Indifferential Privacy: A New Paradigm and Its Applications to Optimal Matching in Dark Pool Auctions

Antigoni Polychroniadou J.P. Morgan AI Research, J.P. Morgan AlgoCRYPT CoE New York, USA antigoni.polychroniadou@jpmorgan.com T.-H. Hubert Chan University of Hong Kong Hong Kong, China hubert@cs.hku.hk Adya Agrawal J.P. Morgan Chase Bengaluru, India adya.agrawal@jpmchase.com

ABSTRACT

Public exchanges like the New York Stock Exchange and NASDAQ act as auctioneers in a public double auction system, where buyers submit their highest bids and sellers offer their lowest asking prices, along with the number of shares (volume) they wish to trade. The auctioneer matches compatible orders and executes the trades when a match is found. However, auctioneers involved in high-volume exchanges, such as dark pools, may not always be reliable. They could exploit their position by engaging in practices like front-running or face significant conflicts of interest—ethical breaches that have frequently resulted in hefty fines and regulatory scrutiny within the financial industry.

Previous solutions, based on the use of fully homomorphic encryption (Asharov et al., AAMAS 2020), encrypt orders ensuring that information is revealed only when a match occurs. However, this approach introduces significant computational overhead, making it impractical for high-frequency trading environments such as dark pools.

In this work, we propose a new system based on differential privacy combined with lightweight encryption, offering an efficient and practical solution that mitigates the risks of an untrustworthy auctioneer. Specifically, we introduce a new concept called Indifferential Privacy, which can be of independent interest, where a user is indifferent to whether certain information is revealed after some special event, unlike standard differential privacy. For example, in an auction, it's reasonable to disclose the true volume of a trade once all of it has been matched. Moreover, our new concept of Indifferential Privacy allows for maximum matching, which is impossible with conventional differential privacy.

KEYWORDS

auctions; differential privacy; dark pools; matching

ACM Reference Format:

Antigoni Polychroniadou, T.-H. Hubert Chan, and Adya Agrawal. 2025. Indifferential Privacy: A New Paradigm and Its Applications to Optimal Matching in Dark Pool Auctions. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025,* IFAAMAS, 9 pages.



This work is licensed under a Creative Commons Attribution International 4.0 License.

1 INTRODUCTION

Dark pools are private trading venues designed for institutional investors to execute large trades anonymously, concealing details such as price and identities until after the transaction. Orders, which include the trade direction (buy or sell), volume, and price, are matched by the operator when they have opposite directions and compatible bid and ask prices. While they help prevent price swings from large orders, there are trust concerns. Operators may engage in front running, using insider knowledge of upcoming trades to execute their own trades first and profit from the price movement. Additionally, dark pool operators, often large financial institutions, may prioritize their own trades over clients', creating a conflict of interest. The lack of transparency also makes it difficult to detect manipulative practices, raising concerns about fairness. Several dark pool operators have been fined for misconduct, including misleading investors and failing to maintain proper trading practices. Notable examples include Barclays (\$70 million) and Credit Suisse (\$84.3 million) in 2016 for misrepresenting their dark pool operations, Deutsche Bank (\$3.7 million in 2017) for similar issues, ITG (\$20.3 million in 2015) for conflicts of interest, and Citigroup (\$12 million in 2018) for misleading clients about trade execution [21].

While dark pools aim to prevent the leakage of large orders, operators still gain privileged access to clients' hidden orders, creating potential for conflicts of interest or misuse of sensitive data. Recent research has focused on cryptographically protecting order information. These systems allow users to submit orders in encrypted form, enabling dark pool operators to compare orders without revealing their contents, only unveiling them when matches occur. This approach ensures greater security and mitigates risks associated with operator access to sensitive trade information.

Asharov et al. [1] introduced a secure dark pool model using Threshold Fully Homomorphic Encryption (FHE), which combines two cryptographic techniques: FHE, allowing computations on encrypted data without decryption, and Threshold Cryptography, where a secret (such as a decryption key) is split among multiple parties. In this approach, data is encrypted with a public key, and computations are performed on the encrypted data by an untrusted party. Decryption requires a threshold number of participants to combine their key shares. In Asharov et al. [1] model, orders are encrypted under a public key, and the operator matches them directly on the encrypted data. Once a match is found, the orders are decrypted by the clients using their decryption shares, ensuring that no single entity, including the operator, can access the sensitive order details. Throughout the process, the orders remain encrypted, preventing any single party from accessing both the data and the decryption key. However, FHE is known for being computationally

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

heavy, and adding a threshold mechanism increases the complexity. Optimizing this for real-time or large-scale applications is still an active research area. Moreover, the need for multiple parties to collaborate on decryption and sometimes computation can introduce significant communication overhead. As a result, the process takes nearly a full second to complete a single match, significantly impacting performance.

A recent development, Prime Match [18], introduces a solution to protect the confidentiality of periodic auctions run by a market operator. Prime Match allows users to submit orders in an encrypted form, with the operator comparing these orders through encryption and only revealing them if a match occurs. The auctions capture the trade direction (buy or sell) and the desired volume, but exclude price. Prime Match represents the first financial tool based on secure multiparty computation (MPC). In the high-stakes, highly competitive financial sector, MPC is gaining significant traction as a crucial enabler of privacy, with J.P. Morgan successfully deploying Prime Match in production.

However, in continuous double auctions without excluding the price, such as those in dark pools, where the computational complexity surpasses that of simpler periodic auctions like Prime Match, secure computation techniques fall short. They cannot support high-frequency trading within acceptable timeframes, limiting the feasibility of these methods for enhancing privacy in dark pools and the broader financial sector, where speed and efficiency are critical.

In this work, we pose the question: Can we achieve a privacypreserving solution to dark pools with efficiency comparable to nonprivate dark pool protocols? We propose a system that combines differential privacy with encryption, providing a more efficient alternative to secure MPC and FHE. Differential Privacy achieves privacy by adding noise to the results of queries or computations on datasets. The level of noise is determined by a privacy parameter, which quantifies the trade-off between privacy and accuracy. By leveraging differential privacy, our dark pool approach ensures that individual orders are obfuscated while still allowing for effective matching. This method conceals the most critical aspect of dark pools—the volumes of orders—thus preserving the primary objective of dark pools, which is to hide large trades.

In summary, integrating differential privacy with encryption offers a streamlined, efficient solution that balances privacy and computational feasibility, making it an attractive practical alternative to impractical methods like MPC and FHE.

1.1 Our Contributions:

Problem Statement: The dark pool consists of *n* agents (clients), and an operator who receives the orders from the clients. Each order takes one of two forms: (1) Buy Order: (buy, p, x), where x is the quantity/volume, and p is the highest price the buyer is willing to pay per share. (2) Sell Order: (sell, p, x), where p is the lowest price the seller is willing to accept per share. A buy order can be matched to a sell order if the buying price is at least the selling price. Our objective is to design matching protocols that maximize the total number of matches while preserving user data privacy. Specifically, we aim to conceal the quantity x of the orders during the process. Our key contributions are as follows:

- (1) Practical Dark Pool Solution: We propose an efficient solution for continuous double auctions (such as dark pools) that conceals both bid and ask quantities, effectively reducing reliance on trusted auctioneers while ensuring privacy guarantees.
- (2) Novel Privacy Concept: We introduce indifferential privacy, a new extension of differential privacy tailored to this context, which can be of independent interest with potential applicability beyond auctions and dark pools.
- (3) Maximum Matching: Our new notion of indifferential privacy allows us to achieve the optimal maximum matching which is impossible to achieve under conventional differential privcy [11].
- (4) Efficiency and Implementation: Our system significantly outperforms previous privacy-preserving auction models, which often struggled with practicality and hindered their adoption in production. We show that our solution rivals nonprivate auction protocols in terms of performance, making it viable for real-world deployment.

High-Level Idea of our techniques: To preserve privacy while matching buy and sell orders, each order is viewed as containing x units, with users submitting x + noise orders to the server based on indifferential privacy. The orders are represented as nodes in a bipartite graph, with sell orders from sellers S_i on the left and buy orders from buyers B_i on the right. See Figure 1 for an example. The server ranks the nodes based on price p, where higher prices for buyers and lower prices for sellers are considered more favorable, referred to as "extreme" prices. The algorithm then constructs a bipartite graph, with edges connecting buy and sell nodes if the buying price meets or exceeds the selling price.

In particular, the algorithm constructs and maintains a bipartite graph, where edges exist between buy and sell nodes when their prices are compatible, i.e., the buying price is at least equal to the selling price. As matches are made, isolated nodes (the gray nodes in Figure 1)—those without neighbors, such as when their price cannot be met—are promptly removed from the graph. In the maximum matching problem, the goal is to identify a set of edges where no two edges share a node, thereby maximizing the total number of matches. As mentioned above, nodes with the most popular price on one side of the graph are naturally connected to nodes with the least popular price on the opposite side. These pairs, referred to as polar opposite nodes, form the basis of an iterative matching strategy that guarantees an optimal matching solution.

Our approach maintains this optimal matching even when users submit a number of noise-injected fake (the red nodes in Figure 1) orders to obscure the true amounts of their order. To achieve this, we introduce in-differential privacy (detailed in Section 2.1) and orders are not only sorted by price but also arranged such that each user's true orders are followed by their corresponding fake orders.

Unlike traditional differential privacy, our new in-differential privacy concept allows for selective disclosure of information postmatch, aligning with realistic scenarios. For instance, in the context of a dark pool trading environment, it becomes acceptable to reveal the actual quantity of a trade once it has been fully matched, as it no longer poses a risk to privacy. To establish this new privacy framework, we integrate graph refinement techniques, ensuring



Figure 1: Maximum Matching Example. The figure illustrates a bipartite graph and its maximum matching where nodes are first sorted by price, followed by the user's genuine orders and then their fake orders. This arrangement ensures optimal matching, maximizing the number of successful pairings according to algorithm 1.

that the protocol not only protects user data but also facilitates an efficient and optimal matching process, in the presence of noise.

Implementation: We present an end-to-end implementation of our system. While previous FHE-based solutions processed fewer than one order per second, our system dramatically outperforms them, handling between 600 and 850 orders per second (according to Table 2), depending on the input volume. Furthermore, we provide an analysis of the overhead introduced by our privacy-preserving mechanism compared to the non-private version. While privacy inevitably incurs some cost and does not come for free, our system's overhead remains minimal and practical, making it highly suitable for high-frequency trading environments.

1.2 Related Work

The works of [5, 6, 15] leverage secure multiparty computation (MPC) with multiple operators instead of a single one. While this method is computationally faster than fully homomorphic encryption (FHE), it comes with significant communication overhead due to the necessary interactions among multiple operators. Moreover, the practicality of this approach is limited, as most current dark pool systems operate with a single operator. Importantly, MPC can only guarantee the privacy of orders if the dark pool operators do not collude, raising concerns in scenarios where collusion is feasible. Given these challenges, it is crucial to focus on solutions that emphasize single-operator architectures, which can streamline communication and enhance privacy.

The work of Massacci et al. [14] proposes a distributed exchange based on a multi-step functionality, including a dark pool component. Their experiments show that the system can support up to ten traders. Notably, their model does not conceal orders; instead, it discloses an aggregated list of all pending buy and sell orders, which sets it apart from our solution. Moreover, there are existing works proposing private dark pool constructions utilizing blockchain technology [2, 10, 17], our focus diverges from this area. Furthermore, all these solutions experience slowdowns due to the reliance on computationally intensive public key cryptographic mechanisms.

The work of Hsu et al. [11] also considered a private matching problem under the notion of *joint differential privacy*, where the view of the adversary consists of the output received by all users except the user whose privacy is concerned. Since the notion is still based on the conventional approach of using divergence on the adversarial views for neighboring inputs, their privacy notion can only lead to an almost optimal matching. In contrast, our new notion can achieve the exact optimal matching.

The authors in [20] employ differential privacy in a distinct and simplified setting of volume matching [3, 8, 18], where prices are predetermined and fixed, to obfuscate aggregated client volumes and conceal the trading activity of concentrated clients. In their auction mechanism, the obfuscated aggregate volumes are published daily, enabling buyers to make informed matching decisions based on this publicly available inventory.

2 PRELIMINARIES

We first describe the problem setting and the adversary model.

Problem Setup. The order of each user *i* has a type $\tau_i \in \text{buy}$, sell, a price $p_i \in \mathbb{R}+$, and a **positive** quantity $x_i \in \mathbb{Z}_+$. An input configuration $C := (\tau_i, p_i, x_i) : i \in [n]$ consists of users' orders. One unit of a buy order can be matched to one unit of a sell order if the buying price is at least the selling price. The goal is to design matching protocols that maximize the total number of matched units while preserving the privacy of users' data.

Adversarial Model. The server can observe the most amount of information, which is modeled as an adversary that is semi-honest, i.e., it will follow the protocol and try to learn about users' data. Observe that when one unit of a buy order is matched to a sell order, the identities of both the buyer and the seller, as well as their bidding prices, must be revealed. Here are possible privacy concepts.

• Given the bid (τ_i, p_i, x_i) of a user *i*, the type τ_i and the price p_i are public information, but the number x_i of units is private. However, the identity of the user *i* is known, and because it is placing a bid, it is also known that $x_i \ge 1$.

We will develop a new privacy notion that only hides the value of x_i when the bid cannot be fully executed. This theoretical privacy guarantee will mainly focus on this new notion.

• In practice, perhaps a user may not even want its presence in the system to be known if no unit of its bid is executed. We will later describe how this can be easily achieved by encrypting the user id, but this is not the main technical focus.

DEFINITION 2.1 (NEIGHBORING INPUT CONFIGURATIONS). Two input configurations C and \hat{C} are neighboring if except for one user i, all orders of other users are identical, and for user i, only the quantity may differ by at most 1, i.e. $|x_i - \hat{x}_i| \le 1$. We denote $C \sim_i \widehat{C}$ in this case.

We review some basic probability tools that are commonly used in differential privacy [9]. A divergence is used to quantify how close two distributions on the same sample space are to each other.

DEFINITION 2.2 (SYMMETRIC HOCKEY-STICK DIVERGENCE). Given distributions P and Q on the same sample space Ω and $\gamma \ge 0$, the symmetric hockey-stick divergence is defined as:

$$\mathsf{HS}_{\gamma}(P || Q) := \sup_{S \subseteq \Omega} \max\{Q(S) - \gamma \cdot P(S), P(S) - \gamma \cdot Q(S)\}.$$

REMARK 2.3. The hockey-stick divergence is related to the wellknown (ϵ, δ) -differential privacy inequality.

Specifically, $\operatorname{HS}_{e^{\epsilon}}(P|Q) \leq \delta$ if and only if for all subsets $S \subseteq \Omega$, $Q(S) \leq e^{\epsilon} \cdot P(S) + \delta$ and $P(S) \leq e^{\epsilon} \cdot Q(S) + \delta$.

DEFINITION 2.4 (DIFFERENTIAL PRIVACY). Suppose when a matching protocol is run on a configuration C, the adversary can observe some information that is denoted by some random object View(C). Then, the protocol is (ϵ, δ) -differentially private if for all neighboring input configurations C and \hat{C} ,

 $\operatorname{HS}_{e^{\epsilon}}(\operatorname{View}(C)|\operatorname{View}(\widehat{C})) \leq \delta.$

Note that the maximum number of matched units can differ by 1 for neighboring input configurations. Therefore, to use the conventional notion of differential privacy, the matching protocol cannot guarantee that an optimal number of matched units is returned. In fact, it is not hard to see that to achieve (ϵ, δ) -DP, the protocol will need to match the number of units that is about $O(\frac{1}{\epsilon} \log \frac{1}{\delta})$ smaller than the maximum possile value.

Truncated geometric distribution. Let *Z* be an even integer, and $\alpha > 1$. The truncated geometric distribution $\text{Geom}^Z(\alpha)$ has support with the integers in [0..Z] such that its probability mass function at $x \in [0..Z]$ is proportional to $\alpha^{-|\frac{Z}{2}-x|}$. Specifically, the probability mass function at $x \in [0..Z]$ is

$$\frac{\alpha-1}{\alpha+1-2\alpha^{-\frac{Z}{2}}}\cdot\alpha^{-\left|\frac{Z}{2}-x\right|}.$$

FACT 2.5 (GEOMETRIC DISTRIBUTION AND DP). For $\epsilon > 0$ and $0 \le \delta < 1$, suppose N is a random variable with distribution Geom^Z(e^{ϵ}), where $Z \ge \left\lceil \frac{2}{\epsilon} \ln \frac{1}{\delta} \right\rceil$ is even. Then, for any integer n,

 $\mathsf{HS}_{e^{\epsilon}}(n+N||n+1+N) \le \delta.$

2.1 Indifferential Privacy: A Relaxed Notion

High-Level Goal. We would like to design a protocol that always returns the optimal number of matched units, but we will relax the conventional notion of differential privacy such that the bidding quantity x_i of a user *i* does not need privacy protection if the order is fully executed.

Closest Refinement Pair. To describe this new privacy notion, we will need some technical notation. The recent work [7] considers finite sample spaces which are sufficient for our purposes.

DEFINITION 2.6 (REFINEMENT). Suppose Ω_0 and Ω_1 are sample spaces and $\mathcal{F} \subseteq \Omega_0 \times \Omega_1$ is a binary relation, which can also be interpreted as a bipartite graph $(\Omega_0 \cup \Omega_1, \mathcal{F})$. Given a distribution P_0 on Ω_0 , a refinement $\widehat{\mathsf{P}}_0$ of P_0 (with respect to \mathcal{F}) is a distribution on \mathcal{F} such that for every $i \in \Omega_0$, $\mathsf{P}_0(i) = \sum_{j:(i,j)\in\mathcal{F}} \widehat{\mathsf{P}}_0(i,j)$.

A refinement for a distribution on Ω_1 is a distribution on \mathcal{F} defined analogously.



Figure 2: Refinement Pair. The figure shows an example of a bipartite graph $(\Omega_0, \Omega_1; \mathcal{F})$, where each node has a probability mass. Each subfigure shows the probability distribution refinement on each side.

DEFINITION 2.7 (CLOSEST REFINEMENT PAIR). Given distributions P_0 and P_1 on sample spaces that are equipped with some relation \mathcal{F} and a divergence notion D, the divergence between P_0 and P_1 with respect to \mathcal{F} is defined to be:

$$\mathsf{D}^{\mathcal{F}}(\mathsf{P}_{0}\|\mathsf{P}_{1}) = \inf_{(\widehat{\mathsf{P}}_{0},\widehat{\mathsf{P}}_{1})} \mathsf{D}(\widehat{\mathsf{P}}_{0}\|\widehat{\mathsf{P}}_{1}),$$

where the infimum is taken over all refinements \widehat{P}_0 and \widehat{P}_1 of P_0 and P_1 , respectively.

REMARK 2.8. As we shall see, in our dark pool application, the relation \mathcal{F} in $\Omega_0 \times \Omega_1$, is actually much simpler. In particular, each node from one side has at most one neighbor on the other side. Hence, for the distribution on each side, there is only one possible refinement. Even though the concept of closest refinement pair is not needed for this application, we still give a more general Definition 2.11 which may be relevant in future applications.

FACT 2.9 (UNIVERSAL CLOSEST REFINEMENT PAIR [7]). Given the above distributions P_0 and P_1 , together with the relation \mathcal{F} on their corresponding sample spaces, there exists a universal refinement pair $(\widehat{P}_0, \widehat{P}_1)$ that minimizes $D(\widehat{P}_0 || \widehat{P}_1)$ for all divergences D satisfying the data-processing inequality¹ (which includes the hockey-stick divergence HS).

Intuition. To fit the above notation to our problem, let us consider a pair of neighboring input configurations C_0 and C_1 , in which the quantities of some user *i* differ by 1. For instance, in C_0 , the user *i* has $n_0 > 0$ units, while in C_1 , the user has $n_1 = n_0 + 1$ units.

For those two input configurations, the corresponding sample spaces of the adversary's views are Ω_0 and Ω_1 , which may not be the same, because the supports of the views for the two configurations may be different.

Next, let us look at each point in the sample space more carefully. As the protocol is being executed, the adversary gains more information step-by-step. Hence, each point σ in the sample space is a time-series sequence $\sigma = (\sigma_1, \sigma_2, ...)$.

¹A divergence D satisfies the data processing inequality if for any pair of joint distributions (X_0, Y_0) and (X_1, Y_1) , the corresponding marginal distributions satisfy $D(X_0||X_1) \leq D((X_0, Y_0)||(X_1, Y_1)).$

DEFINITION 2.10 (INDIFFERENCE RELATION). Given the sample spaces Ω_0 and Ω_1 of the adversary's views corresponding to neighboring input configurations C_0 and C_1 as above, we define an indifference relation $\mathcal{F} \subseteq \Omega_0 \times \Omega_1$ as follows, where a pair of time sequences $(\sigma^{(0)}, \sigma^{(1)}) \in \mathcal{F}$ is related iff

- The two sequences $\sigma^{(0)} = \sigma^{(1)}$ are the same; or
- The two time sequences $\sigma^{(0)}$ and $\sigma^{(1)}$ have a maximal common prefix ρ such that ρ implies that n_0 units of user i's order have been matched, i.e., in C_0 , the order of user i is fully executed.

With the mathematical concepts formalized, we are ready to introduce our new notion of privacy.

DEFINITION 2.11 (INDIFFERENTIAL PRIVACY (IDP)). Suppose when a matching protocol is run on a configuration C, the adversary can observe some information that is denoted by some random object View(C). Then, the protocol is (ϵ , δ)-indifferentially private (IDP) if for all neighboring input configurations C and \widehat{C} with the appropriate indifference relation \mathcal{F} defined,

 $\mathsf{HS}^{\mathcal{F}}_{\rho^{\epsilon}}(\mathsf{View}(C) \| \mathsf{View}(\widehat{C})) \leq \delta.$

Intuition. In Definition 2.10, if we only have the first bullet $\sigma^{(0)} = \sigma^{(1)}$, then this reduces to the usual differential privacy. The second bullet says the indifference relation specifies when the user *i* is indifferent about whether the adversary can distinguish between the two views in a pair $(\sigma^{(0)}, \sigma^{(1)}) \in \mathcal{F}$.

In our application scenario, this captures the idea that when a user's order is fully executed, then the privacy of its bidding quantity no longer needs protection.

Information Theoretic vs Computational IDP. When we describe our protocol, we will use the ideal functionality of cryptographic primitives and prove the privacy under Definition 2.11 that is information theoretic. If we replace the ideal functionality with the real-world cryptographic construct, we can achieve the following notion of computational IDP that is analogous to SIM-CDP introduced in Mironov et al. [16].

DEFINITION 2.12 (COMPUTATIONAL IDP (CIDP)). A protocol Π is (ϵ, δ) -CIDP if there exists a protocol Π' that is (ϵ, δ) -IDP such that Π and Π' are computationally indistinguishable.

Technical Focus. In this work, we focus on showing our protocol using ideal functionality of cryptographic primitives satisfies Definition 2.11. It is straightforward to apply standard hybrid arguments [16] to replace an ideal functionality with the real-world cryptographic primitive to show that the resulting protocol satisfies Definition 2.12.

3 INDIFFERENTIALLY PRIVATE MATCHING PROTOCOL

We describe the components of our private matching protocol. **Bid Type.** For a user *i* with Id_i , its bid has a type $\tau_i \in \{Buy, Sell\}$ with price p_i and quantity $x_i \in \mathbb{Z}_{\geq 0}$.

Bid Submission. Given the bid (Id_i, τ_i, p_i, x_i) from user *i*, suppose a user wants to achieve (ϵ, δ) -IDP. Then, it performs the following during bid submission.

- Sample non-negative noise N_i ∈ Z according to the truncated geometric distribution Geom^Z(e^ϵ), for some even Z ≥ [²/_ϵ ln ¹/_δ].
- (2) Create $y_i = x_i + N_i$ nodes of type (τ_i, p_i) , where the type is known to the server.

Out of the y_i nodes, x_i are real nodes and N_i are fake nodes. Each node contains encrypted information of whether it is real or fake.

The list of created nodes are sent to the server, where all real nodes appear before the fake nodes. Observe that the server cannot distinguish between the real and the fake nodes.

Cryptographic Assumptions. Depending on what security guarantee of the final protocol is needed, we can assume different properties of the ciphertext.

• To achieve just our notion of indifferential privacy, the encrypted information of a node can be achieved by a non-malleable commitment scheme [12]. In practice, we can also use a hash function such as AES.

Ideal Commitment Scheme. For ease of exposition, we describe our protocol using an ideal commitment scheme. A user creates a commitment of a message, which is an opaque object whose contents the adversary cannot observe. At a later time, the user can choose to open the commitment, which must return the original message. In particular, a cryptographic commitment scheme has these key properties: (1) Hiding: The value stays secret until revealed. (2) Binding: The committed value can't be changed. (3) Unforgeability: No one can forge or tamper with the commitment. (4) Correctness: The committed value is always correctly revealed.

• Assuming the shuffler model, the server does not know which nodes come from which user. Hence, in this case, each node also contains encrypted information about the user ID. However, the server still knows which nodes originate from the same user, and the nodes from the same user are sorted in a list such that real nodes appear first before the fake nodes.

Node Popularity. In any case, the server can create an ordering on the nodes on the buyer and the seller sides according to the price. For a buy node, a higher price is more popular; for a sell node, a lower price is more popular. On each side, the most or the least popular price is known as an *extreme* price.

Matching Graph. The algorithm maintains a bipartite graph $G = (\mathcal{B}, \mathcal{S}; E)$, where there is an edge between a buy and a sell node if the prices are compatible, i.e., the buying price is at least the selling price. As some nodes are matched during the process, we assume that the algorithm will immediately remove any isolated node that no longer has a neighbor.

Maximum Matching Problem. Given a bipartite graph $G = (\mathcal{B}, S; E)$, a matching $M \subseteq E$ is a subset of edges such that no two edges in M are incident on the same node. The *maximum matching problem* aims to find a matching M with the maximum number of edges. We next describe a strategy to find a maximum matching on a bipartite graph induced by buyer and seller nodes. **Polar Opposite.** Suppose the current matching graph G has no isolated nodes. For an arbitrary side, observe that any node on this

side with the most popular price has an edge connected to any node on the other side with the least popular price. We say that two such nodes from the two sides are *polar opposite* of each other. The following fact shows that an optimal matching can be returned by iteratively matching polar opposites.

FACT 3.1. Suppose a matching graph G has no fake or isolated nodes, and u and v are any two nodes from the different sides that are polar opposite of each other. Then, there exists a maximum matching in G in which u and v are matched.

PROOF. Without loss of generality, suppose *u* has the most popular price on its side, while v has the least popular price on its side. Suppose M is a maximum matching in which u and v are not matched to each other. We will modify M without decreasing the matching size such that *u* and *v* are matched to each other in the following steps.

- (1) If *u* is not matched to any node, we can make *u* replace any other node on its side, because it has the most popular price. Hence, we may assume that u is matched in M.
- (2) If v is not matched in M, then we can replace the partner of *u* with *v* to make *u* and *v* matched to each other.
- (3) Otherwise, we have u is matched to some v' and v is matched to some u'. Observe that if u' is compatible with the least popular price on the other side, then it must be compatible with v'. Hence, u and u' can swap partners such that u and v become matched.

Matching with Fake Nodes. With our assumption, for each node type, we assume that the nodes are sorted such that the nodes from the same user are together and its real nodes appear first.

In each step, the server picks nodes u and v from the two sides that are polar opposite of each other. The owners of nodes u and vwill participate in the following protocol:

- (1) If the node of an owner is real, the owner will open its commitment to reveal the node is real.
- (2) If the node of an owner is fake, this implies that all real nodes of that owner are already matched and the owner will reveal all its fake nodes.
- (3) If both the nodes *u* and *v* are opened to be real, the nodes *u* and *v* are mathced and removed from the matching graph.
- (4) If one of the nodes is real and the other node is fake, the real node will be considered in the next iteration if it still has a polar opposite in the remaining graph.

Correctness Intuition. The protocol is outlined in Algorithm 1; note that any arbitrary unspecified choice made by the algorithm can either be randomized or deterministic according to some additional rules.

LEMMA 3.2. Algorithm 1 returns a maximum matching between real buy and sell nodes in the matching graph.

PROOF. Comparing with the case with no fake nodes, observe that whenever a fake node is encountered, it will be removed immediately. Hence, the matching behavior is exactly the same as the case with no fake nodes. From Fact 3.1, a maximum matching is returned. П

1 Matching graph $G \leftarrow (\mathcal{B}, \mathcal{S}; E)$

 $_2 \ u \leftarrow \mathsf{NULL}$

3	$M \leftarrow \emptyset$				
4	4 while $E \neq \emptyset$ do				
5	if $u = \text{NULL then}$				
6	From any side and any extreme price on that side,				
7	end				
,	Salast the payt node a that is a palar apposite to u				
8	Select the next node b that is a polar opposite to u.				
9	The algorithm announces that an attempt is made to match <i>u</i> and <i>v</i> .				
10	The owners of the two nodes open their commitments				
	(if they have not already done so) and reveal whether the nodes are real or fake.				
11	The algorithm infers that if an owner reveals a fake				
	node, all the remaining nodes by the same owner are				
	also fake.				
12	if either u or v is fake then				
13	Any fake node and its incident edges are removed				
	from G.				
14	else				
15	if both u and v are real then				
16	The pair (u, v) is matched and added to M ;				
	remove u and v from G .				
17	end				
18	end				
19	Remove any isolated node from <i>G</i> .				
20	If exactly one of <i>u</i> and <i>v</i> is still unmatched and remains				
	in the graph, set <i>u</i> to be that node; otherwise, set <i>u</i> to				
	NULL.				
21	21 end				
22	22 return matching M				

3.1 Privacy Analysis

We show that our dark pool auction protocol satisfies IDP.

Neighboring Input Configurations. Recall that we consider neighboring input configurations C_0 and C_1 in which exactly one user *i* has different bidding quantities. Suppose in C_0 , the quantity is n_0 and in C_1 , the quantity is $n_1 = n_0 + 1$.

Adversarial View. We decompose the view space into $\Gamma \times \Lambda$, where Γ corresponds to the number of nodes created by user *i*, and Λ includes (i) the bid submissions by all other users, and (ii) any random bits used to make choices in the matching procedure in Algorithm 1. The distributions on Γ differ in C_0 and C_1 , but in the two scenarios, the distributions on Λ are identical and independent of the Γ component.

Note that any other information observed by the adversary can be recovered by a point in $\Gamma \times \Lambda$.

Indifference Relation. We define an almost trivial relation on $\Gamma \times \Lambda$, namely, $(\gamma_0, \lambda_0) \sim (\gamma_1, \lambda_1)$ iff $\gamma_0 = \gamma_1$ and $\lambda_0 = \lambda_1$. This means that the same number of nodes are submitted by user *i* in the two scenarios; moreover, all other users submit exactly the same bids in the two scenarios, and the also the same random bits are used in the matching process.

However, this does not mean that the views of the adversary are the same in both scenarios, because the number of real nodes for user *i* are different in C_0 and C_1 . Nevertheless, if the views are different, it must be because n_0 real nodes of user *i* have been matched in both scenarios, and the algorithm tries to match the next node from user *i*. This is consistent with the indifference relation described in Definition 2.10.

LEMMA 3.3. Using an ideal commitment scheme, the dark pool auction protocol is (ϵ, δ) -IDP.

PROOF. Recall that from Definition 2.11, our goal is to prove an upper bound for: $\mathsf{HS}_{\epsilon}^{\mathcal{F}}(\mathsf{View}(C_0) || \mathsf{View}(C_1)).$

Because of the ideal commitment scheme, during the bid submission, the server can only see how many nodes are created by each user. From user *i*, the information collected at this stage is a point in Γ . Suppose that in C_0 , the distribution of the number of created nodes by user *i* is P_0 ; and in C_1 , the corresponding distribution is P_1 . Note that P_0 and P_1 are both distributions on Γ . From Fact 2.5, we have: $HS_{e^{\epsilon}}(P_0||P_1) \leq \delta$.

According to the above discussion, for both C_0 and C_1 , we have the same distribution Q on Λ that represents other users' bids and the random bits used by the server during the matching process.

As argued above, a point in $\Gamma \times \Lambda$ is sufficient to recover the view of the adversary. Hence, we have:

 $\mathsf{HS}^{\mathcal{F}}_{\boldsymbol{\rho}^{\epsilon}}(\mathsf{View}(C_0) \| \mathsf{View}(C_1)) = \mathsf{HS}_{\boldsymbol{\rho}^{\epsilon}}((P_0, Q) \| (P_1, Q)).$

Moreover, observe that if $(\gamma_0, \lambda_0) = (\gamma_1, \lambda_1)$ correspond to two points in $\Gamma \times \Lambda$ under configurations C_0 and C_1 , respectively, it must be the case that during bid submission, exactly $\gamma_0 = \gamma_1$ nodes are created by user *i*. Furthermore, everything is identical in both scenarios, except that the $(n_0 + 1)$ -st node in user *i*'s list is fake in C_0 and is real in C_1 . Therefore, the adversary either has identical views in both scenarios (because at most n_0 nodes in user *i*'s list have been attempted to be matched), or else it must be the case that all real nodes of user *i* in C_0 have been fully matched; this is consistent with the indifference relation \mathcal{F} .

Note that Q is independent of the Γ component. In general, the hockey-stick divergence (as well as other commonly used divergences) satisfies the property that observing the same extra independent randomness should not change the value of the divergence. Hence, we have: $HS_{e^{\epsilon}}((P_0, Q)||(P_1, Q)) = HS_{e^{\epsilon}}(P_0||P_1)$, which we have already shown is at most δ . This completes the proof. \Box

COROLLARY 3.4. Replacing the ideal commitment scheme with a real-world commitment scheme (such as [12]), the dark pool auction protocol is (ϵ, δ) -computational IDP.

4 IMPLEMENTATION AND EVALUATION

4.1 Implementation

We provide a comprehensive end-to-end implementation of our indifferentially private dark pool auction. We have also implemented a non-private auction to compare the total computation and communication time. Our code is available at https://github. com/adyaagrawal/idp-darkpool. Both the non-private and Indifferentially Private (IDP) protocols were incorporated into ABIDES [4],² an open-source high-fidelity simulator tailored for AI research in financial markets such as stock exchanges. ABIDES is ideal for this purpose, as it supports simulations with tens of thousands of clients interacting with a central server for transaction processing, along with customizable pairwise network latencies to simulate real-world communication delays. We run the simulations on a personal x64-based Windows PC equipped with a single Intel Core i5-10210U processor running at 1.6 GHz and 16 GB of DDR4 memory.

ABIDES employs the cubic network delay model, where the latency is determined by a base delay (within a specified range) and a jitter that influences the percentage of messages arriving within a designated timeframe, thereby shaping the tail of the delay distribution. Our experiments were conducted using three distinct network settings: local (client machines in NYC), global (client machines from NYC to Sydney), and very wide network (client machines across the world). For each setting, we configured the base delay according to Table 1 while employing the ABIDES's default parameters for jitter.

Network Setting	Base Delay (ms)
Local (Within NYC)	0.021 - 0.1
Global (NYC to Sydney)	21 - 53
Very Wide Network (Across the World)	10 - 100

Table 1: Base delay for different network configurations.

In our non-private implementation, clients submit their orders without concealing their identities. These orders are organized into two doubly linked lists: one for buy orders and another for sell orders. Our implementation follows the maximum matching algorithm explained in Section 3. Once matches are identified, they are sent back to the clients for execution.

For our IDP implementation, clients generate an array of sorted orders, with a few randomly numbered fake ones to ensure indifferential privacy. They commit to their identity and whether the order is real or fake using the Cryptodome library in Python based on AES. The maximum matching algorithm is employed and if the match contains the order that the individual client has placed, the client opens the commitment to the server to reveal whether the order is real or fake. Then, the server requests the client to reveal the identity or tries to find a new match. Once both the parties have opened their identity and revealed it to the server, the orders are executed and the next round of matching starts.

4.2 Experimentation

Performance comparison with non-private implementation. To evaluate the performance of our order matching system, we compared the total time required to match orders using both the non-private and Indifferentially Private (IDP) implementations in all three different network settings. We conducted experiments with a range of parties each generating 2³ orders, starting from 4 parties (2⁵ orders) and extending up to 1024 parties (2¹³ orders).

²ABIDES has also been used in simulating privacy preserving federated learning protocols such as the most recent work of [13].



Figure 3: Comparison of Time Taken Across Different Settings.

The data generation process is designed to ensure that each set of randomly generated orders consistently achieves a matching rate of 70% to 80%.³ For instance, each client generates 8 sorted orders, with 5 to 7 being real and 1 to 3 being fake. The prices for buy orders are randomly selected between 99 and 101, while sell orders are priced between 98 and 100.

From Figure 3, we can see that the total time taken using indifferential privacy auction does not deviate greatly from the non-private real world implementation. In Table 2 we also mention the orders per second for two different scenarios.

Performance comparison with FHE implementation. Considering throughput as the number of orders per second, FHE-based solutions can only process around 0.07 orders per second, as shown in Table 4 of [15] for 40 orders, using threshold fully homomorphic encryption (tFHE) implemented via the GPU FHE library of [3]. Moreover, the actual runtime is even longer, as Table 4 does not account for the additional overhead from threshold decryption and key management, which increases with the number of clients (as highlighted in Table 2 of [1]).

For context, according to [1], running a comparison on encrypted data for a large amount of orders, 2¹³ orders, takes 7.6 seconds on CPU, while our system processes and matches all orders in 13.32 seconds in total (see Table 2), not just performing a comparison on a single order. The FHE-based solution requires at least one comparison operation per matching attempt, adding to its computational burden. Furthermore, the FHE-based timings from [15] are measured on GPUs; actual performance on CPUs would be even slower, whereas our experiments are conducted entirely on CPUs.

This illustrates that FHE solutions are an overkill, even with the benefit of GPU acceleration, while our approach offers a practical solution. Last but not least, as shown in Table 4 of [15], the latest multi-operator solution, based on secure multiparty computation, is capable of processing only 26 orders per second, even on highperformance hardware.

Scalability with larger datasets. In the experiments above, we compared the run-times of our protocol with a real-world auction protocol for up to 1,024 clients, each submitting 8 orders (resulting in a total of 2¹³ orders). We have also extended our experiments to include scenarios where both the number of orders per client and the total number of clients double incrementally, scaling up to a total of 262K orders. Our results show that the differentially private

Orders	Non-Private Time (secs)	IDP Time (secs)
40	0.019698	0.046887
2^{13}	3.887790	13.328184

Table 2: Comparison of Non-Private and IDP total running times for a small (40) and large size (2^{13}) of orders. The throughput (orders per second) for the non-private case is 2031 and 2107 and for the indifferential private case is 853 and 615 for 40 and 2^{13} orders, respectively.

protocol scales linearly with the increased workload. We provide these additional benchmarks in the full version [19] of the paper.

5 CONCLUSION

In this work, we addressed the limitations of existing privacypreserving auctions in high-frequency trading, such as dark pools, where auctioneers can be untrustworthy. Previous methods, like fully homomorphic encryption, were impractical due to their overhead. Our approach, based on the new notion of Indifferential Privacy, provides an efficient, privacy-preserving continuous double auction that enables maximum matching while minimizing risks. This makes our system a practical and secure alternative addressing both performance and security concerns in modern trading.

ACKNOWLEDGMENTS

T-H. Hubert Chan was partially supported by the Hong Kong RGC grants 17201823 and 17202121. This paper was prepared for informational purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co. and its affiliates ("JP Morgan") and is not a product of the Research Department of JP Morgan. JP Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

³When the % of matches is lower, the total running time naturally decreases, as fewer matches result in less overhead.

REFERENCES

- [1] Gilad Asharov, Tucker Hybinette Balch, Antigoni Polychroniadou, and Manuela Veloso. 2020. Privacy-Preserving Dark Pools. 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, 1747–1749. https://doi.org/10. 5555/3398761.3398969
- [2] Samiran Bag, Feng Hao, Siamak F Shahandashti, and Indranil Ghosh Ray. 2019. SEAL: Sealed-bid auction without auctioneers. *IEEE Transactions on Information Forensics and Security* 15 (2019), 2042–2052.
- [3] Tucker Balch, Benjamin E. Diamond, and Antigoni Polychroniadou. 2020. SecretMatch: inventory matching from fully homomorphic encryption. In ICAIF '20: The First ACM International Conference on AI in Finance, New York, NY, USA, October 15-16, 2020, Tucker Balch (Ed.). ACM, 15:1–15:7. https://doi.org/10.1145/ 3383455.3422569
- [4] David Byrd, Maria Hybinette, and Tucker Hybinette Balch. 2020. ABIDES: Towards High-Fidelity Multi-Agent Market Simulation. In Proceedings of the 2019 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation, SIGSIM-PADS 2020, Miami, FL, USA, June 15-17, 2020, Jason Liu, Philippe J. Giabbanelli, and Christopher D. Carothers (Eds.). ACM, 11–22. https://doi.org/10.1145/3384441. 3395986
- [5] John Cartlidge, Nigel P. Smart, and Younes Talibi Alaoui. 2019. MPC Joins The Dark Side. In Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security, AsiaCCS 2019, Auckland, New Zealand, July 09-12, 2019, Steven D. Galbraith, Giovanni Russello, Willy Susilo, Dieter Gollmann, Engin Kirda, and Zhenkai Liang (Eds.). ACM, 148–159. https://doi.org/10.1145/3321705. 3329809
- [6] John Cartlidge, Nigel P. Smart, and Younes Talibi Alaoui. 2021. Multi-party computation mechanism for anonymous equity block trading: A secure implementation of turquoise plato uncross. *Intell. Syst. Account. Finance Manag.* 28, 4 (2021), 239–267. https://doi.org/10.1002/ISAF.1502
- [7] T.-H. Hubert Chan and Quan Xue. 2024. Symmetric Splendor: Unraveling Universally Closest Refinements and Fisher Market Equilibrium through Density-Friendly Decomposition. *CoRR* abs/2406.17964 (2024).
- [8] Mariana Botelho da Gama, John Cartlidge, Antigoni Polychroniadou, Nigel P. Smart, and Younes Talibi Alaoui. 2022. Kicking-the-Bucket: Fast Privacy-Preserving Trading Using Buckets. In Financial Cryptography and Data Security -26th International Conference, FC 2022, Grenada, May 2-6, 2022, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 13411), Ittay Eyal and Juan A. Garay (Eds.). Springer, 20–37. https://doi.org/10.1007/978-3-031-18283-9_2
- [9] Cynthia Dwork. 2006. Differential Privacy. In ICALP (2) (Lecture Notes in Computer Science, Vol. 4052). Springer, 1–12.
- [10] Hisham S Galal and Amr M Youssef. 2021. Publicly verifiable and secrecy preserving periodic auctions. In *Financial Cryptography and Data Security. FC 2021*

International Workshops: CoDecFin, DeFi, VOTING, and WTSC, Virtual Event, March 5, 2021, Revised Selected Papers 25. Springer, 348–363.

- [11] Justin Hsu, Zhiyi Huang, Aaron Roth, Tim Roughgarden, and Zhiwei Steven Wu. 2016. Private Matchings and Allocations. SIAM J. Comput. 45, 6 (2016), 1953–1984.
- [12] Huijia Lin and Rafael Pass. 2015. Constant-Round Nonmalleable Commitments from Any One-Way Function. J. ACM 62, 1 (2015), 5:1–5:30.
- [13] Yiping Ma, Jess Woods, Sebastian Angel, Antigoni Polychroniadou, and Tal Rabin. 2023. Flamingo: Multi-Round Single-Server Secure Aggregation with Applications to Private Federated Learning. In 44th IEEE Symposium on Security and Privacy, SP 2023, San Francisco, CA, USA, May 21-25, 2023. IEEE, 477–496. https://doi.org/10.1109/SP46215.2023.10179434
- [14] Fabio Massacci, Chan Nam Ngo, Jing Nie, Daniele Venturi, and Julian Williams. 2018. FuturesMEX: secure, distributed futures market exchange. In 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 335–353.
- [15] Sahar Mazloom, Benjamin E. Diamond, Antigoni Polychroniadou, and Tucker Balch. 2023. An Efficient Data-Independent Priority Queue and its Application to Dark Pools. Proc. Priv. Enhancing Technol. 2023, 2 (2023), 5–22. https://doi. org/10.56553/POPETS-2023-0038
- [16] Ilya Mironov, Omkant Pandey, Omer Reingold, and Salil P. Vadhan. 2009. Computational Differential Privacy. In CRYPTO (Lecture Notes in Computer Science, Vol. 5677). Springer, 126–142.
- [17] Chan Nam Ngo, Fabio Massacci, Florian Kerschbaum, and Julian Williams. 2021. Practical witness-key-agreement for blockchain-based dark pools financial trading. In Financial Cryptography and Data Security: 25th International Conference, FC 2021, Virtual Event, March 1–5, 2021, Revised Selected Papers, Part II 25. Springer, 579–598.
- [18] Antigoni Polychroniadou, Gilad Asharov, Benjamin Diamond, Tucker Balch, Hans Buehler, Richard Hua, Suwen Gu, Greg Gimler, and Manuela Veloso. 2023. Prime Match: A {Privacy-Preserving} Inventory Matching System. In 32nd USENIX Security Symposium (USENIX Security 23). 6417–6434.
 [19] Antigoni Polychroniadou, T.-H. Hubert Chan, and Adya Agrawal. 2025. Indif-
- [19] Antigoni Polychroniadou, T.-H. Hubert Chan, and Adya Agrawal. 2025. Indifferential Privacy: A New Paradigm and Its Applications to Optimal Matching in Dark Pool Auctions. arXiv:2502.13415 [cs.CR] https://arxiv.org/abs/2502.13415
- [20] Antigoni Polychroniadou, Gabriele Cipriani, Richard Hua, and Tucker Balch. 2024. Atlas-X Equity Financing: Unlocking New Methods to Securely Obfuscate Axe Inventory Data Based on Differential Privacy. In Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AA-MAS 2024, Auckland, New Zealand, May 6-10, 2024, Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum (Eds.). International Foundation for Autonomous Agents and Multiagent Systems / ACM, 1585–1592. https://doi.org/10.5555/3635637.3663019
- [21] Sec. [n. d.]. Dark Pool Misrepresentations. https://www.sec.gov/newsroom/pressreleases/2018-193.