of $\tau_i^*, i = 1, 2, ..., m$ solved from Nash game can be used as sellers' optimal reactions to p^D. Then, we back to Stage 1 to find the value (rather than the expression) of buyer $\mathcal{B}$'s optimal strategy p^{M^*} (Eq. 27) based on the optimal reactions of the broker as well as sellers in Section 5.1.3. After that, we can get the value of the optimal strategy p^{D^*} by substituting p^{M^*} into the result (Eq. 25) in Stage 2. Finally, we can compute the value of each seller's optimal strategy τ_i^* by substituting p^{D^*} into the result (Eq. 20) in Stage 3. Till now, the complete strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$ has been determined. The detailed deduction is presented as follows.

5.1.1 *Expression of $\boldsymbol{\tau}^*$ in Stage 3.*

We present two approaches to derive the expression of τ_i^* for sellers, i.e., the direct derivation and a mean-field based approximation method.

Direct Derivation. By substituting Eqs. 11,13 into Eq. 12 and instantiating $q_i^D = g(\chi_i, \tau_i)$ as $\chi_i \tau_i$ since q_i^D is positively correlated with χ_i and τ_i, we get each seller's profit

$$\begin{aligned}\Psi_i\left(p^D, \tau_i\right) &= p^D \chi_i \tau_i - \lambda_i (\chi_i \tau_i)^2 \\ &= p^D \cdot N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j} - \lambda_i \left(N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j}\right)^2, i = 1, 2, ..., m.\end{aligned}$$

Ψ_i is correlated to not only seller $\mathcal{S}_i$'s strategy τ_i but also other sellers' strategies $\tau_j, j \neq i$ because of the inner competition formulated as Nash game among sellers. As we discussed before, each seller aims to maximize her own profit. Therefore, we derive each of the first-order derivatives for m sellers' profit functions and let each of them equal to zero, thus getting m equations. The equation for seller $\mathcal{S}_i$ is

$$p^D \frac{\partial \left(N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j}\right)}{\partial \tau_i} - 2\lambda_i \cdot N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j} \cdot \frac{\partial \left(N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j}\right)}{\partial \tau_i} = 0. \quad (17)$$

If $\frac{\partial \left(N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j}\right)}{\partial \tau_i} = 0$, it is an all-zero solution, which does not meet our problem situation, so we can directly eliminate $\frac{\partial \left(N \frac{\omega_i \tau_i^2}{\sum_{j=1}^m \omega_j \tau_j}\right)}{\partial \tau_i}$, and then get

$$\begin{cases} p^D \sum_{i=1}^m \omega_i \tau_i - 2N\lambda_1 \omega_1 \tau_1^2 = 0 \\ p^D \sum_{i=1}^m \omega_i \tau_i - 2N\lambda_2 \omega_2 \tau_2^2 = 0 \\ \vdots \\ p^D \sum_{i=1}^m \omega_i \tau_i - 2N\lambda_m \omega_m \tau_m^2 = 0, \end{cases} \quad (18)$$

where each $\mathcal{S}_i$'s equation not only relates to her own strategy τ_i but also contains other sellers' strategies, requiring us to solve m simultaneous equations together. Finding that

$$2N\lambda_1 \omega_1 \tau_1^2 = 2N\lambda_2 \omega_2 \tau_2^2 = ... = 2N\lambda_m \omega_m \tau_m^2 = p^D \sum_{i=1}^m \omega_i \tau_i. \quad (19)$$

By adding all m equations in Eq. 18, we get

$$mp^D \sum_{i=1}^m \omega_i \tau_i - 2N \sum_{i=1}^m \lambda_i \omega_i \tau_i^2 = 0,$$

and using τ_1 to indicate other τ_i $(i = 2, 3, ..., , m)$ from Eq. 19, we get

$$mp^D \tau_1 \sum_{i=1}^m \sqrt{\frac{\lambda_1 \omega_1 \omega_i}{\lambda_i}} - 2Nm\lambda_1 \omega_1 \tau_1^2 = 0.$$

Therefore,

$$\tau_1^* = \frac{p^D}{2N\sqrt{\omega_1 \lambda_1}} \sum_{i=1}^m \sqrt{\frac{\omega_i}{\lambda_i}},$$

and using Eq. 19 again, we get all sellers' optimal strategies τ_i^* as

$${\tau_i}^* = \frac{p^D}{2N\sqrt{\omega_i \lambda_i}} \sum_{j=1}^m \sqrt{\frac{\omega_j}{\lambda_j}}, i = 1, 2, ..., m. \quad (20)$$

Note that we justify that the second-order derivative $\frac{\partial^2 \Psi_i(p^D, \tau_i)}{\partial \tau_i^2} < 0$, so these solutions can maximize each seller's profit.

Mean-field based Approximate Method. It's theoretically feasible that the optimal $\boldsymbol{\tau}$ can be derived by directly using the derivation method for each seller's profit function and then solving m simultaneous equations as above. However, for complicated function forms (e.g., more complicated loss function rather than the used quadratic one), since the number of sellers m can be quite large in practice, it may be difficult to derive analytical expressions by solving a large number of simultaneous equations each with complex forms. Specifically, the m equations are highly coupled, i.e., each with all $\tau_i, i = 1, 2, ..., m$, and eliminating the similar terms to simplify the equations as we did in Eq. 17 is not always feasible. Therefore, we propose an approximate method which makes each equation with a single τ_i and independent from others. We take a different privacy loss function form for the sellers as an example where the direct derivation is not practically feasible in order to illustrate the mean-field method. Specifically, we replace Eq. 11 with $\mathbf{L}_i(\tau_i) = \lambda_i \chi_i {\tau_i}^2$.

The approximation is based on the mean-field theory [31], which deals with situations that involve a great number of agents, i.e, sellers in our context. When there are a great number of sellers in Nash game, it's reasonable to expect that a single seller has a *tiny* (infinitesimal) influence on the equilibrium and is affected by other sellers through a mean-field state, which we formulate as $\overline{\tau}$, the weighted mean of all sellers' strategies.

$$\overline{\tau} = \frac{\sum_{i=1}^m \omega_i \tau_i}{m}. \quad (21)$$

The mean-field state $\overline{\tau}$ indicates the overall data fidelity provided by sellers at equilibrium and is not intensively affected by the data fidelity from one specific seller.

Using the new privacy loss function, the profit function of seller $\mathcal{S}_i$ in Eq. 12 is changed into

$$\Psi_i\left(p^D, \tau_i\right) = p^D (\chi_i \tau_i) - \lambda_i \chi_i {\tau_i}^2. \quad (22)$$

Using $\overline{\tau}$, χ_i can be simplified as $N \frac{\omega_i \tau_i}{m \overline{\tau}}$. Since $\overline{\tau}$ is not strongly affected by specific τ_i, we can easily derive the first-order derivative of each seller's profit function $\Psi_i\left(p^D, \tau_i\right)$ with respect to τ_i and let them equal to zero.

$$\begin{cases} p^D \cdot N\frac{\omega_1\tau_1^2}{m\overline{\tau}} - \lambda_1 \cdot N\frac{\omega_1\tau_1^3}{m\overline{\tau}} = 0 \\ p^D \cdot N\frac{\omega_2\tau_2^2}{m\overline{\tau}} - \lambda_2 \cdot N\frac{\omega_2\tau_2^3}{m\overline{\tau}} = 0 \\ \vdots \\ p^D \cdot N\frac{\omega_m\tau_m^2}{m\overline{\tau}} - \lambda_m \cdot N\frac{\omega_m\tau_m^3}{m\overline{\tau}} = 0. \end{cases}$$

We derive $\mathcal{S}_i$'s optimal strategy

$$\tau_i^* = \frac{2p^D}{3\lambda_i}, i = 1, 2, ..., m. \tag{23}$$

Note that we justify that the second-order derivative $\frac{\partial^2\Psi_i(p^D,\tau_i)}{\partial\tau_i^2} < 0$, so these solutions can maximize each seller's profit.

Error Analysis. We use fixed $\overline{\tau}$ to replace $\frac{\sum_{i=1}^m \omega_i\tau_i}{m}$ when deriving the derivatives. Such replacement is an approximation and its error depends on the form of the profit function. In this part, we analyze the error bound of the approximated mean-field approach.

Theorem 5.1. *The exact weighted mean of all sellers' strategies by the direct derivation is defined as $\overline{\tau}^{DD}$, and the approximated one by the mean-field method is $\overline{\tau}^{MF}$. The error is $\overline{\tau}^{DD} - \overline{\tau}^{MF}$. Consider the case that the privacy loss function is $\mathbf{L}_i(\tau_i) = \lambda_i\chi_i{\tau_i}^2$. When the number of sellers m is large and by scaling $\omega_1, \omega_2, ..., \omega_m$ such that $\frac{\omega_i}{\lambda_i} \leq \frac{1}{p^D m^2}$, we get*

$$-\frac{1}{6m^2} < \overline{\tau}^{DD} - \overline{\tau}^{MF} < \frac{1}{m} - \frac{2}{3m^2}.$$

Note that what makes sense is the proportional relationship among $\omega_i, i = 1, 2, ..., m$, allowing us to arbitrarily scale them.

Proof. We first calculate the upper bound of $\overline{\tau}^{DD} - \overline{\tau}^{MF}$. By applying direct derivation to Eq. 22, we can get

$$2p^D\sum_{j=1}^m \omega_j\tau_j - p^D\omega_i\tau_i = 3\lambda_i\tau_i\sum_{j=1}^m \omega_j\tau_j - \lambda_i\omega_i{\tau_i}^2.$$

By splitting $\sum_{j=1}^m \omega_j\tau_j$ into $\sum_{j=1,j\neq i}^m \omega_j\tau_j$ and $\omega_i\tau_i$, we obtain a quadratic equation about τ_i by deforming the above formula, and using root formula for the quadratic equation, we can get

$$\tau_i^* = \frac{p^D\omega_i - 3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}} + \sqrt{(3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}} - p^D\omega_i)^2 + 16p^D\lambda_i\omega_i\Sigma_{\boldsymbol{\tau}_{\neg i}}}}{4\lambda_i\omega_i}, \tag{24}$$

where $\Sigma_{\boldsymbol{\tau}_{\neg i}} = \sum_{j=1,j\neq i}^m \omega_j\tau_j$. With the constraint $\frac{\omega_i}{\lambda_i} \leq \frac{1}{p^D m^2}$, we can justify that $3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}} - p^D\omega_i > 0$ when m is very large. Thus according to $\sqrt{x+y} < \sqrt{x} + \sqrt{y}$, we can scale and deform the above formula to get

$$\omega_i\tau_i^* < \frac{\sqrt{16p^D\lambda_i\omega_i\Sigma_{\boldsymbol{\tau}_{\neg i}}}}{4\lambda_i}.$$

Further simplifying and scaling the above formula, we can get

$$\omega_i\tau_i^* < \sqrt{p^D\frac{\omega_i}{\lambda_i}\Sigma_{\boldsymbol{\tau}_{\neg i}}} \leq \sqrt{p^D\frac{\omega_i}{\lambda_i}\sum_{j=1}^m \omega_j\tau_j^*},$$

which applies to all $\tau_i^*, i = 1, 2, ..., m$. Then, by adding m inequalities together and simplifying it, we can obtain

$$\sum_{i=1}^m \omega_i\tau_i^* < \left(\sum_{i=1}^m \sqrt{p^D\frac{\omega_i}{\lambda_i}}\right)^2,$$

and thus

$$\overline{\tau}^{DD} = \frac{1}{m}\sum_{i=1}^m \omega_i\tau_i^* < \frac{1}{m}\left(\sum_{i=1}^m \sqrt{p^D\frac{\omega_i}{\lambda_i}}\right)^2.$$

Additionally, using Eqs. 21 and 23, we can derive $\overline{\tau}^{MF}$ as below.

$$\overline{\tau}^{MF} = \frac{1}{m}\sum_{i=1}^m \frac{2p^D\omega_i}{3\lambda_i}.$$

Then we use $\frac{\omega_i}{\lambda_i} \leq \frac{1}{p^D m^2}$ and get

$$\begin{aligned} \overline{\tau}^{DD} - \overline{\tau}^{MF} &< \frac{1}{m}\left(\sum_{i=1}^m \sqrt{p^D\frac{\omega_i}{\lambda_i}}\right)^2 - \frac{1}{m}\sum_{i=1}^m \frac{2p^D\omega_i}{3\lambda_i} \\ &\leq \frac{1}{m}\left(\sum_{i=1}^m \sqrt{\frac{1}{m^2}}\right)^2 - \frac{1}{m}\sum_{i=1}^m \frac{2}{3m^2} \\ &= \frac{1}{m} - \frac{2}{3m^2}. \end{aligned}$$

Next, we calculate the lower bound. Since $(3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}} - p^D\omega_i)^2 + 16p^D\lambda_i\omega_i\Sigma_{\boldsymbol{\tau}_{\neg i}} > (p^D\omega_i + 3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}})^2$, using Eq. 24 we can get

$$\begin{aligned} \overline{\tau}^{DD} &= \frac{1}{m}\sum_{i=1}^m \omega_i\tau_i^* \\ &> \frac{1}{m}\sum_{i=1}^m \frac{p^D\omega_i - 3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}} + \sqrt{(p^D\omega_i + 3\lambda_i\Sigma_{\boldsymbol{\tau}_{\neg i}})^2}}{4\lambda_i} \\ &= \frac{1}{m}\sum_{i=1}^m \frac{p^D\omega_i}{2\lambda_i}, \end{aligned}$$

and using $\frac{\omega_i}{\lambda_i} \leq \frac{1}{p^D m^2}$ again, we get

$$\begin{aligned} \overline{\tau}^{DD} - \overline{\tau}^{MF} &= \frac{1}{m}\sum_{i=1}^m \omega_i\tau_i^* - \frac{1}{m}\sum_{i=1}^m \frac{2p^D\omega_i}{3\lambda_i} \\ &> \frac{1}{m}\sum_{i=1}^m \frac{p^D\omega_i}{2\lambda_i} - \frac{1}{m}\sum_{i=1}^m \frac{2p^D\omega_i}{3\lambda_i} \geq -\frac{1}{6m^2}. \end{aligned}$$

Therefore, Theorem 5.1 holds. □

Through the above error analysis, we draw the following empirical conclusion: by scaling the value of ω_i $(i = 1, 2, ..., m)$ to satisfy certain conditions, the error of the mean-field approximation method will be bounded in an acceptable range and decrease with increasing m when m is very large.When m approaches infinity, the error is approximately zero. This result is in line with the mean-field theory. When the number of sellers m is big enough, our proposed mean-field method appears reasonable in terms of error.

5.1.2 Expression of p^{D^*} in Stage 2.

We use the direct derivation to derive the expression of p^{D^*} for the broker.

Direct Derivation. By substituting Eq. 20 into Eq. 13, we get

$$\chi_i{}^* = N\frac{\omega_i\tau_i^*}{\sum_{j=1}^m \omega_j\tau_j^*} = N\frac{\sqrt{\frac{\omega_i}{\lambda_i}}}{\sum_{j=1}^m\sqrt{\frac{\omega_j}{\lambda_j}}}.$$

Then we get

$$q^{D^*} = \sum_{i=1}^m \chi_i^*\tau_i^* = \sum_{i=1}^m \frac{p^D}{2\lambda_i}.$$

Since $q^M = h(q^D, v)$ is positively correlated to q^D and v, we instantiate $h(q^D, v)$ as $q^D v$. We get

$$q^{M^*} = q^{D^*}v = \frac{1}{2}\sum_{i=1}^m \frac{1}{\lambda_i}p^D v.$$

By substituting q^{D^*} and q^{M^*} into $\mathcal{A}$'s profit function in Eq. 9, we get

$$\Omega\left(p^M, p^D, \boldsymbol{\tau}\right) = p^M\cdot\left(\frac{1}{2}\sum_{i=1}^m\frac{1}{\lambda_i}p^D v\right) - \mathrm{C}\left(N, v\right) - p^D\cdot\left(\frac{1}{2}\sum_{i=1}^m\frac{1}{\lambda_i}p^D\right).$$

We derive the first-order derivative with respect to p^D and let it equal to 0.

$$\frac{\partial\Omega\left(p^M, p^D, \boldsymbol{\tau}\right)}{\partial p^D} = \frac{1}{2}\sum_{i=1}^m\frac{1}{\lambda_i}vp^M - \sum_{i=1}^m\frac{1}{\lambda_i}p^D = 0.$$

We can thus get the expression of p^{D^*}

$$p^{D^*} = \frac{vp^M}{2}. \tag{25}$$

We justify that the second-order derivative $\frac{\partial^2\Omega(p^M,p^D,\boldsymbol{\tau})}{\partial p^{D^2}} = -\sum_{i=1}^m\frac{1}{\lambda_i} < 0$, so the solution can maximize the broker's profit.

5.1.3 Value of p^{M^*} in Stage 1.

We also use direct derivation in this stage, and by using the results in Sections 5.1.1 and 5.1.2 we can directly derive the value rather than the expression of p^{M^*} for the buyer.

Direct Derivation. By substituting Eq. 20 and Eq. 25 into $\mathcal{B}$'s profit function in Eq. 7, we can obtain the profit of $\mathcal{B}$

$$\begin{aligned}\Phi\left(p^M, \boldsymbol{\tau}\right) &= \theta_1\ln\left(1+\rho_1 q^{D^*}\right) + \theta_2\ln\left(1+\rho_2 v\right) - p^M q^{M^*}\\ &= \theta_1\ln\left(1+\mathtt{c}_1 p^M\right) + \theta_2\ln\left(1+\rho_2 v\right) - \frac{\mathtt{c}_2\theta_1}{2}p^{M^2},\end{aligned}$$

where $\mathtt{c}_1 = \frac{\rho_1 v}{4}\sum_{i=1}^m\frac{1}{\lambda_i}$ and $\mathtt{c}_2 = \frac{v^2}{2\theta_1}\sum_{i=1}^m\frac{1}{\lambda_i}$.

Then, we derive the first-order derivative of $\Phi\left(p^M, \boldsymbol{\tau}\right)$ as follows.

$$\frac{\partial\Phi(p^M, \boldsymbol{\tau})}{\partial p^M} = \frac{\theta_1\mathtt{c}_1}{1+\mathtt{c}_1 p^M} - \mathtt{c}_2\theta_1 p^M. \tag{26}$$

By letting $\frac{\partial\Phi(p^M,\boldsymbol{\tau})}{\partial p^M}$ in Eq. 26 equal to zero, we obtain

$$\mathtt{c}_1\mathtt{c}_2\cdot p^{M^2} + \mathtt{c}_2\cdot p^M - \mathtt{c}_1 = 0.$$

Using characteristic root method, we find buyer $\mathcal{B}$'s optimal strategy p^{M^*} (after discarding the negative solution)

$$p^{M^*} = \frac{-\mathtt{c}_2 + \sqrt{\mathtt{c}_2^2 + 4\mathtt{c}_1^2\mathtt{c}_2}}{2\mathtt{c}_1\mathtt{c}_2}. \tag{27}$$

We justify that the second-order derivative $\frac{\partial^2\Phi(p^M,\boldsymbol{\tau})}{\partial p^{M^2}} = -\frac{\theta_1\mathtt{c}_1^2}{(1+\mathtt{c}_1 p^M)^2} - \theta_1\mathtt{c}_2 < 0$, so the solution can maximize the buyer's profit.

After that, we can determine the optimal value of p^{D^*} by substituting p^{M^*} into Eq. 25 as well as each seller's optimal value of τ_i^* by substituting p^{D^*} into Eq. 20. Till now, the complete optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^*\right\rangle$ has been determined, based on which the market transaction can be conducted.

5.1.4 Equilibrium Analysis.

In this part, we prove the existence and uniqueness of SNE in *Share.*

Theorem 5.2. *The complete optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^*\right\rangle$ determined by backward induction approach uniquely constitutes SNE.*

Proof.

Existence. For the buyer, when the broker and sellers hold the optimal strategy expressions in Eq. 25 and Eq. 20, the buyer's profit $\Phi\left(p^M, \boldsymbol{\tau}^*\right)$ only changes with p^M. In the process of deriving optimal p^{M^*}, the first-order derivation is set to be 0 and the second-order is strictly less than 0, which means that the maximum profit is obtained at p^{M^*}. Thus, Eq. 14 holds at p^{M^*}. For the broker, when the buyer and sellers hold the optimal strategies p^{M^*} in Eq. 27 and $\boldsymbol{\tau}^*$ in Eq. 20, the broker's optimal profit can be obtained at p^{D^*} since the profit function is strictly concave and has a single extreme point p^{D^*}. Thus, Eq. 15 holds at p^{D^*}. For the sellers, each seller determines each τ_i^* simultaneously in the same way, and $\tau_i^*, i = 1, 2, ..., m$ are jointly decided. For each seller $\mathcal{S}_i$, her optimal strategy τ_i^* is determined by letting the first-order derivation equal to 0. Since the profit function is strictly concave, the extreme point τ_i^* maximizes seller $\mathcal{S}_i$'s profit if $\tau_i^* \leq 1$. Otherwise, when the extreme point is larger than 1, the optimal value $\tau_i^* = 1$ can also maximize $\mathcal{S}_i$'s profit since the profit function is monotonically increasing in the feasible range of τ_i and maximized at the right endpoint 1. Thus, Eq. 16 holds at τ_i^*. Therefore, it's proved that SNE exists in our mechanism.

Uniqueness. For the buyer, since her profit function is strictly concave, the maximum profit is obtained only at p^{M^*}. Any other value of $p^M \neq p^{M^*}$ will yield an inferior profit. Such result can be also explained by Convex Optimization [6], i.e., the strategy space of p^M is a convex and compact subspace of Euclidean space, and the profit function $\Phi\left(\cdot\right)$ is a convex function of p^M, leading to the unique optimal p^{M^*} that maximizes $\Phi\left(\cdot\right)$. Thus, Eq. 14 holds only at p^{M^*}. For the broker, her profit function is also strictly concave and only has a single extreme point p^{D^*}. Any other value of $p^D \neq p^{D^*}$ will lower the broker's profit. Thus, Eq. 15 holds only at p^{D^*}. For sellers, seller $\mathcal{S}_i$ can only have lower profit by deciding

$\forall \tau_i \neq \tau_i^*$. If $\tau_i^* \leq 1$, seller $\mathcal{S}_i$'s profit function is concave and only maximized at τ_i^*. Otherwise, the profit can only be maximized at the right endpoint, $\tau_i^* = 1$, since the profit function is monotonically increasing. For each seller, if she chooses other $\forall \tau_i \neq \tau_i^*$, she can only have lower profit with $\boldsymbol{\tau}_{\neg i}^*$ kept the same. Thus, the unique Nash equilibrium among sellers is achieved and Eq. 16 holds only at τ_i^*. Therefore, it's proved that other strategy profile except our solution cannot satisfy SNE, which infers the uniqueness of SNE in our mechanism.

□

5.2 Complete Data Trading Dynamics

We summarize the complete dynamics of data markets in Algorithm 1, which integrates the equilibrium solving process in Section 5.1.

The first phase is *Parameter Collection*. We assume that each party can give their specific input parameters. The buyer sets appropriate parameters $\theta_1, \theta_2, \rho_1, \rho_2$ for her utility function and proposes demand parameters ν and N for the product (Line 2). Note that the form of the product is not restricted and can range from simple data aggregation to deep learning models. The broker determines $\sigma_k, k \in \{0, 1, 2, 3, 4, 5\}$ for her translog cost function and maintains the weights $\omega_i, i = 1, 2, ..., m$ of sellers' datasets (Line 3). Specifically, the weights are initialized to be the same when the market is established. To decide the real weights before the first transaction, the broker can use dummy buyers to iterate several times, and in each iteration, Shapley values of sellers' datasets are calculated after production and can be used to update weights in the next iteration. Sellers give their privacy sensitivity $\lambda_i, i = 1, 2, ..., m$ (Line 4). We assume that participants provide their truthful parameters in line with the practical situation under the supervision of market regulators (e.g., by regular spot-check). Plus, m is fixed according to the number of sellers in practice (Line 5).

The second phase is *Strategy Decision*. Using the strategy mechanism, buyer $\mathcal{B}$, broker $\mathcal{A}$, and sellers $\mathcal{S}_i, i = 1, 2, ..., m$ give strategies of the product price p^{M^*}, data price p^{D^*}, and data fidelity $\boldsymbol{\tau}_i^*$ in order according to Eqs. 27, 25, 20, respectively (Line 7).

Then *Data Transaction* between the broker and sellers begins. The data quantity chosen from each seller can be calculated according to Eq. 13 (Line 9). Next, each seller picks χ_i^*-sized dataset (Line 11) and pre-processes it for privacy protection based on $\epsilon_i{}^*$ calculated from Eq. 10 (Lines 12-13). After that, seller $\mathcal{S}_i$ gives her protected dataset D_i^t to the broker in exchange of the compensation $p^{D^*} q_i^{D^*}$ (Line 14).

Next phase is *Product Production*. The broker collects the data as D^t and uses it to make the product (Line 16). Moreover, the weights of sellers' datasets are updated by the broker based on their corresponding contributions to the data product (Line 17). We give one update formula based on Shapley value as an example: $\omega_i' = 0.2\omega_i + 0.8\mathcal{SV}_i, i = 1, 2, ..., m$ where $\mathcal{SV}_i$ is the Shapley value of D_i^t to the product, and the updated weights ω_i' can be used in the subsequent transaction.

The last phase is *Product Transaction* between the broker and the buyer. The broker gives the product to the buyer and the buyer pays $p^{M^*} q^{M^*}$ to the broker (Line 19). So far, the current round of data trading among buyer $\mathcal{B}$, broker $\mathcal{A}$, and sellers $\mathcal{S}_i, i = 1, 2, ..., m$ has finished. When the next buyer comes, the next round of data trading will start and can use the updated $\omega_i', i = 1, 2, ..., m$.

Algorithm 1: Data trading dynamics.

1 %% Parameter Collection;
2 From the current buyer $\mathcal{B}$, parameters $N, \nu, \theta_1, \theta_2, \rho_1, \rho_2$ are provided;
3 From broker $\mathcal{A}$, $\sigma_k(k \in \{0, 1, 2, 3, 4, 5\}), \omega_i$ $(i = 1, 2, ..., m)$ are given;
4 From existing m sellers, each seller $\mathcal{S}_i$ decides λ_i;
5 m is given according to the pratical situation;
6 %% Strategy Decision;
7 Through three-stage Stackelberg-Nash game, the optimal strategy profile $\left\langle p^{M^*}, p^{D^*}, \boldsymbol{\tau}^* \right\rangle$ is determined by the buyer, the broker, and sellers, respectively;
8 %% Data Transaction;
9 The quantity of data each seller can sell, i.e., $\boldsymbol{\chi}^*$, is calculated according to Eq. 13;
10 **for** *each seller* $\mathcal{S}_i, i = 1, 2, ...m$ **do**
11   Randomly pick $\chi_i{}^*$ data pieces from her dataset D_i;
12   Calculate $\epsilon_i{}^*$ from the strategy $\tau_i{}^*$ according to Eq. 10;
13   Conduct LDP with $\epsilon_i{}^*$ on her $\chi_i{}^*$-sized dataset, and then give the protected D_i^t to broker $\mathcal{A}$;
14 Broker $\mathcal{A}$ gets data from sellers to form dataset D^t for production and pays compensation $p^{D^*} q_i^{D^*}$ to each seller;
15 %% Product Production;
16 Broker $\mathcal{A}$ then uses D^t to produce the data product;
17 After manufacturing the product, broker $\mathcal{A}$ updates $\omega_1, \omega_2, ..., \omega_m$ (might scale down proportionally as needed) based on the contribution to the product from each seller's D_i^t;
18 %% Product Transaction;
19 Broker $\mathcal{A}$ gives the product to buyer $\mathcal{B}$ and meantime $\mathcal{B}$ pays $p^{M^*} q^{M^*}$ to $\mathcal{A}$.

Time Complexity. As seen from Algorithm 1, the phase of *Parameter Collection* costs $O(m)$ since m sellers need to provide $\lambda_i, i = 1, 2, ..., m$. *Strategy Decision* costs $O(m)$ based on the optimal strategy profile. *Data Transaction* costs $O(m + N)$ because each seller needs to form and protect her χ_i-sized dataset, and N data records in total are preprocessed and sold to the broker. The time cost of *Product Production* depends on the exact product type, production mode, and the way of updating the weights $\omega_1, \omega_2, ..., \omega_m$. The last phase of *Product Transaction* takes constant time. Therefore, the complexity of data trading algorithm excluding *Product Production* is $O(m + N)$.

6 EXPERIMENTS

In this section, we present experimental studies validating the effectiveness and efficiency of *Share*. We first describe our experiment setup including the datasets and parameter settings in Section 6.1. Sections 6.2 and 6.3 show the results verifying the effectiveness and efficiency of *Share*, respectively. Section 6.4 shows the effect of parameters used in *Share*.

6.1 Experiment Setup

We conduct experiments on a machine with an Intel Core i7-11700KF running Ubuntu with 64GB memory. We choose Linear Regression model as the data product.

Datasets. We conduct extensive experiments on a real dataset of a Combined Cycle Power Plant (CCPP) [14]. The dataset contains 9,568 data points, each having four features. The Linear Regression task is to predict the net hourly electrical energy output of the plant. Besides the real dataset, we augment CCPP by replicating 100 times and then adding Gaussian noise $\mathcal{N}(0, 0.1^2)$ to generate a synthetic dataset with the size of $1,000,000$ to test the efficiency of *Share*.

Parameter Settings. Our parameters include the number of sellers m, model requirements demanded by the buyer, i.e., the needed total data quantity N and the model performance v which refers to the explained variance of the model, and individual parameters of each party, i.e., buyer $\mathcal{B}$'s $\theta_1, \theta_2, \rho_1, \rho_2$ related to model utility, broker $\mathcal{A}$'s cost parameters $\sigma_k, k \in \{0, 1, 2, 3, 4, 5\}$, and seller $\mathcal{S}_i$'s privacy sensitivity $\lambda_i, i = 1, 2, ..., m$. Without loss of generality, we set the buyer's requirement parameters as $N = 500, v = 0.8$ and utility parameters as $\theta_1 = 0.5, \theta_2 = 0.5, \rho_1 = 0.5, \rho_2 = 250$ (in order to balance the impacts of the product quality and the dataset quality). The cost parameters of the broker are related to the practical manufacturing situation and are set as default values $\sigma_0 = 1 \times 10^{-3}, \sigma_1 = -2, \sigma_2 = -3, \sigma_3 = 1 \times 10^{-3}, \sigma_4 = 2 \times 10^{-3}, \sigma_5 = 1 \times 10^{-3}$. We set $m = 100$, and sellers' $\lambda_i, i = 1, 2, ..., m$ are picked randomly in $(0, 1)$.

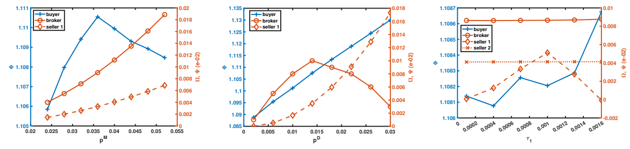
We assume each seller owns a dataset of the same size and distribute 9,000 data records of the CCPP dataset (the remaining data is used for test) equally to 100 sellers. In the real world, the datasets of sellers vary in quality. To simulate this characteristic, we first sort data by quality measured by Shapley value, which indicates the contribution of each data record to model training. The Shapley value is calculated based on Monte Carlo Method [7, 19] with the permutation number set as 100. Then by distributing data in decreasing quality over sellers, each seller owns 90 data records with different quality. Laplace mechanism [16] is used to ensure local differential privacy for each seller.

$\omega_1, \omega_2, ..., \omega_m$ are generated by using buyer $\mathcal{B}$ as the dummy buyer to iterate the mechanism which takes five times to stabilize the profits. We consider buyer $\mathcal{B}$ as a general buyer coming after several transactions have finished. Shapley values of sellers' datasets can be calculated after model training to update weights for the next transaction.

We will show the results of using the direct derivation in the mechanism. Note that the mean-field approach used when the direct derivation fails functions similarly to the markets in terms of the effectiveness, efficiency, and parameter influence.

6.2 Effectiveness

We implement the mechanism and unilaterally change the strategies of the buyer, the broker, and sellers respectively to verify the profit maximization of all parties as well as the corresponding equilibrium state.



(a) Vary $\mathcal{B}$'s strategy (b) Vary $\mathcal{A}$'s strategy (c) Vary $\mathcal{S}_1$'s strategy

Figure 2: Effectiveness.

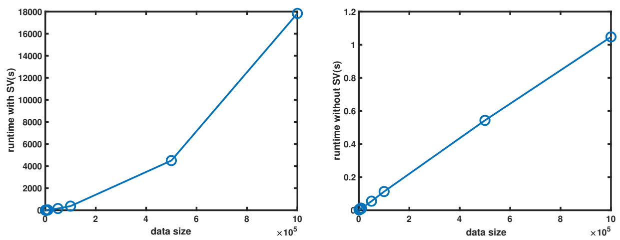
Fig.2(a) shows the results of profits when we change p^M around the optimal strategy p^{M*} while maintaining the rest. Seller $\mathcal{S}_1$ acts as a representative of sellers. It's found that the peak of the buyer's profit $\Phi(\cdot)$ appears when her optimal strategy $p^{M*} = 0.036$ determined in SNE is adopted. Whatever strategy the buyer chooses except p^{M*}, she will get lower profit when all other participants' strategies are fixed. The change of the profits of the broker and the seller is intuitive. Specifically, with growing p^M, the broker can gain more profit, which can further add the compensations for sellers and make their profits higher as well.

Fig.2(b) shows the results of profits when we change p^D around the optimal strategy $p^{D*} = 0.014$ while maintaining the rest. Similarly, it is found that the broker cannot increase her profit by unilaterally changing her strategy. The change of the profits of the buyer and seller is also intuitive. Specifically, the growing p^D brings more compensations to sellers, adding their profits. Due to more compensations, the dataset quality from sellers can therefore be improved, which causes the rise of the buyer's profit.

Fig.2(c) shows the results of profits when we change τ_1 around the optimal strategy $\tau_1^* = 0.001$ while maintaining the rest. Note that the first two sellers $\mathcal{S}_1$ and $\mathcal{S}_2$ are chosen as representatives. It is the same that the seller who changes her strategy unilaterally gets no more profit. Even if one seller changes her strategy, the broker can nearly keep her profit as before, benefiting from the *inner* competition among sellers formulated as Nash game. Specifically, the effect of sellers' bounded rationality is almost limited among sellers and is corrected automatically in Stage 3, which signifies the transparency of Stage 3 to upper stages. The change of the buyer's profit may be due to the effect of data on the model, which is not always predictable, causing the irregular curve of $\Phi(\cdot)$. In theory, varying τ_1 surely makes differences on other sellers. However, since the number of sellers is large, this effect is *diluted* and negligible, making the profit of $\mathcal{S}_2$ almost unchanged.

6.3 Efficiency

Fig.3(a) and Fig.3(b) show the runtime of the proposed data trading algorithm with and without Shapley value to update weights, respectively. We use the synthetic dataset with $1,000,000$ data records and adjust the number of sellers m from 5 to $10,000$ while fixing the other parameters and the average number of data records chosen from each seller as 100. Fig.3(a) shows that the runtime grows as m goes higher but with an acceptable rate. Even when $m = 10,000$, it does not take too much time. Note that our mechanism contains an extremely time-consuming part to calculate Shapley values. Fig.3(b) shows that our mechanism without Shapley value calculation can run very fast with a linear time complexity, which corresponds to the complexity analysis of Algorithm 1 in Section 5.2.



(a) Runtime with $\mathcal{SV}$ (b) Runtime without $\mathcal{SV}$

Figure 3: Efficiency.

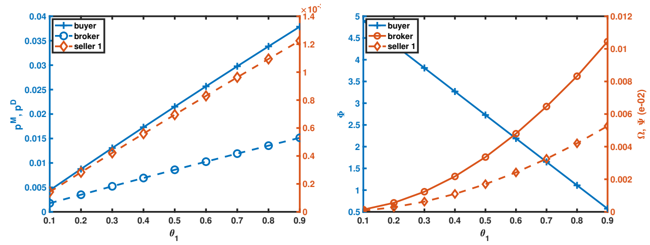
6.4 Parameter Influence

In this section, we make sensitivity analyses on the parameters in our mechanism and investigate how the parameters affect the strategies and profits of three parties.

Fig.4(a) and Fig.4(b) present the effect of θ_1 on strategies and profits, respectively. Note that θ_1 and θ_2 refer to the buyer's relative concerns on the dataset quality and the product performance, respectively. We alter θ_1 from 0.1 to 0.9 (θ_2 from 0.9 to 0.1 accordingly) and fix other parameters. Fig.4(a) shows that all the strategies boost in a linear rate with increasing θ_1, which is intuitive since more concern on data leads to higher prices and better data. Fig.4(b) shows that the profit of the buyer decreases in a linear trend, and the profits of other two parties increase with increasing θ_1. The change of the buyer's profit indicates the difference between the significance of product performance and dataset quality to the product utility in different tasks, and the descent of $\Phi(\cdot)$ here suggests that in this task the product plays a more important role in the buyer's profit than the dataset quality.

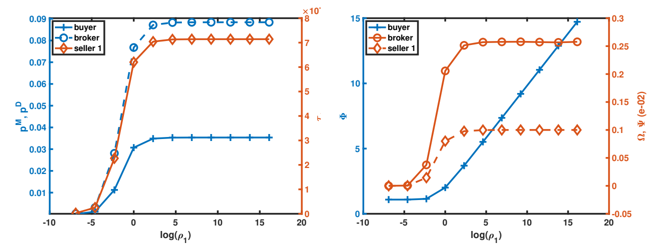
Fig.5(a) and Fig.5(b) present the effect of ρ_1 on strategies and profits, respectively. Note that ρ_1 is a parameter relevant to the buyer's sensitivity to the dataset quality, which objectively reflects the relationship between the dataset quality and the product utility. Fig.5(a) shows that too small of ρ_1 can hardly lead to effective markets because of the buyer's indifference on data. When ρ_1 reaches a certain level, all the strategies stay the same and the market reaches the equilibrium. The influence of ρ_1 is limited within the utility for the buyer and can no longer disturb the market equilibrium, which may be due to the common sense that the dataset quality cannot increase unlimitedly and even with sharper sensitivity to data, higher prices wouldn't bring about better data any more. Fig.5(b) shows that the profit of the buyer surges as ρ_1 increases because she will get more utility from the raise of the dataset quality. When ρ_1 is big enough, the increase of ρ_1 has little effect on the profits of the broker and sellers, which can be explained by the trend of strategies discussed above.

Fig.6(a) and Fig.6(b) present the effect of ρ_2 on strategies and profits, respectively. Note that ρ_2 is related to the sensitivity of the buyer to the product performance. Fig.6(a) shows that ρ_2 influences little on the strategies. Fig.6(b) shows that the buyer gains more as ρ_2 is raised due to the same reason in the above analysis about ρ_1. The profits of the broker and sellers stay unchanged because the



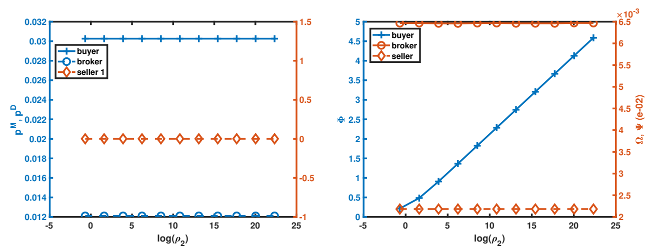
(a) Strategy vs. θ_1 (b) Profit vs. θ_1

Figure 4: Effect of θ_1.



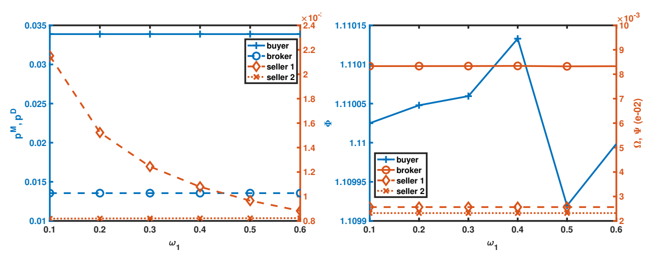
(a) Strategy vs. ρ_1 (b) Profit vs. ρ_1

Figure 5: Effect of ρ_1.



(a) Strategy vs. ρ_2 (b) Profit vs. ρ_2

Figure 6: Effect of ρ_2.

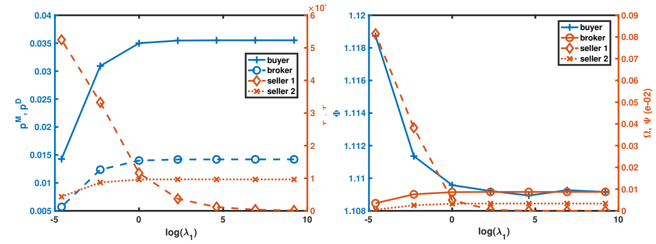


(a) Strategy vs. ω_1 (b) Profit vs. ω_1

Figure 7: Effect of ω_1.

broker still aims to achieve the buyer's fixed demanding v in the production and sellers only deal with the data not the product.

Fig.7(a) and Fig.7(b) present the effect of ω_1 on strategies and profits, respectively. Note that $\omega_1, \omega_2, ..., \omega_m$ are the weights of sellers' datasets and assess the sellers' data in previous transactions. We select $\mathcal{S}_1$ and $\mathcal{S}_2$ as representatives. Fig.7(a) shows that ω_1 only affects the strategy of the corresponding seller $\mathcal{S}_1$. The strategies of the buyer and the broker remain the same because ω_1 only affects



(a) Strategy vs. λ_1 (b) Profit vs. λ_1

Figure 8: Effect of λ_1.

the inner competition among sellers. Since the number of sellers is large, varying ω_1 makes little difference on other sellers, making the strategy of $\mathcal{S}_2$ almost unchanged. Fig.7(b) shows that when ω_1 varies from 0.1 to 0.6, all profits except the buyer's are stable. Once ω_1 gets a non-appropriate value, the data of this seller $\mathcal{S}_1$ won't work as expected and affects the profit of the buyer, leading to the unsmooth curve of $\Phi(\cdot)$.

Fig.8(a) and Fig.8(b) show the effect of $\mathcal{S}_1$'s parameter λ_1 on strategies and profits, respectively. Note that λ_i is related to seller $\mathcal{S}_i$'s privacy sensitivity. Fig.8(a) shows that τ_1 sinks with increasing λ_1 since $\mathcal{S}_1$ will strengthen her data protection if more sensitive to privacy risks. p^M and p^D increase possibly because higher prices may be provided to encourage conservative sellers to offer high-fidelity data in spite of heavy privacy risks. Fig.8(b) shows that λ_1 mainly influences the buyer's and the corresponding seller $\mathcal{S}_1$'s profits. The profit of $\mathcal{S}_1$ decreases because bigger λ_1, more privacy loss $\mathcal{S}_1$ will suffer. The profit of buyer $\mathcal{B}$ dives probably because the seller would enhance the protection on her data when faced with huge privacy risks, thus lowering the data fidelity and further harming the buyer's profit. The profit of the broker remains unchanged because the broker herself does not rely on data fidelity but just *transfers* data from the sellers to the buyer.

7 CONCLUSION AND FUTURE WORK

We presented the first Stackelberg-Nash game based data markets *Share*. A three-stage Stackelberg-Nash game is proposed to model the data trading mechanism in the buyer-leading multi-seller data markets with absolute pricing rules. The mutual interaction among three parties (buyers, brokers, and sellers) is considered as a three-stage Stackelberg game, which maximizes the profits of all participants. The inner competition among sellers is considered as a Nash game, which neatly solves the seller selection problem. To derive equilibrium, the backward induction approach is used. Specifically, to solve the inner Nash game, we proposed two methods, the direct derivation and a novel mean-field approximation Our proposed data market framework performs well on the real and synthetic datasets in terms of both effectiveness and efficiency.

Our data market framework can be easily adapted to a variety of market settings, e.g., broker-leading instead of buyer-leading. Thus we believe the market dynamics described in this work can be a natural and scalable way for data trading. There are also interesting and practical issues to address for deployment as future work. For example, the deficiency of real-world historical trading records brings about the challenge of parameter fitting for each party.

REFERENCES

[1] Anish Agarwal, Munther Dahleh, and Tuhin Sarkar. 2019. A marketplace for data: An algorithmic solution. In *Proceedings of the 2019 ACM Conference on Economics and Computation*. ACM, 701–726.
[2] Baoyi An, Mingjun Xiao, An Liu, Xike Xie, and Xiaofang Zhou. 2021. Crowdsensing Data Trading based on Combinatorial Multi-Armed Bandit and Stackelberg Game. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*. IEEE, 253–264.
[3] Gaurang Bansal and Biplab Sikdar. 2021. Security Service Pricing Model for UAV Swarms: A Stackelberg Game Approach. In *2021 IEEE Conference on Computer Communications Workshops, INFOCOM Workshops 2021, Vancouver, BC, Canada, May 10-13, 2021*. IEEE, 1–6. https://doi.org/10.1109/INFOCOMWKSHPS51825.2021.9484577
[4] Tamer Başar and Geert Jan Olsder. 1998. *Dynamic noncooperative game theory*. SIAM.
[5] Bloomberg. 1981. https://www.bloomberg.com/professional/product/market-data/.
[6] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.
[7] Javier Castro, Daniel Gómez, and Juan Tejada. 2009. Polynomial calculation of the Shapley value based on sampling. *Computers & OR* 36, 5 (2009), 1726–1730. https://doi.org/10.1016/j.cor.2008.04.004
[8] Lingjiao Chen, Paraschos Koutris, and Arun Kumar. 2019. Towards Model-based Pricing for Machine Learning in a Data Marketplace. In *SIGMOD*, Peter A. Boncz, Stefan Manegold, Anastasia Ailamaki, Amol Deshpande, and Tim Kraska (Eds.). ACM, 1535–1552. https://doi.org/10.1145/3299869.3300078
[9] Laurits R Christensen, Dale W Jorgenson, and Lawrence J Lau. 1975. Transcendental logarithmic utility functions. *The American Economic Review* 65, 3 (1975), 367–383.
[10] Zicun Cong, Xuan Luo, Jian Pei, Feida Zhu, and Yong Zhang. 2022. Data pricing in machine learning pipelines. *Knowl. Inf. Syst.* 64, 6 (2022), 1417–1455. https://doi.org/10.1007/s10115-022-01679-4
[11] Vincent Conitzer and Tuomas Sandholm. 2003. Complexity Results about Nash Equilibria. In *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, Georg Gottlob and Toby Walsh (Eds.). Morgan Kaufmann, 765–771. http://ijcai.org/Proceedings/03/Papers/111.pdf
[12] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. 2009. The complexity of computing a Nash equilibrium. *SIAM J. Comput.* 39, 1 (2009), 195–259.
[13] DAWEX. 2015. https://www.dawex.com/en/.
[14] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml
[15] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. 2013. Local Privacy and Statistical Minimax Rates. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*. IEEE Computer Society, 429–438. https://doi.org/10.1109/FOCS.2013.53
[16] Cynthia Dwork. 2006. Differential Privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 4052)*, Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener (Eds.). Springer, 1–12. https://doi.org/10.1007/11787006_1
[17] Cynthia Dwork and Jing Lei. 2009. Differential privacy and robust statistics. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, Michael Mitzenmacher (Ed.). ACM, 371–380. https://doi.org/10.1145/1536414.1536466
[18] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* 9, 3-4 (2014), 211–407. https://doi.org/10.1561/0400000042
[19] S. Shaheen Fatima, Michael J. Wooldridge, and Nicholas R. Jennings. 2008. A linear approximation method for the Shapley value. *Artif. Intell.* 172, 14 (2008), 1673–1699. https://doi.org/10.1016/j.artint.2008.05.003
[20] Raul Castro Fernandez. 2022. Protecting Data Markets from Strategic Buyers. In *SIGMOD '22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 - 17, 2022*, Zachary Ives, Angela Bonifati, and Amr El Abbadi (Eds.). ACM, 1755–1769. https://doi.org/10.1145/3514221.3517855
[21] D. Gale and L. S. Shapley. 1962. College Admissions and the Stability of Marriage. *The American Mathematical Monthly* 69, 1 (1962), 9–15. http://www.jstor.org/stable/2312726
[22] Hui Gao, Chi Harold Liu, Jian Tang, Dejun Yang, Pan Hui, and Wendong Wang. 2019. Online Quality-Aware Incentive Mechanism for Mobile Crowd Sensing with Extra Bonus. *IEEE Trans. Mob. Comput.* 18, 11 (2019), 2589–2603. https://doi.org/10.1109/TMC.2018.2877459
[23] Amirata Ghorbani and James Zou. 2019. Data shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning*. PMLR, 2242–2251.

[24] John C Harsanyi. 1967. Games with incomplete information played by "Bayesian" players, I–III Part I. The basic model. *Management science* 14, 3 (1967), 159–182.

[25] Ken-Ichi Inada. 1963. On a two-sector model of economic growth: Comments and a generalization. *The Review of Economic Studies* 30, 2 (1963), 119–127.

[26] Chandra K. Jaggi, Mamta Gupta, Amrina Kausar, and Sunil Tiwari. 2019. Inventory and credit decisions for deteriorating items with displayed stock dependent demand in two-echelon supply chain using Stackelberg and Nash equilibrium solution. *Ann. Oper. Res.* 274, 1-2 (2019), 309–329. https://doi.org/10.1007/s10479-018-2925-9

[27] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nezihe Merve Gurel, Bo Li, Ce Zhang, Costas Spanos, and Dawn Song. 2019. Efficient task-specific data valuation for nearest neighbor algorithms. *Proceedings of the VLDB Endowment* 12, 11 (2019), 1610–1623.

[28] Changkun Jiang, Lin Gao, Lingjie Duan, and Jianwei Huang. 2018. Data-Centric Mobile Crowdsensing. *IEEE Trans. Mob. Comput.* 17, 6 (2018), 1275–1288. https://doi.org/10.1109/TMC.2017.2763956

[29] Paraschos Koutris, Prasang Upadhyaya, Magdalena Balazinska, Bill Howe, and Dan Suciu. 2012. Query-based data pricing. In *PODS*. ACM, 167–178.

[30] Paraschos Koutris, Prasang Upadhyaya, Magdalena Balazinska, Bill Howe, and Dan Suciu. 2013. Toward practical query pricing with QueryMarket. In *SIGMOD*. ACM, 613–624.

[31] Jean-Michel Lasry and Pierre-Louis Lions. 2007. Mean field games. *Japanese journal of mathematics* 2, 1 (2007), 229–260.

[32] Chao Li, Daniel Yang Li, Gerome Miklau, and Dan Suciu. 2014. A Theory of Pricing Private Data. *ACM Trans. Database Syst.* 39, 4 (2014), 34:1–34:28. https://doi.org/10.1145/2691190.2691191

[33] Henger Li, Wen Shen, and Zizhan Zheng. 2020. Spatial-Temporal Moving Target Defense: A Markov Stackelberg Game Model. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020*, Amal El Fallah Seghrouchni, Gita Sukthankar, Bo An, and Neil Yorke-Smith (Eds.). International Foundation for Autonomous Agents and Multiagent Systems, 717–725. https://dl.acm.org/doi/abs/10.5555/3398761.3398847

[34] Man Li, Jiahu Qin, Qichao Ma, Wei Xing Zheng, and Yu Kang. 2021. Hierarchical Optimal Synchronization for Linear Systems via Reinforcement Learning: A Stackelberg-Nash Game Perspective. *IEEE Trans. Neural Networks Learn. Syst.* 32, 4 (2021), 1600–1611. https://doi.org/10.1109/TNNLS.2020.2985738

[35] Jinfei Liu, Qiongqiong Lin, Jiayao Zhang, Kui Ren, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. 2021. Demonstration of Dealer: An End-to-End Model Marketplace with Differential Privacy. *Proc. VLDB Endow.* 14, 12 (2021), 2747–2750. http://www.vldb.org/pvldb/vol14/p2747-zhang.pdf

[36] Jinfei Liu, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. 2021. Dealer: An End-to-End Model Marketplace with Differential Privacy. *Proc. VLDB Endow.* 14, 6 (2021), 957–969. http://www.vldb.org/pvldb/vol14/p957-liu.pdf

[37] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*. IEEE Computer Society, 94–103. https://doi.org/10.1109/FOCS.2007.41

[38] Oskar Morgenstern and John Von Neumann. 1953. *Theory of games and economic behavior*. Princeton university press.

[39] Anna Nagurney and Pritha Dutta. 2019. Supply chain network competition among blood service organizations: a Generalized Nash Equilibrium framework. *Ann. Oper. Res.* 275, 2 (2019), 551–586. https://doi.org/10.1007/s10479-018-3029-2

[40] John Nash. 1951. Non-cooperative games. *Annals of mathematics* (1951), 286–295.

[41] John F Nash Jr. 1950. Equilibrium points in n-person games. *Proceedings of the national academy of sciences* 36, 1 (1950), 48–49.

[42] Chaoyue Niu, Zhenzhe Zheng, Fan Wu, Shaojie Tang, Xiaofeng Gao, and Guihai Chen. 2018. Unlocking the Value of Privacy: Trading Aggregate Statistics over Private Correlated Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, Yike Guo and Faisal Farooq (Eds.). ACM, 2031–2040. https://doi.org/10.1145/3219819.3220013

[43] Jian Pei. 2021. A Survey on Data Pricing: from Economics to Data Science. *IEEE Trans. Knowl. Data Eng.* (2021). https://doi.org/10.1109/TKDE.2020.3045927

[44] Jian Pei, Feida Zhu, Zicun Cong, Xuan Luo, Huiwen Liu, and Xin Mu. 2021. Data Pricing and Data Asset Governance in the AI Era. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, Feida Zhu, Beng Chin Ooi, and Chunyan Miao (Eds.). ACM, 4058–4059. https://doi.org/10.1145/3447548.3470818

[45] Alvin E. Roth. 2016. Lloyd Shapley *(1923-2016)*. *Nat.* 532, 7598 (2016), 178. https://doi.org/10.1038/532178a

[46] SafeGraph. 2016. https://www.safegraph.com/.

[47] Reinhard Selten. 1965. Spieltheoretische behandlung eines oligopolmodells mit nachfrageträgheit. *Journal of Institutional and Theoretical Economics*, H. 2 (1965), 301–324.

[48] Lloyd S Shapley. 1953. A value for n-person games. *Contributions to the Theory of Games* 2, 28 (1953), 307–317.

[49] Arunesh Sinha, Fei Fang, Bo An, Christopher Kiekintveld, and Milind Tambe. 2018. Stackelberg Security Games: Looking Beyond a Decade of Success. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, Jérôme Lang (Ed.). ijcai.org, 5494–5501. https://doi.org/10.24963/ijcai.2018/775

[50] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. 2018. A Review on Bilevel Optimization: From Classical to Evolutionary Approaches and Applications. *IEEE Trans. Evol. Comput.* 22, 2 (2018), 276–295. https://doi.org/10.1109/TEVC.2017.2712906

[51] Mohammad Amin Tajeddini, Hamed Kebriaei, and Luigi Glielmo. 2020. Decentralized Hierarchical Planning of PEVs Based on Mean-Field Reverse Stackelberg Game. *IEEE Trans Autom. Sci. Eng.* 17, 4 (2020), 2014–2024. https://doi.org/10.1109/TASE.2020.2986374

[52] Heinrich Von Stackelberg. 2010. *Market structure and equilibrium*. Springer Science & Business Media.

[53] Omar Abdel Wahab, Jamal Bentahar, Hadi Otrok, and Azzam Mourad. 2021. Resource-Aware Detection and Defense System against Multi-Type Attacks in the Cloud: Repeated Bayesian Stackelberg Game. *IEEE Trans. Dependable Secur. Comput.* 18, 2 (2021), 605–622. https://doi.org/10.1109/TDSC.2019.2907946

[54] Kaidi Wang, Zhiguo Ding, Daniel K. C. So, and George K. Karagiannidis. 2021. Stackelberg Game of Energy Consumption and Latency in MEC Systems With NOMA. *IEEE Trans. Commun.* 69, 4 (2021), 2191–2206. https://doi.org/10.1109/TCOMM.2021.3049356

[55] Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. 2020. A principled approach to data valuation for federated learning. In *Federated Learning*. Springer, 153–167.

[56] Shuyue Wei, Yongxin Tong, Zimu Zhou, and Tianshu Song. 2020. Efficient and Fair Data Valuation for Horizontal Federated Learning. In *Federated Learning - Privacy and Incentive*, Qiang Yang, Lixin Fan, and Han Yu (Eds.). Lecture Notes in Computer Science, Vol. 12500. Springer, 139–152. https://doi.org/10.1007/978-3-030-63076-8_10

[57] Detlof Von Winterfeldt and Gregory W Fischer. 1975. Multi-attribute utility theory: models and assessment procedures. *Utility, probability, and human decision making* (1975), 47–85.

[58] Hui Yin, Ye-Hwa Chen, and Dejie Yu. 2020. Stackelberg-Theoretic Approach for Performance Improvement in Fuzzy Systems. *IEEE Trans. Cybern.* 50, 5 (2020), 2223–2236. https://doi.org/10.1109/TCYB.2018.2883729

[59] Jinsung Yoon, Sercan Arik, and Tomas Pfister. 2020. Data valuation using reinforcement learning. In *International Conference on Machine Learning*. PMLR, 10842–10851.

[60] Jin Zhang and Qian Zhang. 2009. Stackelberg game for utility-based cooperative cognitiveradio networks. In *Proceedings of the 10th ACM Interational Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2009, New Orleans, LA, USA, May 18-21, 2009*, Edward W. Knightly, Carla-Fabiana Chiasserini, and Xiaojun Lin (Eds.). ACM, 23–32. https://doi.org/10.1145/1530748.1530753