Trading Fresh Data with Correlation

Junyi He, Meng Zhang, Qian Ma, and Jianwei Huang

***Abstract*—The increasing reliance on fresh data in real-time applications underscores the significance of commoditized fresh data. However, current research often neglects the crucial data correlation, essential in applications like intelligent transportation. This paper examines the trading of correlated fresh data, where a platform monitors the time-varying numerical status of multiple correlated data sources. Data users arrive stochastically, each seeking to obtain data from the platform to estimate the real-time status of a specific source of interest. To facilitate data trading, we propose a dynamic pricing policy that allows the platform to adjust prices in real-time. We demonstrate that dynamic pricing is an NP-hard mixed integer programming problem and propose an approximate algorithm. Our approach begins with threshold-based data allocation and uses linear programming to optimize pricing, achieving a logarithmic approximation ratio. For binary data sources, we derive an optimal closed-form solution, revealing that data correlation can benefit both the platform and users by offsetting data aging with spatially correlated fresher data. Insterestingly, despite users placing a higher valuation on fresher data, the presence of correlation results in fresher data being priced lower. This counterintuitive pricing strategy is designed to encourage users to engage in cross-source data purchasing. Numerical results show that the proposed approximate dynamic pricing policy can achieve at least 90% of the maximum dynamic pricing revenue. Additionally, data correlation can amplify the platform's revenue by up to 100% compared to scenarios without data correlation.**

***Index Terms*—Age of Information, fresh data trading, data correlation.**

I. Introduction

A. Motivations

Data is increasingly recognized as a critical digital asset, essential for data-driven decision tasks. Consequently, this growing importance has spurred the proliferation of data trading platforms, focusing on trading data from Internet of Things (e.g., [1], [2]) and mobile crowdsensing (e.g., [3]). The surge in real-time applications like intelligent transportation, online data analytics, and environmental monitoring, highlights the necessity of data freshness. The *Age of Information (AoI)* metric, which measures the time since the most recent data update, is vital for maintaining the timeliness and accuracy of decisions in these applications [4]–[10]. For instance, in intelligent transportation systems, up-to-date traffic data is essential for efficient routing [11]. This emphasis on acquiring fresh data has driven the development of a novel fresh data trading paradigm (e.g., [12]–[18]), where a platform determines the pricing of fresh data, while data users make decisions regarding data procurement.

However, most existing works on fresh data trading often overlook the crucial spatio-temporal correlations present in real-time applications (e.g., [12]–[18]). For instance, nearby intersections on a road exhibit similar traffic conditions due to their proximity and the temporal nature of traffic patterns [9], [10]. Similarly, stock prices in the same industry frequently display correlated movements. Understanding data correlation is crucial for developing a pricing mechanism that accurately captures the spatio-temporal value of data, thereby enhancing the efficiency of fresh data trading. However, this leads to several challenges.

The first challenge is the absence of definitive criteria for evaluating users' data valuation in the context of spatio-temporal correlations, which complicates fresh data pricing. Unlike traditional physical commodities, including, data valuation does not necessarily depend on data volume but on its ability to inform users' decision-making processes. Different users often assign different valuations to the same data because they have different motivations for using it. Moreover, the interplay of correlations across different times and places of interest (PoIs) means that data from one PoI at certain time can also inform predictions for others at different time points. Therefore, data valuation must account for these temporal and spatial dynamics, prompting us to address a key question.

Key Question 1: How to characterize users' valuation of data that exhibits spatio-temporal correlation?

The second challenge is the combinatorial complexity introduced by spatio-temporal data correlation into data pricing. For example, a user who focuses on a single PoI would traditionally require only data for that location. However, the presence of data correlation allows a user to infer information about one PoI from others. This makes the user's data demand combinatorial in temporal and spatial dimensions. When setting prices, the platform must account for these complex interdependencies across time and space. This complexity gives rise to our second key question:

Key Question 2: How should fresh data with spatio-temporal correlation be priced?

To address the first key question, we model the data correlation using a Gaussian Process and derive the users' data

Junyi He is with Shenzhen Institute of Artificial Intelligence and Robotics for Society (AIRS), School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong, China (e-mail: junyihe@link.cuhk.edu.cn).

Meng Zhang is with Zhejiang University/University of Illinois at Urbana-Champaign Institute, Zhejiang University, Haining 314400, China (e-mail: mengzhang@intl.zju.edu.cn).

Qian Ma is with the School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen, Guangdong 518107, China (e-mail: maqian25@mail.sysu.edu.cn).

Jianwei Huang is with the School of Science and Engineering, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen Key Laboratory of Crowd Intelligence Empowered Low-Carbon Energy Network, and CSIJRI Joint Research Centre on Smart Energy Storage, The Chinese University of Hong Kong, Shenzhen, Guangdong, 518172, P.R. China (corresponding author, e-mail: jianweihuang@cuhk.edu.cn).

valuation as the expected reduction in their estimation error based on the information extracted from data. Regarding the second key question, we investigate a *dynamic pricing policy* that enables the platform to set prices in real time, taking into account both data freshness and data correlation.

B. Results and Key Contributions

Our key results and contributions are summarized as follows:

- *Characterize Data Valuation with Spatio-temporal Correlation.* We utilize a Gaussian Process to model the intricate spatio-temporal correlations in data, allowing us to estimate its value by measuring the expected reduction in estimation error for users. To the best of our knowledge, this is the first work for data valuation that incorporates spatio-temporal correlation.
- *Optimal Dynamic Pricing Policy.* We address the NP-hard dynamic pricing problem by proposing a two-step approximate algorithm. It initiates with threshold-based data allocation and subsequently refines data pricing based on linear programming, achieving a logarithmic approximation ratio (in terms of the number of PoIs) for maximizing revenue. In a specific scenario, we derive a closed-form solution, revealing that data correlation can reduce prices, benefiting both the platform and users due to the platform's ability to counteract data obsolescence by integrating spatially correlated, fresher data.
- *Performance Evaluation.* Numerical results reveal that the proposed approximate dynamic pricing policy can reach at least 90% of the revenue generated by the optimal dynamic pricing policy. Importantly, data correlation can substantially enhance revenue, potentially doubling it compared to scenarios lacking data correlation.

We organize the rest of the paper as follows. In Section II, we review the literature. Section III introduces the system model and Section IV presents problem formulation. Section V studies the platform's revenue maximization with a dynamic pricing policy. Section VI gives the numerical results. Finally, Section VII concludes the paper.

II. Related Work

In this section, we review the existing literature on AoI minimization and economics in AoI, identifying key gaps that our work addresses.

AoI Minimization: Existing works along this line mainly focused on minimizing time-average AoI under a variety of system settings (e.g., [6]–[10]). For example, Zhao *et. al.* [6] proposed an age-threshold slotted ALOHA protocol to optimize the AoI in mobile network. Kadota *et. al.* [7] optimized the average AoI in random access networks with stochastic update generation. Sun *et. al.* [8] investigated optimal sampling policies to minimize the average AoI. However, these works assume that information across different sources is uncorrelated. Considering correlated data sources, Ramakanth *et al.* [9] investigated the design of scheduling policies aimed at minimizing monitoring error in systems with multiple correlated sources. They modeled the correlated sources as multiple Wiener processes, with increments characterized as multivariate Gaussian random variables sharing a covariance matrix. In a complementary approach, Tripathi *et al.* [10] introduced a probability-based correlation model, wherein updates from one source provide partial information about the current state of other sources with a specified probability. Based on this model, they presented AoI-minimizing scheduling policies. *However, these efforts did not consider economic AoI management from a platform perspective. Given the operational costs of generating data updates, providing economic incentives is essential.*

Economics in AoI: Recent studies on fresh data pricing have emphasized creating economic incentives for data sources to produce timely updates (e.g., [12]–[18]). For example, Li *et al.* [12] considered that a platform design a linear AoI-based pricing mechanism to incentivize sources to report fresh data in time. Wang *et al.* [13] investigated dynamic pricing strategies for data sources, aiming to minimize expected AoI and total payment over time. In a similar vein, Wang *et al.* [14] proposed an age-based pricing mechanism to incentivize sources to report fresh data. Zhang *et al.* [15] studied the optimal design of the reward when the sources' data generating cost is private and Wang *et al.* [16] dealt with the time-varying private information among platforms and the data sources. He *et al.* [17] proposed profit-maximizing fresh data pricing policies to strategic users who could wait for data purchase. *However, these works typically consider a single data source and overlook the complexities introduced by spatio-temporal data correlation in fresh data trading.*

III. System Model

In this section, we start with an introduction to the system overview. Next, we present the data freshness and data correlation model.

A. System Overview

As illustrated in Fig. 1, we consider a real-time monitoring system where a platform tracks the time-varying numerical status of different PoIs (e.g., traffic conditions on roadways) and provides the collected data to a population of stochastic arrival users. Within this system, there are I data sources situated at distinct PoIs, denoted as s_i where $i \in \mathcal{I} = \{1, 2, ..., I\}$. Each source s_i monitors a physical process at its respective PoI, represented by the random variable $Z(s_i, t)$ at a particular time t. The status information $Z(s_i, t)$ gathered from these data sources is spatially correlated. For instance, traffic congestion data from geographically proximate locations tends to change synchronously, reflecting the spatial relationships and mutual influence between nearby PoIs [9], [10].

On the demand side, each data user seeks to obtain data from platform for estimating the real-time status of a specific source of interest. We define users interested in estimating the real-time status at the source s_j as type-j users. Given the true real-time status $Z(s_j, t)$ of the source s_j at the time

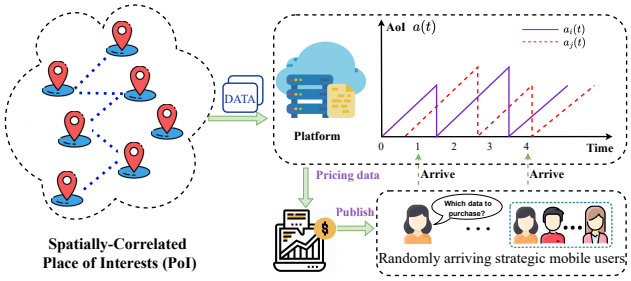

Fig. 1. System Model.

t and an estimator $\hat{Z}(s_j,t)$ chosen by a type-j user, its ex-post estimation error is $(Z(s_j,t)-\hat{Z}(s_j,t))^2$. For the sake of notational simplicity, we also denote the set of users' types as $\mathcal{I}$. Users of type $j \in \mathcal{I}$ arrive according to a Poisson Process with rate λ_j. Upon arrival, each type-j user has a unit demand for acquiring data from the platform to aid in its decision-making process, i.e., estimating the real-time status at the source s_j.

Next, we introduce the data freshness in Section III-B, and then characterize data spatio-temporal correlation in Section III-C.

B. Data Freshness Model

To quantify the freshness of the data stored on the platform sensed by the source s_i, we introduce the concept of AoI, denoted as $a_i(t)$.

***Definition** 1 (Age of information [4]):* Let $H_i(t)$ denote the time stamp of the platform's most recently received update from the source s_i before time t. The AoI $a_i(t)$ of the platform's data from the source s_i at the time t is

$$a_i(t) = t - H_i(t).$$

As in works [19], [20], the platform receives data update packets from data source s_i according to a Poisson Process with an arrival rate r_i. We consider that the platform will instantaneously receive updates once the updates are generated at the source [11], [14].

C. Data Correlation Model

In this subsection, we first define Gaussian Process and then model the data's spatio-temporal correlation accordingly.

1) Gaussian Process: Gaussian process models are widely adopted in machine learning for non-linear regression and classification, and often have impressive empirical performance [21], [22]. Formally, a Gaussian process, denoted as $Z(x)$, is a random function that encapsulates a stochastic process. Any finite collection of those random variables $Z(x_1), Z(x_2), ..., Z(x_n)$ adheres to a multivariate normal distribution [22]. A way of defining a Gaussian process is through the following kernel (covariance function) formalism:

***Definition** 2 (Gaussian Process [22]):* A Gaussian process $Z(x) \sim GP(m(x), \phi(x,x'))$ is defined in terms of a mean function $m(x)$ and a covariance function $\phi(x,x')$, where

$$m(x) = \mathbb{E}[Z(x)], \tag{1}$$

$$\phi(x,x') = \mathbb{E}[(Z(x)-m(x))(Z(x')-m(x'))]. \tag{2}$$

The mean function represents how the random functions behave on average, while the covariance function encodes the correlation structure, enabling us to model the data correlation.

We next present the spatio-temporal Gaussian Process, a model that captures the joint correlation of information across time and space. This approach is pivotal for understanding data correlations in fields, such as environmental monitoring and traffic prediction [23]–[25]. Let $x=(s,t)$, where s symbolizes the spatial input and t represents the temporal input; a spatio-temporal Gaussian process is

$$Z(s,t) \sim GP(m(s,t), \phi(s,t,s',t')).$$

2) Data Spatio-temporal Correlation: As in works [26], in this paper, we model the observed phenomenon $Z(s,t)$ as a normalized spatio-temporal Gaussian process wherein the mean function $E[Z(s,t)]=0$ for any s,t. Following works [26], [27], we focus on the widely used exponential kernel, defined as follows:

$$\phi(s,t,s',t') = \exp\left(-\frac{h|t-t'|}{2} - \frac{\epsilon\|s-s'\|_2}{2}\right), \tag{3}$$

where h is the constant temporal decay parameter and ϵ is constant scaling factors in spatial domains. The temporal component h captures the temporal dependencies between consecutive time points, considering factors such as temporal trends. The spatial component ϵ captures the spatial dependencies between different locations, considering spatial proximity and auto-correlation.

The covariance between two correlated statuses, i.e., $Z(s,t)$ and $Z(s',t')$, is a function of their Euclidean distance $\|s-s'\|_2$ and their generation time difference $|t-t'|$. The Gaussian Process with exponential kernel has been widely used to model numerical sensing data, e.g., traffic data [24], [25].

IV. Problem Formulation

In this section, we start with introducing a two-stage game to model the interactions between the platforms and stochastic arrival users. Next, we formulate the user's data purchasing problem and the platform's data pricing problem.

A. Game-based Data Trading

We consider the trading of fresh data under an incomplete information scenario, where each user's type is private information. The platform does not know the exact type of each arriving user but only knows the arrival rate of each user type.

At each time t, the data available on the platform may have aged, and the AoI of the platform's data at t is denoted by $\boldsymbol{a}(t)=\{a_i(t), \forall i \in \mathcal{I}\}$. We represent the data on the platform at time t as $\{Z(s_i, t-a_i(t)), \forall i \in \mathcal{I}\}$, where the data item $Z(s_i, t-a_i(t))$ refers to the observation made by the data

source s_i at the time $t - a_i(t)$. We model the data trading interaction between users and the platform as a Stackelberg game, outlined as follows:

- **Stage I (Platform's Data Pricing):** At each time t, the platform announces a data pricing policy $\boldsymbol{p}(t) = \{p_i(t), \forall i \in \mathcal{I}\}$. If a user arrives at the time t opts to purchase the data item with AoI $a_i(t)$ from the source s_i, i.e., $Z(s_i, t-a_i(t))$, it needs to pay $p_i(t)$ to the platform. The platform's goal is to set the data price at each time t to maximize its time-average revenue.
- **Stage II (User's Data Purchasing):** Upon arrival at time t, each type-j user decides whether to purchase data and, if so, which data item to buy to maximize its payoff. This decision involves weighing the value of the data—considering its freshness and the correlation with other data—against the price quoted by the platform. After making a purchase, if any, each arriving type-j user will select an estimator to minimize the Mean Square Error (MSE).

B. Stage II: User's Data Purchasing

In this subsection, we first derive users' data valuation and then formulate the users' data purchasing problem.

1) Users' Data Valuation: A user's data valuation is contingent upon the quality of the data they purchase, which is tied to the precision of their estimate of the real-time status. This accuracy is influenced by the AoI of the acquired data and the spatio-temporal correlation. We will initially describe the value of the user's initial information and next, determine the incremental value added by the platform's provided data.

Value of User's Initial Information: Upon arrival at the time t, each type-j user tries to estimate real-time status $Z(s_j, t)$ at the source s_j as accurately as possible, that is, by choosing an estimator $\hat{Z}(s_j, t)$ with its prior information to minimize the MSE,

$$\min_{\hat{Z}(s_j,t)} \mathbb{E}[(Z(s_j,t) - \hat{Z}(s_j,t))^2]. \tag{4}$$

Bayesian Updating and Conditional Distribution: Upon acquiring data $Z(s_i, t - a_i(t))$ from the platform, a type-j user can refine its estimate of the real-time status $Z(s_j, t)$ at the source s_j more precisely. Unlike scenarios without data correlation, the existence of the correlation between different data sources presents a type-j user with a broader range of purchasing options. The data acquired by a type-j user may be either the data from the specific source of interest, namely $Z(s_j, t - a_j(t))$, or other data that demonstrates correlation with the data source the user aims to estimate, that is, $Z(s_i, t - a_i(t))$, where $i \neq j$.

By observing the data $Z(s_i, t - a_i(t))$, the type-j user can obtain the conditional distribution of the random variable $Z(s_j, t)|Z(s_i, t-a_i(t))$ via Bayes' rule. Since $Z(s, t)$ follows a Gaussian process, we know that $Z(s_j, t)$ and $Z(s_i, t-a_i(t))$ will follow a multivariate Gaussian distribution [22], i.e.,

$$\mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1, & \phi(s_j,t,s_i,t-a_i(t))\\ \phi(s_j,t,s_i,t-a_i(t)), & 1\end{bmatrix}\right). \tag{5}$$

According to the property of multivariate Gaussian distribution [22], we can get the posterior distribution of the random variable $Z(s_j, t)$ conditioned on $Z(s_i, t-a_i(t))$ as in following Lemma.

***Lemma** 1 (Conditional Gaussian distribution [22]):* The posterior distribution of the random variable $Z(s_j, t)$ conditioned on $Z(s_i, t - a_i(t))$ is also Gaussian, where the mean value $\mathbb{E}[Z(s_j, t)|Z(s_i, t - a_i(t))]$ is

$$\phi(s_j, t, s_i, t - a_i(t))Z(s_i, t - a_i(t)), \tag{6}$$

and the posterior covariance is

$$\text{Var}[Z(s_j,t)|Z(s_i,t-a_i(t))] = 1-\phi(s_j,t,s_i,t-a_i(t))^2. \tag{7}$$

Next, the user will choose an estimator $\hat{Z}(s_j, t)$ which minimizes the conditional MSE:

$$\min_{\hat{Z}(s_j,t)} \mathbb{E}[(Z(s_j,t) - \hat{Z}(s_j,t))^2|Z(s_i,t-a_i(t))]. \tag{8}$$

Improvement in Estimating Accuracy: Upon receipt of data $Z(s_i, t - a_i(t))$ from the platform, a type-j user experiences an improvement in estimating accuracy. We can quantify the user's valuation of the data by the decrease in MSE.

***Definition** 3 (Users' Data Valuation):* The type-j user's valuation about the data $Z(s_i, t - a_i(t))$ is the reduction in MSE, given by:

$$\begin{aligned}\Psi_j(Z(s_i,t-a_i(t))) = &\min_{\hat{Z}(s_j,t)} \mathbb{E}[(Z(s_j,t) - \hat{Z}(s_j,t))^2]\\ &- \min_{\hat{Z}(s_j,t)} \mathbb{E}[(Z(s_j,t) - \hat{Z}(s_j,t))^2|Z(s_i,t-a_i(t))].\end{aligned} \tag{9}$$

In the fresh data market, the data correlation model presented in Equation (3) and its associated parameters are common knowledge among users. Thus, give the platform's data AoI $\boldsymbol{a}(t)$, a type-j user arriving at the time t can calculate its data valuation $\Psi_j(Z(s_i, t - a_i(t)))$ about the data item $Z(s_i, t - a_i(t))$ based on Definition 3 before purchasing data, as shown in the following result.

***Proposition** 1 (Users' Data Valuation):* For a type-j user who arrives at time t, its valuation about the data $Z(s_i, t - a_i(t))$ is

$$\Psi_j(Z(s_i, t - a_i(t))) = \exp\left(-ha_i(t) - \epsilon\|s_i - s_j\|_2\right). \tag{10}$$

Proof Sketch: To derive user's data valuation, we will first derive the users' minimum prior MSE and next calculate its minimum conditional MSE. Under the covariance function in (3), a type-j user arriving at time t has prior knowledge that $Z(s_j, t) \sim \mathcal{N}(0, 1)$ and thus incurs a minimum MSE loss of 1. Upon acquiring the data $Z(s_i, t - a_i(t))$ from the platform, this user derives the posterior distribution of the random variable $Z(s_j, t)$ conditioned on $Z(s_i, t - a_i(t))$. As in Lemma 1, this posterior distribution is also Gaussian. The estimator that minimizes its conditional MSE is $\phi(s_j, t, s_i, t - a_i(t))Z(s_i, t - a_i(t))$, and the corresponding minimum conditional MSE loss is $1-(\phi(s_j, t, s_i, t-a_i(t)))^2$. Thus, for a type-$j$ user who arrives at time t, its valuation

about the data $Z(s_i, t-a_i(t))$ is $(\phi(s_j, t, s_i, t-a_i(t)))^2$, which completes the proof.

Proposition 1 shows that a type-j user's data valuation of aged data $Z(s_j, t-a_j(t))$ from the source of interest is $\exp(-ha_j(t))$, solely determined by the AoI $a_j(t)$. However, when given the correlated data $Z(s_i, t-a_i(t)), i \neq j$, the type-$j$ user must account for the impact of the data AoI $a_i(t)$ and spatial correlation $\epsilon\|s_i - s_j\|_2$.

2) User's Data Purchasing Problem: We first introduce the users' data purchasing decision and define the payoff function, followed by the formulation of the users' data purchasing problem.

Given data items $\{Z(s_i, t-a_i(t)), \forall i \in \mathcal{I}\}$ offered by the platform at time t, we use $b_j(\boldsymbol{a}(t)) \in \mathcal{I} \cup \{0\}$ to denote the data purchasing decision of an arrival type-j user. If the user purchases the data item $Z(s_i, t-a_i(t))$, we let $b_j(\boldsymbol{a}(t)) = i$. Conversely, if the user chooses to leave the market without data purchase, we represent this decision by $b_j(\boldsymbol{a}(t)) = 0$.

Considering the AoI $\boldsymbol{a}(t)$ of platform's data at t and the data pricing policy $\boldsymbol{p}(t)$, the payoff of an arriving type-j user at time t is

$$u_j(b_j(\boldsymbol{a}(t)), \boldsymbol{p}(t)) = \begin{cases} \Psi_j(Z(s_i, t-a_i(t))) - p_i(t) & \text{if } b_j(\boldsymbol{a}(t)) = i, \\ 0 & \text{if } b_j(\boldsymbol{a}(t)) = 0. \end{cases} \quad (11)$$

Such a user decides its data purchasing policy to maximize its payoff by solving the following optimization problem

$$\max_{b_j(\boldsymbol{a}(t))} \quad u_j(b_j(\boldsymbol{a}(t)), \boldsymbol{p}(t)) \quad (12a)$$

$$\text{var.} \quad b_j(\boldsymbol{a}(t)) \in \mathcal{I} \cup \{0\}, \quad (12b)$$

and we denote his optimal data purchasing decision as

$$b_j^*(\boldsymbol{a}(t)) = \underset{b_j(t) \in \mathcal{I} \cup \{0\}}{\operatorname{argmax}} \; u_j(b_j(\boldsymbol{a}(t)), \boldsymbol{p}(t)), \forall j, t. \quad (13)$$

When users are indifferent between purchasing two data items, we assume that users will opt for the fresher data.

C. Stage I: Platform's Data Pricing Problem

Next, we formulate the platform's data pricing problem. Let $\mathbb{1}_E$ an indicator function that equals 1 if the event E is true and 0 otherwise. By substituting the optimal solution $b_j^*(\boldsymbol{a}(t))$ from Stage II into Stage I's problem, we can write the platform's time-average revenue maximization problem as follows:

***Problem** 1:* Revenue Maximization Problem.

$$\max \quad \lim_{T\to\infty} \frac{1}{T} \int_{t=0}^{T} \sum_{i\in\mathcal{I}} p_i(t) \left(\sum_{j\in\mathcal{I}} \lambda_j \mathbb{1}_{\{b_j^*(\boldsymbol{a}(t))=i\}} \right) \mathrm{d}t \quad (14a)$$

$$\text{s.t.} \quad (13). \quad (14b)$$

$$\text{var.} \quad p_i(t) \geq 0, \forall i \in \mathcal{I}, \forall t. \quad (14c)$$

V. Optimal Dynamic Pricing policy

In this section, we first formulate the dynamic pricing problem as a mixed integer programming problem and show it is NP-hard. Next, we propose an approximate dynamic pricing algorithm and prove its approximation ratio. Finally, to understand the impacts of the data correlation, we further consider a special case with binary data sources.

A. Platform's Revenue Maximization Problem

In this subsection, we demonstrate that the dynamic pricing problem is decoupled across different time instances. We formulate the revenue maximization problem at each time as a mixed integer programming problem, which we establish to be NP-hard.

Under the dynamic pricing policy, the platform has the flexibility to set prices at each time t. As each new user arrives and makes an immediate purchase decision without delay, the dynamic pricing problem is decoupled over time. For each time instance t, the platform's optimal dynamic pricing policy is to choose data prices $\boldsymbol{p}(t)$ to maximize the revenue at each time t, given the current data AoI $\boldsymbol{a}(t)$. For simplicity, we will omit the time index t in the rest of this subsection.

Given the AoI $\boldsymbol{a} = \{a_i, \forall i \in \mathcal{I}\}$, according to Proposition 1, we let $v_{ij} = \exp(-ha_i - \epsilon\|s_i - s_j\|_2)$ denote the type-j user's valuation of the platform's data from the data source s_i. We represent $x_{ij} \in \{0, 1\}$ as an indicator of whether a type-j user purchases data from the source s_i. The platform's revenue maximization is formulated as a mixed integer programming problem:

***Problem** 2:* Dynamic Pricing Problem.

$$\max \quad \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{I}} \lambda_j x_{ij} p_i \quad (15a)$$

$$\text{s.t.} \quad x_{ij}(v_{ij} - p_i) \geq 0, \forall i, j \in \mathcal{I}, \quad (15b)$$

$$\sum_{i\in\mathcal{I}} x_{ij} \leq 1, \forall j \in \mathcal{I}, \quad (15c)$$

$$x_{ij}(v_{ij} - p_i) \geq x_{ij}(v_{kj} - p_k), \forall i, j, k \in \mathcal{I}, \quad (15d)$$

$$\text{var.} \quad p_i \geq 0, \forall i \in \mathcal{I}, \quad (15e)$$

$$x_{ij} \in \{0, 1\}, \forall i \in \mathcal{I}, j \in \mathcal{I}. \quad (15f)$$

Unfortunately, Problem 2 is a combinatorial optimization problem, and the following Proposition shows that Problem 2 is NP-hard. Due to space constraints, proofs are provided in [28].

***Proposition** 2:* Problem 2 is NP-hard.

Problem 2 represents a combinatorial optimization challenge. Solving it for smaller instances is feasible with Integer Programming solvers like Gurobi or IBM CPLEX Optimizer, which can identify optimal solutions. However, for larger-scale scenarios, the complexity necessitates the development of an approximate algorithm. Before that, we will first identify a specially structured instance of Problem 2 for which we can find the optimal solution.

***Proposition** 3:* Given AoI $\boldsymbol{a}$, if for all $j \in \mathcal{I}$,

$$\exp(-ha_j) \geq \max_{i \in \mathcal{I}, i \neq j} \exp(-ha_i - \epsilon \|s_i - s_j\|_2),$$

the optimal data price is $p_j^* = \exp(-ha_j)$ for each $j \in \mathcal{I}$.

Proposition 3 means that if the platform's data from the source s_j is sufficiently fresh, making it the most valued by type-j users for all $j \in \mathcal{I}$, then encouraging cross-purchasing does not increase the platform's benefits. Instead, the platform should sell data from the source s_j to type-j users.

We will develop an approximate dynamic pricing algorithm to solve Problem 2.

B. Approximate Algorithm Design

In this subsection, we focus on developing an approximate dynamic pricing algorithm (ADP). Our approach decomposes Problem 2 into two key subproblems: data allocation, which determines x_{ij}, and data pricing, which determines prices p_i.

We present the ADP in Algorithm 1. Let $v_j^{\max} = \max_{i \in \mathcal{I}} \exp(-ha_i - \epsilon \|s_i - s_j\|_2)$ denote the maximum data valuation of a type-j user across all data. We rank user types by $v_j^{\max}$ in an ascending order, establishing a sequence that guides our data allocation effectively: $v_{k_1}^{\max} \leq v_{k_2}^{\max} \leq ... \leq v_{k_I}^{\max}$ (Line 2). Our algorithm addresses Problem 2 through an iterative process with I iterations. In each iteration, we first make the data allocation decision based on users' maximum data valuation. Then, fixing the allocation decision x_{ij}, we solve the remaining pricing problem, which is linear programming and can be efficiently solved.

1) Data Allocation in Iteration l: In each iteration l, we select the corresponding valuation threshold $v_{k_l}^{\max}$ from the sorted valuation sequence $(v_{k_1}^{\max}, v_{k_2}^{\max}, ..., v_{k_I}^{\max})$ (Line 5). A type-j user whose maximum data valuation exceeds this threshold is allocated data that maximizes its payoff, identified as $i^* = \operatorname{argmax} v_{ij}$ (Line 8-9). We set $x_{i^*j} = 1$ and $x_{ij} = 0$, for each $i \neq i^*, i \in \mathcal{I}$. If a type-$j$ user's maximum valuation is lower than the threshold, we will not allocate any data to this user and set $x_{ij} = 0$, for each $i \in \mathcal{I}$ (Line 11).

2) Data Pricing in Iteration l: Once data allocation is determined, we proceed to optimize the data prices. Fixing the allocation matrix $\{x_{ij}, \forall i \in \mathcal{I}, \forall j \in \mathcal{I}\}$, we define $G_i = \{j \in \mathcal{I} : x_{ij} = 1\}$ as the set of users allocated data $Z(s_i, t - a_i)$. The platform can then optimize the data price by solving the following linear programming:

***Problem** 3:* Data Pricing Problem in Iteration l.

$$\max \quad \sum_{i \in \mathcal{I}} \sum_{j \in G_i} \lambda_j p_i \tag{16a}$$

$$\text{s.t.} \quad v_{ij} - p_i \geq 0, \forall i \in \mathcal{I}, \forall j \in G_i, \tag{16b}$$

$$v_{ij} - p_i \geq v_{kj} - p_k, \forall i, k \in \mathcal{I}, j \in G_i, \tag{16c}$$

$$\text{var.} \quad p_i \geq 0, \forall i \in \mathcal{I}. \tag{16d}$$

Problem 3 is a linear programming problem, which can be effectively solved in polynomial time [29].

Algorithm 1 Approximate Dynamic Pricing (ADP)

Input: AoI $\boldsymbol{a} = \{a_i, \forall i \in \mathcal{I}\}$, arrival rate $\{\lambda_i, \forall i \in \mathcal{I}\}$, h, ϵ.

1: Initialize iteration index $l = 1$.
2: Calculate $v_j^{\max} = \max_{i \in \mathcal{I}} \exp(-ha_i - \epsilon \|s_i - s_j\|_2)$.
3: Sort $\{v_j^{\max}, \forall j \in \mathcal{I}\}$ in an ascending manner such that

$$v_{k_1}^{\max} \leq v_{k_2}^{\max} \leq ... \leq v_{k_I}^{\max}.$$

4: **repeat**
5: Choose a valuation threshold $v^{\text{Thr}} = v_{k_l}^{\max}$.
6: **for** user type $j \in \mathcal{I}$ **do**
7: **if** $v_j^{\max} \geq v^{\text{Thr}}$ **then**
8: Determine $i^* = \operatorname{argmax}_{i \in \mathcal{I}} \exp(-ha_i - \epsilon \|s_i - s_j\|_2)$.
9: Set $x_{i^*j} = 1$ and $x_{ij} = 0, \ \forall i \neq i^*, i \in \mathcal{I}$.
10: **else**
11: Set $x_{ij} = 0, \ \forall i \in \mathcal{I}$.
12: **end if**
13: **end for**
14: Obtain the optimal data prices $\boldsymbol{p}_l^*$ and the revenue Υ_l^* by solving Problem 3.
15: $l = l + 1$
16: **until** $l > I$.

return: The maximum revenue Υ^* and data prices $\boldsymbol{p}^*$.

At the end of each iteration, we record the maximum revenue achieved and the corresponding data prices. After I iterations, we identify the maximum revenue and the corresponding data prices from all iterations.

C. Platform's Time-average Revenue

Next, we compute the platform's time-average revenue under the dynamic pricing policy. We first derive the stationary distribution of the AoI process $a_i(t)$ for the platform's data about the data source s_i, which is defined as

$$F(x_1, x_2, ..., x_I) = \lim_{t \to \infty} \Pr(a_1(t) \leq x_1, ..., a_I(t) \leq x_I). \tag{17}$$

***Proposition** 4:* The stationary cumulative density function of the platform's data AoI $\boldsymbol{a} = \{a_i, \forall i \in \mathcal{I}\}$ is $F(x_1, x_2, ..., x_I) = \Pi_{i=1}^I (1 - \exp(-r_i x_i))$ and the probability density function $f(x_1, x_2, ..., x_I) = \Pi_{i=1}^I (r_i \exp(-r_i x_i))$.

Proof Sketch: Since the platform independently receives updates from each data source s_i following a Poisson process with rate r_i, without transmission delay, the stationary distribution of AoI a_i is an exponential distribution with the parameter r_i and the joint probability density function is $f(x_1, x_2, ..., x_I) = \Pi_{i=1}^I (r_i \exp(-r_i x_i))$ [30].

Let $\Phi'(\boldsymbol{a})$ represent the revenue generated by the proposed approximate algorithm for a given AoI $\boldsymbol{a} = \{a_i, \forall i \in \mathcal{I}\}$. Based on Proposition 4, we can write the time-average revenue under the proposed approximate dynamic pricing algorithm as

$$\mathbb{E}[\Phi'(\boldsymbol{a})] = \int \cdots \int_{\boldsymbol{a} \in \mathcal{A}} \Phi'(\boldsymbol{a}) f(\boldsymbol{a}) \, \mathrm{d}\boldsymbol{a}, \tag{18}$$

where $\mathcal{A} \triangleq [0, +\infty] \times [0, +\infty] \times \cdots \times [0, +\infty] \subseteq \mathbb{R}^I$.

D. Computational Complexity and Approximation Ratio

We now derive the computational complexity and approximation ratio of the proposed algorithm ADP. Let $\Phi^*(\boldsymbol{a})$ represent the revenue generated by the optimal dynamic pricing

policy for a given AoI $\boldsymbol{a} = \{a_i, \forall i \in \mathcal{I}\}$ and $\mathbb{E}[\Phi^*(\boldsymbol{a})]$ is the maximum time-average revenue.

***Definition** 4 (γ-Approximate Pricing Policy):* A pricing policy is a γ-approximate if the expected revenue of the policy $\mathbb{E}[\Phi'(\boldsymbol{a})]$ is at least $\mathbb{E}[\Phi^*(\boldsymbol{a})]/\gamma$.

***Theorem** 1:* Algorithm ADP runs in polynomial time and is $\lambda_{\max}\ln(I)/\lambda_{\min}$-approximate, where $\lambda_{\min} = \min_i \lambda_i$ and $\lambda_{\max} = \max_i \lambda_i$.

In order to understand the impact of the data correlation, we further consider a special case with binary data sources, i.e., $I = 2$, in which case we are able to characterize the optimal dynamic pricing policy in a closed form.

E. Special Case: Binary Data Sources

With $I = 2$ data sources, we will derive the closed-formed optimal dynamic pricing policy and investigate the impact of correlation on both the platform and users.

For notational convenience, we will denote the fraction of type-i users as $\beta_i = \frac{\lambda_i}{\lambda_1+\lambda_2}$. We will use index $-i$ to denote the index $j \neq i$, and let $d = \epsilon\|s_1 - s_2\|_2$ represent the spatial correlation between the two data sources.

1) Optimal Dynamic Pricing Policy: We can determine the optimal dynamic pricing policy as follows:

***Lemma** 2 (Optimal Dynamic Pricing Policies with Binary Sources):* At time t with AoI $\boldsymbol{a}(t) = \{a_1(t), a_2(t)\}$, for the data item $Z(s_i, t - a_i(t)), i \in \{1, 2\}$:

(i) If $\beta_i < e^{-d}$, the optimal dynamic price is

$$p_i^*(t) = \begin{cases} e^{-ha_i(t)-d}, & \text{if } a_i(t) \leq a_{-i}(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_i}{1-\beta_i}\right), \\ e^{-ha_i(t)}, & \text{otherwise.} \end{cases} \tag{19}$$

(ii) If $\beta_i \geq e^{-d}$, the optimal dynamic price is $p_i^*(t) = e^{-ha_i(t)}$.

Lemma 2 sheds light on an interesting pattern of the pricing policy: the pricing of data $Z(s_i, t - a_i(t))$ does not always decrease with its own AoI $a_i(t)$, as shown in (24). *This can lead to a fresher data $Z(s_i, t - a_i(t))$ being priced lower, which is counter-intuitive.* The rationale behind reducing the price is to incentivize users of type $-i$ to purchase fresher data $Z(s_i, t - a_i(t))$, resulting in larger revenue for the platform.

2) Impacts of Data Correlation: Next, we explore the effects of data correlation under the optimal dynamic pricing by comparing scenarios with and without data correlation.

***Corollary** 1 (Effects of Data Correlation):* In the presence of data correlation, if $\beta_i < e^{-d}$: then for type-i users arriving when $a_i(t) \leq a_{-i}(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_i}{1-\beta_i}\right)$, the price of data $Z(s_i, t - a_i(t))$ is lower compared to a scenario without such correlation. Meanwhile, the payoff for these users and the platform's revenue increase.

Corollary 1 demonstrates that, contrary to scenarios lacking data correlation, the platform can achieve higher revenue even with lower data prices with data correlation. Reducing the price of data $Z(s_i, t - a_i(t))$ incentivizes users of type $-i$ to choose this data over the others, thereby enhancing revenue through users' cross-purchasing. Meanwhile, the type-i users also benefit.

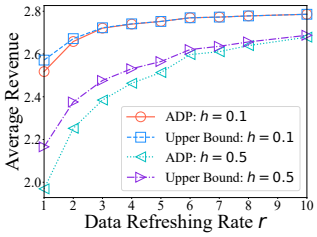

Fig. 2. The average revenue under dynamic pricing policy ADP with data refreshing rate r.

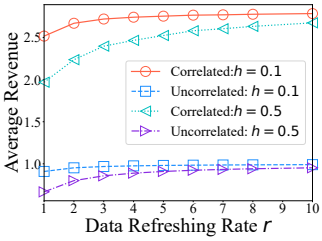

Fig. 3. The value of data correlation under dynamic pricing policy with data refreshing rate r.

VI. Numerical Results

In this section, we evaluate the proposed pricing policy in terms of the achieved revenue. Next, we investigate the value of data correlation.

We set the spatial correlation coefficient $\epsilon = 0.01$. The arrival rate of each type of user follows a uniform distribution in $[0, 1]$, and the distance between any two data sources is uniformly distributed in $[10, 100]$.

A. Performance of the Pricing Policy

To evaluate our proposed pricing policies, we benchmark them against the maximum achievable average social welfare, which serves as an upper bound for the platform's maximum average revenue. By doing so, we avoid the challenging task of computing the optimal dynamic pricing policy for an NP-hard problem at each time t. Given the AoI $\boldsymbol{a}$, let $v_j^{\max}(\boldsymbol{a})$ denotes the maximum data valuation of a type-j user. Consequently, the maximum social welfare is the aggregate of all users' maximum valuations, calculated as $\sum_{j\in\mathcal{I}} v_j^{\max}(\boldsymbol{a})\lambda_j$. The maximum average social welfare is $\int_{\boldsymbol{a}\in\mathcal{A}} \sum_{j\in\mathcal{I}} v_j^{\max}(\boldsymbol{a})\lambda_j f(\boldsymbol{a})\, \mathrm{d}\boldsymbol{a}$, which is the highest average revenue the platform can attain under any pricing strategy.

Fig. 2 compares the average revenue achieved by the proposed approximate dynamic pricing policy and the revenue upper bound, the maximum average social welfare. We consider $I = 5$ and set a uniform data refreshing rate r for all data sources. We find that our approximate dynamic pricing policy, ADP, consistently achieves at least 90% of the revenue upper bound under different values of r, outperforming the theoretic performance guarantee $1/\ln 5 \approx 0.621$. This result underscores ADP's effectiveness in capturing a significant portion of the potential revenue. As the data refresh rate increases, ADP's performance nearly matches the upper bound, positioning it as a competitive and efficient solution for pricing correlated data.

B. Value of Data Correlation

In this subsection, we explore the value of data correlation in the context of a platform's revenue generation. Specifically, we examine how the platform's maximum revenue changes in the presence or absence of data correlation.

Fig. 3 shows that how the value of data correlation changes with the data refreshing rate. Here, we consider $I = 5$ and set a uniform data refreshing rate r for all data sources.

For the case without data correlation, we derive the optimal dynamic pricing policy and compute the corresponding average revenue. For the case with data correlation, we calculate the average revenue achieved by the proposed approximate dynamic pricing policy. We find that data correlation can *double* the revenue compared to a scenario without correlation. This suggests that data correlation yields significant economic benefits.

VII. Conclusion

This paper highlighted the critical yet often overlooked role of spatio-temporal correlation in monetizing fresh data for real-time applications. By demonstrating the NP-hardness of formulating an optimal dynamic pricing policy, we proposed an efficient algorithm with a logarithmic approximation ratio. Our findings revealed the counterintuitive effect of data correlation: it reduces prices while fostering mutual benefits for platforms and users, as platforms can refresh their data with spatially correlated, more recent information. Numerical results validated the effectiveness of our approach, achieving performance close to the revenue upper bound. Looking ahead, future research could explore a more complex data trading mechanism where the platform first aggregates data from multiple data sources and selles aggregated results to users.

References

[1] Z. Zheng, W. Mao, Y. Xing, and F. Wu, "On designing market model and pricing mechanisms for IoT data exchange," *IEEE Transactions on Mobile Computing*, pp. 1–16, 2024.

[2] Q. Li, Z. Li, Z. Zheng, F. Wu, S. Tang, Z. Zhang, and G. Chen, "Capitalize your data: Optimal selling mechanisms for iot data exchange," *IEEE Transactions on Mobile Computing*, vol. 22, no. 4, pp. 1988–2000, 2021.

[3] H. Sun, M. Xiao, Y. Xu, G. Gao, and S. Zhang, "Privacy-preserving stable crowdsensing data trading for unknown market," in *Proceedings of IEEE INFOCOM*, 2023.

[4] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1183–1210, 2021.

[5] Y. Sun, I. Kadota, R. Talak, and E. Modiano, *Age of information: A new metric for information freshness*. Springer Nature, 2022.

[6] F. Zhao, N. Pappas, C. Ma, X. Sun, T. Q. Quek, and H. H. Yang, "Age-threshold slotted aloha for optimizing information freshness in mobile networks," *IEEE Transactions on Wireless Communications*, 2024.

[7] I. Kadota and E. Modiano, "Age of information in random access networks with stochastic arrivals," in *Proceedings of IEEE INFOCOM*. IEEE, 2021, pp. 1–10.

[8] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[9] R. V. Ramakanth, V. Tripathi, and E. Modiano, "Monitoring correlated sources: Aoi-based scheduling is nearly optimal," *IEEE Transactions on Mobile Computing*, 2024.

[10] V. Tripathi and E. Modiano, "Optimizing age of information with correlated sources," in *Proceedings of ACM Mobihoc*, 2022, pp. 41–50.

[11] F. Li, Y. Sang, Z. Liu, B. Li, H. Wu, and B. Ji, "Waiting but not aging: Optimizing information freshness under the pull model," *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 465–478, 2020.

[12] B. Li and J. Liu, "Achieving information freshness with selfish and rational users in mobile crowd-learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1266–1276, 2021.

[13] X. Wang and L. Duan, "Dynamic pricing and mean field analysis for controlling age of information," *IEEE/ACM Transactions on Networking*, vol. 30, no. 6, pp. 2588–2600, 2022.

[14] Z. Wang, Q. Meng, S. Zhang, and H. Luo, "Incentivizing fresh information acquisition via age-based reward," *IEEE Transactions on Networking*, 2025.

[15] M. Zhang, A. M. Arafa, E. Wei, and R. Berry, "Optimal and quantized mechanism design for fresh data acquisition," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1226–1239, 2021.

[16] Z. Wang, L. Gao, and J. Huang, "Taming time-varying information asymmetry in fresh status acquisition," *Proceedings of IEEE INFOCOM*, 2021.

[17] J. He, M. Zhang, Q. Ma, and J. Huang, "How to price fresh data with strategic users," in *Proceedings of IEEE International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)*, 2023, pp. 1–8.

[18] M. Zhang, A. Arafa, J. Huang, and H. V. Poor, "Pricing fresh data," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1211–1225, 2021.

[19] A. Javani, M. Zorgui, and Z. Wang, "Age of information for multiple-source multiple-server networks," *IEEE/ACM Transactions on Networking*, 2024.

[20] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing the age of information through queues," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 5215–5232, 2019.

[21] A. Wilson and R. Adams, "Gaussian process kernels for pattern discovery and extrapolation," in *Proceedings of ICML*. PMLR, 2013, pp. 1067–1075.

[22] M. Seeger, "Gaussian processes for machine learning," *International journal of neural systems*, vol. 14, no. 02, pp. 69–106, 2004.

[23] X. Feng, X. Ling, H. Zheng, Z. Chen, and Y. Xu, "Adaptive multi-kernel svm with spatial–temporal correlation for short-term traffic flow prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2001–2013, 2018.

[24] Z. Liu, C. Lyu, Z. Wang, S. Wang, P. Liu, and Q. Meng, "A gaussian-process-based data-driven traffic flow model and its application in road capacity analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 1544–1563, 2023.

[25] F. Rodrigues, K. Henrickson, and F. C. Pereira, "Multi-output gaussian processes for crowdsourced traffic data imputation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 594–603, 2018.

[26] H. Zhang, Z. Jiang, S. Xu, and S. Zhou, "Error analysis for status update from sensors with temporally and spatially correlated observations," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2136–2149, 2020.

[27] J. Hribar, M. Costa, N. Kaminski, and L. A. DaSilva, "Using correlated information to extend device lifetime," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2439–2448, 2018.

[28] J. He, M. Zhang, Q. Ma, and J. Huang, "Trading fresh correlated data: Online appendix," 2024. [Online]. Available: https://www.dropbox.com/s/nf8o8qcde698whj/StrategicAP.pdf?dl=0

[29] G. K. Leonid, "A polynomial algorithm for linear programming," *Doklady Akademii Nauk SSSR*, vol. 244, pp. 1093–1096, 1979.

[30] Y. Inoue, H. Masuyama, T. Takine, and T. Tanaka, "A general formula for the stationary distribution of the age of information and its application to single-server queues," *IEEE Transactions on Information Theory*, vol. 65, no. 12, pp. 8305–8324, 2019.

[31] S. Roch, P. Marcotte, and G. Savard, *Design and Analysis of an approximation algorithm for Stackelberg network pricing*. Citeseer, 2003.

VIII. Appendix

A. Proof of Proposition 1.

To derive user's data valuation, we will first derive the users' minimum prior MSE and next calculate its minimum conditional MSE.

Under the covariance function in, a type-j user has prior knowledge that $Z(s_j,t) \sim \mathcal{N}(0,1)$ and thus incurs a minimum MSE loss of 1. Upon acquiring the data $Z(s_i, t-a_i(t))$ from the platform, this user derives the posterior distribution of the random variable $Z(s_j,t)$ conditioned on $Z(s_i,t-a_i(t))$. As in Lemma 1, this posterior distribution is also Gaussian. The estimator that minimizes its conditional MSE is

$\phi(s_j, t, s_i, t - a_i(t))Z(s_i, t - a_i(t))$, and the corresponding minimum conditional MSE loss is $1-(\phi(s_j, t, s_i, t-a_i(t)))^2$. Thus, for a type-$j$ user who arrives at time t, its valuation about the data $Z(s_i, t-a_i(t))$ is $(\phi(s_j, t, s_i, t-a_i(t)))^2$, which completes the proof.

B. Proof of Proposition 2.

We present a polynomial-time reduction from the River Crossing Tariff Problem (RTP), confirmed as strongly NP-hard [31], to Problem 2. RTP involves selecting prices for a set of disjoint toll edges, symbolizing river bridges, to maximize profit while considering users' intent to use the paths with minimum costs for routing flows. For detailed instances of RTP, we refer to [31] due to space constraints.

By recognizing the set of toll edges as the set of data items, one can transform an instance of RTP to an instance of Problem 2 in polynomial time. Using this transformation, it is possible to solve RTP through an algorithm for Problem 2. Since RTP is NP-hard, Problem 2 is NP-hard as well, which completes the proof.

C. Proof of Proposition 3

Given AoI $\boldsymbol{a}$, if for all $j \in \mathcal{I}$, we have

$$\exp(-ha_j) \geq \max_{i\in\mathcal{I}, i\neq j} \exp(-ha_i - \epsilon\|s_i - s_j\|_2),$$

then type-j users, value the data from the source of interest, i.e., data $Z(s_j, t-a_j(t))$ most. In this scenario, the maximum social welfare (the highest revenue the platform can achieve) is

$$\Phi^*(\boldsymbol{a}) = \sum_{i=1}^{I} \lambda_i \exp(-ha_i).$$

If the platform sets the price $p_j^* = \exp(-ha_j)$ for data $Z(s_j, t - a_j(t))$ for all $j \in \mathcal{I}$, type-j users will choose to purchase data $Z(s_j, t - a_j(t))$. Otherwise, if they purchase data $Z(s_i, t - a_i(t))$ for any $i \neq j$, their payoff will be

$$\exp(-ha_i - \epsilon\|s_i - s_j\|_2) - \exp(-ha_i) < 0,$$

which is negative. Therefore, the platform's revenue in this case is $\sum_{i=1}^{I} \lambda_i \exp(-ha_i)$, which matches the maximum social welfare.

Hence, given AoI $\boldsymbol{a}$, if for all $j \in \mathcal{I}$, $\exp(-ha_j) \geq \max_{i\in\mathcal{I}, i\neq j} \exp(-ha_i - \epsilon\|s_i - s_j\|_2)$, the optimal data price is $p_j^* = \exp(-ha_j)$ for each $j \in \mathcal{I}$, which concludes the proof.

D. Proof of Theorem 1.

We first show that the algorithm DyP-Apx is polynomial and then prove its approximation ratio.

In each iteration, determining the data allocation requires $O(I^2)$ time. Once the data allocation is fixed, solving the resulting linear programming problem can be done in polynomial time [29]. With a total of I iterations, the overall complexity of the approximate algorithm DyP-Apx is polynomial.

We next prove the approximation ratio. In an iteration with the chosen valuation threshold v_{k_i}, the approximate dynamic pricing algorithm DyP-Apx only allocates data to the users whose maximum valuation is greater than v_{k_i}. The number of users whose maximum valuation is greater than v_{k_i} is at least $(I-i+1)\lambda_{\min}$ and we can have $\Phi'(\boldsymbol{a}) \geq (I-i+1)\lambda_{\min} v_{k_i}$.

Next, we bound the maximum revenue with the maximum social welfare. Given the maximum data valuation v_{k_i} of a type-k_i user, the maximum social welfare is $\sum_i \lambda_i v_{k_i}$. Since the maximum social welfare provides an upper bound on the maximum revenue that can be generated by any pricing policy, we can show

$$\Phi^*(\boldsymbol{a}) \leq \sum_i \lambda_{\max} v_{k_i} \leq \frac{\lambda_{\max}}{\lambda_{\min}} \sum_{i=1}^{I} \frac{\Phi'(\boldsymbol{a})}{I-i+1} \tag{20}$$

Since $\sum_{i=1}^{I} \frac{1}{I-i+1} \leq \ln(I)$, we can have $\Phi^*(\boldsymbol{a}) \leq \lambda_{\max}\ln(I)\Phi'(\boldsymbol{a})/\lambda_{\min}$ holds for each $\boldsymbol{a}$. Hence, we can show that $\mathbb{E}[\Phi'(\boldsymbol{a})] \geq \mathbb{E}[\Phi^*(\boldsymbol{a})]/\gamma$, where $\gamma = \lambda_{\max}\ln(I)/\lambda_{\min}$. This completes the proof.

E. Proof of Lemma 2

In the proof, we focus on the pricing of data $Z(s_1, t-a_1(t))$. The case for the data $Z(s_2, t - a_2(t))$ follows similarly.

First, we show that when the platform sets the prices at $p_1(t) = e^{-ha_1(t)}$ and $p_2(t) = e^{-ha_2(t)}$ at time t with AoI $a_1(t), a_2(t)$, type-1 users will purchase the data $Z(s_1, t-a_1(t))$, while the type-2 users will purchase the data $Z(s_2, t-a_2(t))$. Consequently, the platform's revenue is given by

$$\lambda_1 \exp(-ha_1(t)) + \lambda_2 \exp(-ha_2(t)).$$

Next, we consier two cases.

Case 1: $a_1(t) < a_2(t)$. Type-1 users' valuation about data $Z(s_1, t - a_1(t))$ is $\exp(-ha_1(t))$, which is higher than that of data $Z(s_2, t - a_2(t))$, i.e, $\exp(-ha_2(t) - d)$. Similarly, the valuation of type-2 users for the data $Z(s_1, t - a_1(t))$ is $\exp(-ha_1(t) - d)$, and its valuation for data $Z(s_2, t - a_2(t))$ is $\exp(-ha_2(t))$.

If both types of users purchase the fresher data $Z(s_1, t - a_1(t))$, the platform's revenue is at most $(\lambda_1 + \lambda_2)\exp(-ha_1(t) - d)$, and the prices are $p_1(t) = e^{-ha_1(t)-d}$ and $p_2(t) = e^{-ha_2(t)}$.

We consider the following inequality

$$(\lambda_1+\lambda_2)\exp(-ha_1(t)-d) > \lambda_1\exp(-ha_1(t))+\lambda_2\exp(-ha_2(t)). \tag{21}$$

This inequality can be rearranged to yield:

$$((\lambda_1 + \lambda_2)\exp(-d) - \lambda_1)\exp(-ha_1(t)) \geq \lambda_2\exp(-ha_2(t)). \tag{22}$$

To let (22) hold true, we require $(\lambda_1+\lambda_2)\exp(-d)-\lambda_1 > 0$ which is $\beta_1 < e^{-d}$.

Moreover, the inequality (22) holds true when

$$a_1(t) \leq a_2(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_1}{1-\beta_1}\right).$$

This implies that the optimal price $p_1^*(t) = \exp(-ha_1(t) - d)$ if $a_1(t) \leq a_2(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_1}{1-\beta_1}\right)$ and $p_1^*(t) = \exp(-ha_1(t))$ otherwise.

(i) If $\beta_1 < e^{-d}$, the optimal dynamic price is

$$p_1^*(t) = \begin{cases} e^{-ha_1(t)-d}, & \text{if } a_1(t) \leq a_2(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_1}{1-\beta_1}\right), \\ e^{-ha_1(t)}, & \text{otherwise.} \end{cases} \tag{23}$$

(ii) If $\beta_i \geq e^{-d}$, the optimal dynamic price is $p_1^*(t) = e^{-ha_1(t)}$.

Case 2: $a_1(t) \geq a_2(t)$. The valuation of type-1 users for data $Z(s_1, t-a_1(t))$ is $\exp(-ha_1(t))$, and its valuation for data $Z(s_2, t-a_2(t))$ is $\exp(-ha_2(t)-d)$. Similarly, the valuation of type-2 users for the data $Z(s_1, t-a_1(t))$ is $\exp(-ha_1(t)-d)$, and its valuation for data $Z(s_2, t-a_2(t))$ is $\exp(-ha_2(t))$.

If both types of users purchase the older data $Z(s_1, t-a_1(t))$, the platform's revenue is at most $(\lambda_1+\lambda_2)\exp(-ha_1(t)-d)$, which is smaller than $\lambda_1\exp(-ha_1(t)) + \lambda_2\exp(-ha_2(t))$. Hence, we only need to compare two scenarios: (i) Both types of users purchase the data $Z(s_2, t-a_2(t))$; (ii) Type-1 users purchase the data $Z(s_1, t-a_1(t))$ and only type-2 users purchase the data $Z(s_2, t-a_2(t))$. In both cases, the price for the data $Z(s_1, t-a_1(t))$ is $p_1^*(t) = e^{-ha_1(t)}$.

In conclusion, at time t with AoI $\boldsymbol{a}(t) = \{a_1(t), a_2(t)\}$, for the data item $Z(s_i, t-a_i(t)), i \in \{1,2\}$:

(i) If $\beta_i < e^{-d}$, the optimal dynamic price is

$$p_i^*(t) = \begin{cases} e^{-ha_i(t)-d}, & \text{if } a_i(t) \leq a_{-i}(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_i}{1-\beta_i}\right), \\ e^{-ha_i(t)}, & \text{otherwise.} \end{cases} \tag{24}$$

(ii) If $\beta_i \geq e^{-d}$, the optimal dynamic price is $p_i^*(t) = e^{-ha_i(t)}$.

F. Proof of Corollary 1

Without data data correlation, since type-i users are only interested in getting data $Z(s_i, t-a_i(t))$ and its data valuation is $e^{-ha_i(t)}$. Hence, the optimal dynamic price is $p_i^*(t) = e^{-ha_i(t)}$ and type-i users will purchase data from the source of interest.

Combining Lemma 2, we can show that if $\beta_i < e^{-d}$: then for type-i users arriving when $a_i(t) \leq a_{-i}(t) + \frac{1}{h}\ln\left(\frac{e^{-d}-\beta_i}{1-\beta_i}\right)$, the price of data $Z(s_i, t-a_i(t))$ is lower compared to a scenario without such correlation. In this case, the payoff for these users increase due to lower prices. Meanwhile, the platform's revenue increase.