Cryptographic Combinatorial Securities Exchanges

Christopher Thorpe and David C. Parkes

Harvard University School of Engineering and Applied Sciences
cat@seas.harvard.edu, parkes@seas.harvard.edu

Abstract. We present a useful new mechanism that facilitates the atomic exchange of many large baskets of securities in a combinatorial exchange. Cryptography prevents information about the securities in the baskets from being exploited, enhancing trust. Our exchange offers institutions who wish to trade large positions a new alternative to existing methods of block trading: they can reduce transaction costs by taking advantage of other institutions' available liquidity, while third party liquidity providers guarantee execution—preserving their desired portfolio composition at all times. In our exchange, institutions submit encrypted orders which are crossed, leaving a "remainder". The exchange proves facts about the portfolio risk of this remainder to third party liquidity providers without revealing the securities in the remainder, the knowledge of which could also be exploited. The third parties learn either (depending on the setting) the portfolio risk parameters of the remainder itself, or how their own portfolio risk would change if they were to incorporate the remainder into a portfolio they submit. In one setting, these third parties submit bids on the commission, and the winner supplies necessary liquidity for the entire exchange to clear. This guaranteed clearing, coupled with external price discovery from the primary markets for the securities, sidesteps difficult combinatorial optimization problems. This latter method of proving how taking on the remainder would *change* risk parameters of one's own portfolio, without revealing the remainder's contents or its own risk parameters, is a useful protocol of independent interest.

1 Introduction

In [21] we introduced the idea of a cryptographic securities exchange for individual equities, motivated by the unfavorable price impact and possible exploitation of information associated with block trades.[1] In that paper, we consider an exchange of single securities, and, typically, securities are traded as single asset types in most alternative trading systems.

We now introduce the *cryptographic combinatorial securities exchange,* where entire *baskets* of securities may be bought or sold, rather than single positions. This has important applications for portfolios of securities where entering various positions piecemeal would subject the investor to portfolio risk. Specifically, if a large portfolio is optimized to have certain correlations among its assets, and it takes hours or days to find a counterparty to fill each of various positions in a basket trade that liquidates a percentage of or rebalances that portfolio, the correlations no longer hold whenever one order is

[1] Exchanges of very large positions of securities.

R. Dingledine and P. Golle (Eds.): FC 2009, LNCS 5628, pp. 285–304, 2009.

filled before another order. Our exchange, which provides for atomic trades that are guaranteed to clear, eliminates this execution risk on portfolio balance.

Another benefit of the cryptographic combinatorial exchange is that cryptography hides valuable information about intended trades that can be exploited. As described in detail in our previous work [21], knowledge of investors' upcoming trades is often exploited – and has a measurable price impact. It would likely be impossible to operate a combinatorial securities exchange without cryptography, because few institutions would trust any third party with the details of their intended trades "in the clear". Our solution employs cryptography as well as hardware and network security to build an exchange that protects the secrecy of institutions' trades before and after the exchange takes place.

We complete our introduction with a discussion of existing commercial protocols and related work from the finance and cryptography literature. In Section 2 we define the cryptographic combinatorial securities exchange. Section 3 describes our proofs of portfolio risk on an encrypted basket of securities that represents the net holdings after multiple baskets are combined in a transaction. In Sections 4, 5 and 6 we discuss real-world requirements our exchange might have in determining reasonable exchange fees, protecting the exchange from exploitative trading practices, and securing data after a round of the exchange is over. An appendix includes efficiency calculations showing that a Paillier-based cryptosystem permits a practical implementation of our protocol, and further discussion on calculating optimal fees and commissions for participants in the exchange.

1.1 Existing Commercial Protocols

While many existing alternative trading systems (ATS's) exist for block trades, no existing ATS protects traders' information and guarantees atomic execution of baskets of securities. Institutions still fear that knowledge of their liquidity can be exploited in various ways, and rely on information broker ATS's like Liquidnet who strictly limit membership to the trading network to parties who are only trading for liquidity reasons. A second problem with many ATS's is that there is typically no guarantee of execution.

We work to ameliorate all of these concerns: our proposal enhances trust by not only keeping trades secret until the market is to clear but also proving the results correct; it also improves liquidity by giving the exchange an efficient mechanism to guarantee execution for all of the trades submitted to it— while still keeping the particular equities in the incoming institutions' baskets secret; and it provides an atomic basket trading paradigm.

Currently for large basket trades (involving more than one security), the transactions are too complex for the pairwise trade matching that existing ATS's like Liquidnet and Pipeline offer. Institutions who need to trade a basket of securities atomically to maintain the integrity of a diversified portfolio may not wish to undertake the risk of executing the trades one security at a time. Thus, institutional investors who wish to trade several large positions at once in a *basket order* often hire an investment bank. They describe the basket to a small number of trusted investment banks who agree to provide liquidity, without disclosing the exact securities that comprise the basket in advance— information that could be exploited. When deciding how much to charge for liquidating

a basket, the banks learn only certain risk parameters, such as index membership, daily trading volume, and market correlation; these enable them to estimate their risk and costs in the absence of complete data. This process takes some time: typically institutions will send information about a basket to a liquidity provider in the morning, who then analyzes the information and replies within hours.

Our new cryptographic combinatorial exchange provides the improved efficiency of institution-to-institution trading with the reduced portfolio risk from guaranteed execution of atomic basket trades. Cryptography makes such an exchange feasible by providing necessary trust: exploitable data remain secret, and every action and result can be proven correct.

In our combinatorial exchange, institutions submit baskets of buy and sell orders which are filled by other institutions' sell and buy orders (respectively). The unfilled orders comprise a remainder basket, which clears the exchange when filled by a cooperating third party (assumed to be an investment bank). Prices for each security can be determined by the primary markets, so that the exchange need only discover trading interest.

We believe this to be the first characterization of a cryptographic combinatorial exchange: a number of participants submit bundles to buy and sell goods (in our example, securities), and the market finds an optimal allocation of trades to maximize the benefit of all participants. While such combinatorial exchanges typically require significant computation to find optimal allocations,[2] our exchange makes two important simplifications that eliminate the hard combinatorial problem. First, prices are defined externally by the primary markets, and second, our clearing of the remainder via a third party means that all bundles are filled and the market clears at equilibrium.

1.2 Related Work

Bossaerts et al. [1] describe a "combined-value trading mechanism" similar to our approach and survey related work from the finance literature. We argue that one important reason that such mechanisms have still not been adopted is because institutions are unwilling to divulge the composition of their baskets. Cryptography solves this problem, and may well hold the answer to implementing more expressive trading mechanisms in practice.

Szydlo [20] first proposed the application of zero-knowledge proofs to disclosing facts about equities portfolios. In his highly relevant and pioneering work, a hedge fund proves that its portfolio complies with its published risk guidelines without revealing the contents of its portfolio. Szydlo's proofs are not situated in a transactional context, but rather in the context of a hedge fund reporting portfolio risk characteristics that are based on the claimed securities in its portfolio. In our case, we are interested in proving portfolio risk on a portfolio derived from combining baskets of securities, for example, in order to liquidate a newly derived remainder basket computed from a combination of many incoming baskets.

Another difference in our work is the use of encryption over commitments. Encryptions allow the exchange to issue proofs about combinations of the institutions' baskets

[2] Indeed, even defining "optimal" in such an exchange is challenging!

without requiring their continued involvement. Were we to employ commitments, we would require institutions to decommit their baskets before computing the remainder; this provides an opportunity for repudiation. While the homomorphic Pedersen commitments Szydlo employs are more efficient than homomorphic encryptions, we desire nonrepudiation: once a basket is committed to in a transaction, the institution may not later refuse to reveal that basket. Since any non-repudiatable commitment is equivalent to an encryption,[3] we elect to employ encryptions directly. This may also mitigate so-called protocol completion incentive problems (see [3] for a related discussion in the context of auctions), because institutions who lose their incentive to participate cannot benefit from refusing to complete the protocol.

While surprisingly little academic research has been published on applications of cryptography in securities trading (see [21] for a discussion), more work has been done on combinatorial exchanges (CE's). In a CE, buyers and sellers come together in a common exchange to trade bundles of various goods (where bundles may have instructions to buy or sell, or both.) In the general case, solving the price and winner determination problems in a combinatorial exchange is extremely difficult; in our cryptographic combinatorial securities exchange, we get around these by taking all prices from the fair prices already established by the primary markets (price determination), and employing "liquidity providers" who guarantee enough liquidity for the entire exchange to clear (winner determination). See Parkes et al. [13], and Smith et al. [19] for a formal treatment of combinatorial exchanges and related work.

2 Cryptographic Combinatorial Securities Exchanges

Our cryptographic combinatorial securities exchange offers basket traders guaranteed execution and efficient liquidity discovery. It keeps information completely secret until it is necessary, eliminating opportunities for fraud, and proves every result correct without revealing unnecessary information.

Our protocol is simple: institutions submit encrypted baskets; the exchange closes; the exchange creates an encrypted remainder and proves risk characteristics to third party liquidity providers; these liquidity providers bid on their commission; and the winning provider clears the market by liquidating the remainder. Prices clear at prices determined by the primary markets.

Any basic cryptographic protocols supporting provably correct, secrecy-preserving computation over private inputs, such as those described in [15,16,21], are sufficient to construct our exchange. As our protocol does not depend on specific features, such as a particular homomorphism, we do not burden our exposition with specific implementation details. Rather, we assume implementors of our protocol will select an underlying cryptosystem appropriate to their specific needs at the time.

Moreover, these protocols are practically efficient and support the calculations of risk and interval proofs essential to our protocol. To verify this claim, we implemented the cryptographic operations necessary to conduct our protocol and report results in

[3] To enjoy nonrepudiation, a commitment must be deterministically invertible. A function that is binding, hiding, and invertible (presumably via some secret) is clearly equivalent to an encryption.

Appendix A. We discuss the implications of the partial trust in our third party required by these protocols and mechanisms for mitigating such trust in Section 6.

2.1 Preliminaries

We employ the following primitive operations necessary to reveal the portfolio risk profile:

- Prove that a ciphertext is the encrypted result of a polynomial function over multiple encrypted values and/or constants x, y. We write $E(x) \oplus E(y)$ to signify the computation yielding $E(x+y)$; $E(x) \otimes E(y)$ yields $E(x \times y)$. $E(X) \odot E(Y)$ signifies the "dot product" of vectors X and Y of encrypted values. In addition, $\bigoplus_i E(x_i)$ yields $E(\sum_i x_i)$.
- Prove whether one encrypted value is greater than another. We write $E(x) \trianglelefteq E(y)$ to signify the computation proving that $x \leq y$ given the two encryptions; we use analagous notation for the other inequality operators.
- Prove whether one encrypted value is (not) equal to another.

If a homomorphic cryptosystem is used for the computations, such as the system described by Paillier [11] and elaborated in [5] and [15], then additional preparation is required to prove results of computations employing both additions and multiplications. Since no known cryptosystem is doubly homomorphic,[4] we require instead that whatever underlying cryptosystem is employed support proofs of correct computation of both addition and multiplication. In a homomorphic cryptosystem, a verifier would check one operation by direct computation over ciphertexts, and the other by receiving information from the prover. For example, using Paillier encryption, a verifier could check addition by simply multiplying ciphertexts; she would only be able to check multiplication with the help of a prover using (non-interactive) protocols such as those described in [5,15].

We assume that interactive interval proofs (see, for example, [2,9,15]) can also be performed efficiently in a non-interactive setting using the Fiat-Shamir heuristic [7]; a strong cryptographic hash of input data simulates the verifier's actions during an interactive proof. Since the encrypted inputs are probabilistic encryptions generated by independent parties, the output of a suitable cryptographic hash on those values should yield data with sufficient (apparent) randomness.

2.2 Problem Definition

We construct a protocol to operate a cryptographic combinatorial securities exchange in which multiple parties may exchange baskets of securities while limiting exploitation of any information submitted to the exchange. The participants in the protocol include the "exchange" itself, "institutions" who submit basket orders to the exchange, and "liquidity providers" who clear unfilled orders. The institutions, liquidity providers, and external auditors also, as "verifiers", verify the accuracy of any information promulgated

[4] That is, there exist two distinct operations over the space of ciphertexts that correspond directly to addition *and* multiplication over the space of plaintexts.

by the exchange. When describing a protocol to communicate the risk of accepting a basket of securities, we refer to the "institutions" who send the basket to a "recipient" counterparty. We employ these functional terms throughout our work.

Before a specified "closing time", each participating institution publishes an encrypted basket of securities it wishes to liquidate. Before the closing time, the exchange may not decrypt the baskets; after that time, baskets may not be withdrawn or modified, and execution is guaranteed by the exchange.

The exchange then computes the remainder necessary for the exchange to reach equilibrium, i.e. the basket filling all trading interest not met by other parties. It reveals information about this remainder to various third-party "liquidity providers" who have agreed to liquidate large remainder baskets for the exchange; they in turn quote a price or liquidating the remainder.

The information provided might be direct risk analysis measurements on the remainder, or it might reveal the differences in risk incorporating the remainder would have on a sample portfolio provided by each third party. The liquidity providers then submit encrypted bids for liquidating the portfolio, and the exchange accepts the best price and issues a zero-knowledge style proof to all participating institutions and liquidity providers that it is optimal.

In practice today, liquidating these large basket trades takes hours or even days. Millisecond execution time is critical for high-frequency trading of single securities, but not for these relatively infrequent but high-value transactions that occur only several times a week and are based on liquidity, not price fluctuations. Thus, the cryptographic operations required to implement such an exchange are within reach of contemporary commodity computing hardware. See Appendix A for example calculations.

The exchange preserves the secrecy of the institutions' identities by acting as the middleman between all transactions. In our current setting, institutions may be known to participate in the exchange by virtue of their publishing encrypted baskets, but they can hide whether they are trading or not each day by submitting empty baskets on days they do not wish to trade. Where even further anonymity is desired (that is, the exchange never learns the institutions' identities), real-world entities, such as law or accounting firms, can be employed to represent the institutions; constructing a cryptographic protocol to preserve institutions' identities is beyond the scope of the present work. See [6] for one approach to the problem of privacy in securities exchanges.

This implies the following desiderata:

- The information in the baskets must remain secret, even from the exchange, until all baskets have been submitted.
- Once baskets have been submitted, they may not be modified or retracted.
- No party other than the exchange may learn anything about the direct composition of the baskets other than what is implied by any disclosures, including risk information sent to liquidity providers.
- The exchange must clear completely, that is, all orders are guaranteed to be filled.
- The exchange must clear efficiently: any computations must be completed within a few hours at reasonable cost.

– The cryptosystem employed can convince an independent verifier that the result of performing a computation on hidden inputs is either a particular value or lies in a range of values.

2.3 The Protocol

We consider n institutions P_i, where $i \in [1,n]$, each of which submits an integer vector (representing a basket) B_i, comprised of m integers (representing securities) S_j, where $j \in [1,m]$. Thus in a universe of 6 securities, B_3, P_3's basket, might be $\langle 0,-20000,32000,0,45000,0\rangle$. We assume the exchange operates on a fixed universe of these m commonly traded and reasonably liquid securities, such as listed equities, standardized options, and government securities. The double subscript notation B_{ij} denotes the (unencrypted) quantity of security j in P_i's basket; in our example, $B_{35} = 45000$. $E(B_{ij})$ is the encrypted form of one such value. Zeroes are included to hide the number of distinct equities in the basket.

We assume a public price vector V of length m contains the values for the m securities at the time the exchange clears; V_j is the price for security j. This might be obtained from current market prices or the previous day's closing prices.

Since most underlying cryptosystems employ modular arithmetic, short positions can be easily represented as "negative numbers" (that is, very large numbers that are the additive inverses of the corresponding positive number). Alternatively, long and short positions may be represented by two encrypted vectors: one of the absolute values of the quantites and the other of 1 (long), -1 (short), or 0 (no position).

An encryption of a basket of equities is simply an integer vector one for each equity in the universe, including zeros. For visual comfort, we may write $E(B_i)$ as the encryption of an entire basket, which is in fact m separate encryptions: $\langle E(B_{i1}),E(B_{i2}),\ldots,E(B_{im})\rangle$.

Step 1. The exchange announces clearing times, the universe of equities to be traded on the exchange, and any rules governing the composition of baskets participating in the exchange. If time-lapse cryptography (TLC) [17] or another technique used to enforce nonrepudiation requires posting of public information (for example, a public TLC encryption key), the exchange posts it.

Step 2. Before each clearing time, each institution P_i chooses which equities she wishes to trade and creates basket B_i and its encrypted form $E(B_i)$. She then creates a commitment to her basket, $\mathbf{Com}_i(E(B_i))$, and publishes that commitment where the exchange and other parties to the transaction can see them. The reason we add this pre-clearing commitment step is to prevent the exchange from observing the contents of any baskets and revealing that information before the "clearing time". This extra step ensures that the exchange cannot influence the outcome of the exchange even if it can somehow successfully leak data, because no baskes may be submitted or retracted after the auctioneer receives any material information.

Step 3. When the clearing time is reached, the institutions decommit: each institution P_i publishes $E(B_i)$, the encryption of its basket, and any additional information necessary to verify $\mathbf{Com}_i(E(B_i))$ matches. If a institution fails to decommit, and a

nonrepudiation technique is employed, the commitment is forced open and the encryption of his basket is published.[5]

Step 4. Either the exchange, or each institution P_i, proves, using the now public $E(B_i)$, that B_i conforms to any announced basket composition requirements by proving a set of constraints on the encrypted number of shares of each security in the universe. These constraints can take the form of any equation or inequality representing a polynomial function of the encrypted baskets (security quantity vectors) B_i, public price vector V, and necessary constants. These constants might include minimum or maximum basket size, or a constant bound for what percentage of the basket is in a particular class (such as market sector or index member). Because P_i encrypted the basket itself, it is capable of proving its basket meets any such constraints (see Section 5) without the cooperation of the exchange, if necessary.

Step 5. Anyone can verify the "remainder" basket B_0 as above by computing its encrypted form from $E(B_i)$ (for all i). Table 1 illustrates an example of this on unencrypted values. Using our notation from Section 2.1, we write:

$$B_0 = \langle \bigoplus_{i=1}^{n} E(B_{i1}), \ldots, \bigoplus_{i=1}^{n} E(B_{im}) \rangle = \langle E(\sum_{i=1}^{n} B_{i1}), \ldots, E(\sum_{i=1}^{n} B_{im}) \rangle$$

Table 1. Example set of cross-clearing portfolios $B_1, \ldots, B_4$ with a "remainder" B_0

Security	B_1		B_2		B_3		B_4		B_0
ABC	+500		-200		0		0		+300
DEF	+300		-800		+300		+200		0
GHI	0	+	+100	+	-300	+	0	=	-200
JKL	+200		0		-400		+300		+100
MNO	-800		0		+500		0		-300

Step 6. The exchange privately decrypts the baskets, and obtains the unencrypted remainder basket.

Step 7. The exchange proves the constraints about the composition of the remainder basket B_0 to the third party liquidity providers, who individually or jointly determine transaction costs for the remainder basket and agree to provide liquidity to the pool.

Step 8. After the market-clearing liquidity has been secured, the exchange announces the protocol is complete and the market clears at prices fixed in accordance with a published standard procedure.

For example, the market might clear at the midpoint between the bid and ask quoted on the current primary market, or an agreement to trade at the volume-weighted average

[5] An alternative to the use of commitments is to employ distributed key generation for a public encryption key, then only reconstruct the private key after the clearing time is reached; this idea, formalized in TLC, still ensures that the exchange cannot decrypt the baskets prematurely.

price for a particular period of time. The mechanics of clearing securities trades are beyond the scope of this work; we assume that all parties trade with a trusted intermediary who accepts all securities sold and distributes those bought, clearing the market.

The exchange issues proofs that the procedures are followed, again by proving that a set of constraints are met over the institutions' encrypted baskets, the public price vector, necessary constants, and any (possibly encrypted) data provided by liquidity providers.

3 Secrecy-Preserving Proofs of Impact on Portfolio Risk

In the introduction, we describe how large basket orders are traded by revealing portfolio risk measurements of the baskets themselves, rather than the actual risk undertaken by the liquidity providers the baskets.

We propose a secure system that makes price discovery for basket trades more accurate by offering liquidity providers limited but more specific characteristics of their actual risks — how the risk of their inventory changes — not the characteristics of the incoming basket. In this section, we refer to an "institution" who is offering a basket and a "recipient" of that basket – a liquidity provider in our primary protocol. However, our protocol has more general applicability and may be used in any transaction in which a recipient wishes to estimate its risk in accepting a basket of equities. That basket may be the combination of many baskets (e.g. in a combinatorial exchange) or a single counterparty's basket.

Our protocol employs a server as a partially trusted third party, accepting encrypted forms of the institution's portfolio and the provider's book, and providing a set of risk characteristics of the recipient's resulting book after the integration of the equities in the portfolio. The protocol proves these characteristics correct in a zero-knowledge fashion based on the encrypted inputs, to assure the recipient that it received an accurate picture even if it does not win the bid. (Presently, only winners can verify the correctness of the submitted values because they are the only party who ever discovers the actual contents of the basket.)

Finally, we remark that wherever we refer to a recipient's "inventory", the recipient may use any representative portfolio in the protocol and compute the risk of accepting the basket on the basis of risk changes in this particular portfolio. This may be due to reluctance to reveal the exact portfolio to even a partially trusted third party, or to optimize price discovery by a specially tailored portfolio.

3.1 Mechanics of the Protocol

The protocol is comprised of a series of simple steps: the parameters of the transaction are agreed on; the transacting parties publish their encrypted information to all; the "institution" and "recipients" P_i for $i \in [1,n]$ send information to the partially trusted third party, the "exchange"; the exchange issues proofs to the recipient about its portfolio risk; and the recipients verify the proofs using the published information. When used in conjunction with the above protocol, the "institution's" basket is the remainder basket representing all unfilled orders.

Step 1. The institution and recipients agree on a set of risk characteristics to evaluate the portfolio resulting from each recipient's accepting the institution's portfolio. This

protects the secrecy of the institution's information while providing enough information to the recipient to quote an accurate price. Each risk characteristic will be computed by performing a computation over the institution's encrypted portfolio and recipient's encrypted inventory. The institution may also require that certain outputs be reported as "bounds", where the results are only quoted accurately enough for the recipient to price the portfolio by proving they lie within a certain small range. This is of extreme importance to prevent any recipient from "backing out" private information from the encrypted data by carefully constructed queries. See also the more detailed discussion in the following section, 3.2.

Step 2. The institution prepares an encrypted basket B_0 as above in the combinatorial case. The encryptions are carried out in accordance with the underlying cryptographic protocol.[6] The institution submits the encrypted basket to the exchange.

Step 3. Each recipient prepares a similar basket B_i with its inventory, into which the basket would be integrated, and shares this encrypted portfolio with the exchange. It does *not* need to share it with the institution.

Step 4. The exchange and each recipient computes the encrypted result of incorporating the new basket B_0, $\hat{B}_i = B_i \oplus B_0$. The exchange then computes the risk characteristics of $\hat{B}_i$ and reveals them to recipient P_i with a correctness proof. Note that P_i never learns the exact composition of $\hat{B}_i$: only its risk profile.

Step 5. When the protocol is used to compute the cost of liquidating a basket of securities (for example, a remainder basket), the recipient examines the new risk characteristics of the resulting portfolio, estimates carrying and execution costs and submits a bid to the institution. (In practice, the computed characteristics might be sent to a portfolio management software system that compares the "before" and "after" portfolios to automatically estimate risk and hedging costs.)

3.2 What Information Should Be Revealed?

Presently, institutions submit the characteristics of their baskets to investment banks in spreadsheets with specific numbers in each category. This process "leaks" information, especially where the number of equities in a particular category is small. Occasionally, the information can create obvious implications: for example, if there is only one equity listed in the telecommunications sector, comprising 89,000 shares whose total value is $3,546,650, the bank probably has an excellent idea of the company's name. Institutions sometimes "white out" some information in their basket descriptions to prevent such information leakage, usually to eliminate obvious information leaks.

Yet even when such information is redacted, rigorous statistical analyses of the information submitted can still yield information about the composition of the baskets, and this is also possible in more complex situations where a large number of equities contribute to one line-item. Since values are often supplied to the penny, if the number of equities, total dollar amount as of a particular market close, and total number

[6] Providing the value quotation is a matter of convenience, as the encrypted value can be computed as the encrypted product of public previous close price and the encrypted number of shares.

of shares is known, it is possible that a computer could efficiently search the possible baskets created by equities in that sector and propose a small number of alternatives to the bank. While we have no reason to believe that the reports are being so exploited by the banks, eliminating any potential information leakage while still providing accurate risk assessments is an important benefit of our proposed protocol.

Because the cryptographic framework we describe supports interval proofs on encrypted values (or functions on encrypted values) the exchange can reveal approximate risk characteristics that are sufficient for price discovery but are more resistant to statistical analysis to back out the composition of the baskets. For instance, instead of reporting the sector breakdown exactly, the exchange can report values rounded to the nearest percentage point or thousands of dollars or shares. Although there is no reason that institutions can't submit baskets with such obfuscated data, they would not be able to prove it correct without cryptography. The ability to reveal "just enough" information (while still proving it correct) is an important feature of our proposal.[7]

3.3 How the Information Is Revealed

Rather than proving portfolio risk of a single portfolio, we are interested in revealing facts about a hypothetical portfolio that results from the combination of other portfolios.

Once our protocol is followed, the exchange privately knows the combined portfolio. To reveal a fact, the exchange obtains the result of the desired computation and sends the result to the verifiers, along with special verification data that allow them to verify the result.

3.4 Revealing Portfolio Value and Dividends

In most cases, the incoming basket order will involve long and short trades, and an important element of the risk is the "skew" — the difference between the total value of the short and long trades. Sometimes, when an institution is trading a basket with a significant skew (or even entirely one-sided) it may not wish the size of the skew to be known. In this case, the recipient might respond not with a specific cash price, but rather a discount quotation, an agreement to accept the equities in the basket at a particular volume-weighted average price, or other quotation based on the market prices of the equities after they are revealed. Because the recipient can accurately assess its risk profile in accepting these, it can offer more competitive discounts or execution quotes for less risky baskets, or, similarly, charge more for a riskier basket.

The institution and the recipient(s) may agree to reveal:

- The full value of the long and short sides of the portfolio:
  The exchange provides a proof that allows the recipient to decrypt the sum of all long positions and the sum of all short positions.[8]

[7] See Section 6 for a discussion of why this feature is best supported by protocols based on a partially trusted third party.

[8] While possible, the details of doing this without revealing *which* securities are long and short require great care and describing such a proof is beyond the scope of this paper.

- The value or range of the "skew" only:
  In this case, the exchange provides the recipient a proof of the sum of the portfolio's value: all long positions' values minus all short positions' values. Assuming that $\hat{B}_i$ holds signed quantities, the verifier simply computes the encrypted dot product of the portfolio and the price vector V: $E(W) \equiv \hat{B}_i \odot V$. The exchange might reveal the precise value W, or only that W lies within a particular interval.
- No information about the value of the incoming basket:
  In this case, the position values, quotes, and number of shares must all be kept secret; the risk profile of the resulting portfolio can still be evaluated by other means.

A similar approach can be applied to dividends, where the recipient receives aggregate calculations of historical and expected dividend payments, so that it can estimate any dividend payments it will make (for short sales) and receive (for long positions).

3.5 Portfolio Composition Statistics

For risk management and hedging calculations, the recipient may wish to know the composition of the combined portfolio based on various factors, including:

- Market sector (technology, health care, consumer goods, etc.)
- Market capitalization
- Index membership
- Dividend amount (as a percentage of share price)
- Average daily trading volume (possible in terms of both shares and notional value)
- Historical price volatility

Using our protocol, the institution need not reveal any information about the incoming basket's sector breakdown — for example, if there are balanced long and short trades in technology, and zero trades in utilities, this is indistinguishable to the recipient from a portfolio with zero technology and balanced utilities trades, provided that the balanced trades do not change the risk profile of the recipient's inventory. This provides additional secrecy to the institution while still meeting the needs of the recipient.

The exchange calculates the portfolio composition and proves it to the accepting recipient, who verifies the result using its own encrypted portfolio and the encrypted basket provided by the institution. Because the exchange can offer proofs that each sector's breakdown lies within a particular interval (say to the percentage point or 1/10 of 1%), the institution can reveal enough information for the recipient to offer an accurate price while making reconstruction of the portfolio infeasible.

Using the general cryptographic operations described above, the exchange can prove breakdowns for the various aspects of the portfolio as follows. We write that the portfolio B_0 is the sum of all n institutions' baskets B_i for all $i \in [1,n]$, each of which contains m securities. B_{i_j} is the jth security in basket i.

Step 1. Because the exchange knows the breakdown for each equity (e.g. market capitalization, market sector, etc.), it can compute encrypted sums of the number of shares and total value for each item in the breakdown by summing up the encrypted number of shares and total value from the combined portfolio and prove them correct. The recipient also recalls the encrypted total number of shares and encrypted total value of the

basket. We recall that this is the *combined* portfolio, where any long and short trades in the incoming basket have already been incorporated into the recipient's inventory.

Step 2. The exchange first proves the sums are correct, namely, $E(B_{0j}) \equiv \bigoplus_{i=1}^{n}(B_{ij})$, for $j \in [1,m]$; and computes the encrypted total portfolio value $E(W) \equiv \bigoplus_{j=1}^{m} B_{0j} \cdot V_j$ from the encrypted combined portfolio and constant price vector.

Step 3. The exchange then prepares an encrypted "unit size" Z by computing Z and designating a public constant K such that $ZK \leq W$ and $(Z+1)K > W$. The exchange proves this by providing the recipient $E(Z)$ and a trivial encryption $E(K)$ and proving that $E(Z) \otimes E(K) \trianglelefteq E(W)$ and $(E(Z) \oplus E(1)) \otimes E(K) \rhd E(W)$. Thus there are K "units" of size Z in the breakdown.[9]

Step 4. For each element of the breakdown, the exchange prepares an interval proof of how many "units" that element comprises. It begins by calculating and revealing two integer constants a_i, b_i and their "trivial" encryptions $E(a_i), E(b_i)$; the recipient can verify these are correct encryptions. For example, a_i might be 10 and b_i 12, to show the result is between 10 and 12 units.

Step 5. The exchange completes the interval proof, showing that $E(a_i) \otimes E(Z) \trianglelefteq E(v_i) \trianglelefteq E(b_i) \otimes E(Z)$. This proves that $a_i Z \leq v_i \leq b_i Z$. This bounds the value of the portfolio in bucket i without revealing any further information.

Step 6. Steps 4 and 5 are repeated for each "bucket" in the breakdown until the entire portfolio has been classified. The recipient might check that $\sum_i a_i \leq K \leq \sum_i b_i$ to be sure that the breakdown provided is appropriate.

3.6 Other Measurements of Risk

Because of the flexibility of the mathematical operations that can be performed on the recipient's basket and the incoming basket, other, more complicated risk measurements are possible. While the above examples are of linear functions, which permit the recipient to compute the incoming baskets' risk characteristics from the output risk characteristics and his own inputs, our protocol provides for computation of polynomial functions of modest degree by using repeated multiplications (including repeated squaring) of encrypted values to calculate exponents. This permits the computation of more complex risk analysis measurements whose definition under our framework we leave for future work.

4 Pricing and Payment

Two types of prices must be computed: the price at which each security is valued when the exchange clears, and the price that the third parties charge for providing the market-clearing liquidity. We treat these in turn, referring to the winning third party (which

[9] Care must be taken so that $W \mod K$ is not too large, because this could skew the results. The exchange can even show the recipient that value by revealing the verifiable result $E(W) \ominus (E(K) \otimes E(Z))$, or proving that it is less than a small constant. Since K is public, the recipient can refuse a K that is too small.

might be a consortium) as the liquidity provider or recipient. We note that if our second protocol is used independently between a single institution and one or more liquidity providers for proving characteristics about a single basket trade, the institution's basket functions as the remainder.[10]

Because each of the securities in the exchange is presumed to be traded on a primary market, we adopt the common practice in block trading to allow the primary market to dictate a fair market price for the securities at the time of trading. The financial industry uses many reasonable methods for price determination in block trading, and we do not advocate a particular pricing model over another—provided that the trading prices are determined in a manner exogenous to the exchange. Examples of these methods include the closing or settlement price for the day of the transaction, average prices over time such as the volume-weighted average price (VWAP), or simply the midpoint of the best bid and offer at the time the market clears.

After the proofs are obtained, the third parties have learned enough information to calculate a price for the incoming basket. They can accurately assess the changes in risk on their own inventories if they accept the basket, and by measuring those changes, estimate hedging costs for equities it will carry and execution costs for unwinding the trades it does not wish to keep.

In Appendix B, we consider approaches to allocating the liquidation costs among the market participants; this can also be done in a provably correct fashion.

5 Keeping the Pool Safe

Although our methods are designed to provide transparency without revealing exploitable information, there remain ways in which unscrupulous traders might try to exploit the exchange we propose.

One misuse of our exchange might be for institutions to use its guaranteed liquidity to unload especially high-risk or illiquid securities. If the exchange becomes filled with undesirable assets, then liquidity providers will be less likely to want to participate. This is an important reason we advocate a pricing mechanism that charges institutions according to the amount of the remainder basket their trades represent—if the pricing mechanism is correctly defined, then institutions who submit less desirable portfolios will pay more for their liquidation costs.

Yet it might be desirable to make sure that the baskets the institutions submit to the exchange meet basic criteria for acceptability and portfolio risk. Using the same portfolio risk analysis techniques described above, institutions can issue zero-knowledge proofs about the baskets they submit so that all can be confident that their trades are acceptable. This should also reduce the third-party liquidation costs, because the third parties will be more confident that they won't receive a basket that has nice overall characteristics but might be comprised of less desirable individual securities.

As we mentioned in the introduction, other common exploits associated with dark pools are less of a concern because our protocol features guaranteed execution. Exploits such as probing for existing liquidity and baiting (where someone places an order and

[10] In fact, this is equivalent to operating our exchange with a single institutional participant.

then retracts it) are less of a problem, since once an order is placed, it cannot be retracted, and learning that your order was filled reveals nothing about existing opposite interest—every order is filled. Johnson [8] describes "toxic dark pools" that are known for being exploited.

6 Strengthening Secrecy

While our solutions offer an appropriate degree of secrecy and are practical to implement, the exchange does learn private data that it could reveal to others after the fact. It learns the trades that took place, which may be undesirable to certain institutions (notably hedge funds), and could learn something about the recipient's inventory in the context of proving changes to the recipient's risk without revealing the incoming portfolio characteristics directly. While the trades must eventually be reported to the exchanges and become a matter of public record, and no such information could have any bearing on a particular round of the exchange, this information still has value. We thus consider how to mitigate the trust not to leak any information that we might place in the exchange operator.

The most compelling complement to our cryptographic solutions includes secure computing infrastructure such as Trusted Computing [18] hardware and network monitoring. We advance this idea in our previous work on cryptographic securities trading [21] and auctions protocols [14,16]. In this scheme, specially designed hardware and software are trusted not to leak information, and monitored for security. Moreover, the secrecy-preserving correctness proofs we advance in this work complement such "black boxes" extremely well, because we need not trust the black box to produce correct results: we only use it to mitigate *ex post* disclosure. Thus, the actions of the exchange remain provably correct under all circumstances—even an undetected bug in the black box cannot result in incorrect behavior.

Even in these high-security settings, a determined adversary might be able to engineer steganographic leaks by "hiding" information in the protocol itself, often in predetermined bits of "random" help values. Doing so would be a significant effort, because most trusted computing infrastructures will not run software that has not been verified and signed by a third party, but we mention that small risk nonetheless. Fair Zero-Knowledge, introduced by Lepinski et al. [10], describes a mechanism to combat such attacks and surveys related work.

Another approach is be to distribute trust among a group of entities who jointly act as the exchange. While this theoretically possible solution does eliminate any one single trusted third party, the architecture retains a functional entity of a trusted third party which happens to be comprised of several entities. Employing such a solution successfully in practice would require the cooperation of disparate, disinterested business entities to prevent collusion; moreover, the efficiency of such secure multiparty computation schemes may not be able to support the computations we require.

Finally, we observe that perfect security is never attainable in real life where humans are involved: any dishonest party "in the know" can always pick up the phone to deliver an out-of-band information leak. And, even where there is no intentional disclosure, Brandt and Sandholm proved impossibility results for achieving complete secrecy

in some auction settings [4]. These ideas lead to interesting security questions about modern markets where more and more trades are performed without human input: automated trading agents running on secure hardware could offer an unprecedented level of security against the human element.

7 Conclusions and Future Work

We have implemented a useful new mechanism for block trading of securities that meets two market requirements: institutions can trade directly with each other when liquidity is available, while still having guaranteed execution for their entire order to limit portfolio and carrying risk. We employ a combinatorial exchange model, but make it tractable through external price discovery and a third party who provides necessary liquidity to achieve market equilibrium so that all orders are filled.

We protect the secrecy of sensitive data while giving the third party information necessary to calculate a fair commission by combining two novel cryptographic protocols. They are efficient, straightforward to understand, and can be implemented using already accepted cryptographic primitives.

More general formulations of these protocols may be of independent interest. Consider an arbitrary function over a finite field with encrypted inputs and a prover who proves facts about the output of this function. Clearly, there are many functions for which a precise output reduces the space of possible inputs dramatically — an unintended consequence of revealing a single output. Our mechanisms can offer provably correct yet *approximate* outputs using interval proofs, where exact results would reveal too much information.

The protocol we describe to prove changes to a recipient's risk also generalizes into a new class of price discovery. We can construct a more general protocol that allows a buyer to evaluate a purchase on the basis of a change in a buyer's utility function, rather than calculating the utility of the good directly. This means that in many business settings, where direct revelation of the good in question might have negative consequences, a buyer can engage in "zero-knowledge due diligence" where the buyer can satisfy many concerns by learning about how her utility function changes based on incorporating the good into her possessions, without learning enough about the good to allow the information to be exploited. These settings might include the sale of a significant commercial building, a business unit of a large corporation, or, other methods of trading financial instruments.

We leave for future work a number of mechanism design questions. We believe it is possible to approach a true combinatorial exchange in which both institutions and liquidity providers post their desired baskets, where institutions post a maximum price they are willing to pay for liquidating their baskets, and whether and how their baskets are divisible; liquidity providers post "chunks" of liquidity associated with transaction costs for each chunk. The exchange then finds the optimal feasible allocation satisfying all possible atomic trades, and proves the outcome correct. Moreover, the use of such "chunks" could significantly reduce the size of any remainder basket, thereby reducing the size of any portfolio that needs to be traded blindly.

In addition to generalizing the protocols as described here, future work may also include a reference implementation of a prototype exchange or a more detailed technical specification based on a particular cryptosystem.

Acknowledgments

We thank Stephanie Borynack and Imad Labban, who brought our attention to an important class of large transations described in this work. We also thank Eric Budish, John Y. Campbell, and Luis Viceira for useful discussions about how this research might be used in practice, and ideas that improved the presentation of the work to readers with an economics background. We thank Aggelos Kiayias for suggesting a number of improvements we adopted in preparing our work for FC 2009. Finally, we thank the anonymous reviewers who evaluated various versions of this paper and offered many helpful suggestions we have incorporated into this work.

References

1. Bossaerts, P., Fine, L., Ledyard, J.: Inducing liquidity in thin financial markets through combined-value trading mechanisms. European Economic Review 46(9), 1671–1695 (2002)
2. Boudot, F.: Efficient proofs that a committed number lies in an interval. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 431–444. Springer, Heidelberg (2000)
3. Bradford, P.G., Park, S., Rothkopf, M.H.: Protocol completion incentive problems in cryptographic Vickrey auctions. Technical Report RRR 3-2004, Rutgers Center for Operations Research, RUTCOR (2004)
4. Brandt, F., Sandholm, T.: (Im)possibility of unconditionally privacy-preserving auctions. In: Proc. 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems, pp. 810–817 (2004)
5. Damgård, I., Jurik, M.: A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In: Proceedings of Public Key Cryptography 2001 (2001)
6. Di Crescenzo, G.: Privacy for the stock market. In: Syverson, P.F. (ed.) FC 2001. LNCS, vol. 2339, p. 259. Springer, Heidelberg (2002)
7. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
8. Johnson, J., Tabb, L.: Groping in the dark: Navigating crossing networks and other dark pools of liquidity, January 31 (2007)
9. Kiayias, A., Yung, M.: Efficient cryptographic protocols realizing e-markets with price discrimination. In: Financial Cryptography and Data Security, pp. 311–325 (2006)
10. Lepinski, M., Micali, S., Shelat, A.: Fair zero-knowledge. In: Proc. Theory of Cryptography Conference, pp. 245–263 (2005)
11. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–239. Springer, Heidelberg (1999)
12. Parkes, D.C., Cavallo, R., Elprin, N., Juda, A., Lahaie, S., Lubin, B., Michael, L., Shneidman, J., Sultan, H.: ICE: An iterative combinatorial exchange. In: ACM Conf. on Electronic Commerce, pp. 249–258 (2005)
13. Parkes, D.C., Kalagnanam, J.R., Eso, M.: Achieving budget-balance with Vickrey-based payment schemes in combinatorial exchanges. Technical report, IBM Research Report RC 22218 (2001)

14. Parkes, D.C., Rabin, M.O., Shieber, S.M., Thorpe, C.A.: Practical secrecy-preserving, verifiably correct and trustworthy auctions. In: ICEC 2006: Proceedings of the 8th international conference on Electronic commerce, pp. 70–81. ACM Press, New York (2006)
15. Parkes, D.C., Rabin, M.O., Shieber, S.M., Thorpe, C.A.: Practical secrecy-preserving, verifiably correct and trustworthy auctions. Electronic Commerce Research and Applications (to appear, 2008)
16. Rabin, M.O., Servedio, R.A., Thorpe, C.: Highly efficient secrecy-preserving proofs of correctness of computations and applications. In: Proc. IEEE Symposium on Logic in Computer Science (2007)
17. Rabin, M.O., Thorpe, C.: Time-lapse cryptography. Technical Report TR-22-06, Harvard University School of Engineering and Computer Science (2006)
18. Smith, S.W.: Trusted Computing Platforms: Design and Applications. Springer, New York (2005)
19. Smith, T., Sandholm, T., Simmons, R.: Constructing and clearing combinatorial exchanges using preference elicitation. In: AAAI 2002 workshop on Preferences in AI and CP: Symbolic Approaches (2002)
20. Szydlo, M.: Risk assurance for hedge funds using zero knowledge proofs. In: S. Patrick, A., Yung, M. (eds.) FC 2005. LNCS, vol. 3570, pp. 156–171. Springer, Heidelberg (2005)
21. Thorpe, C., Parkes, D.C.: Cryptographic securities exchanges. In: Dietrich, S., Dhamija, R. (eds.) FC 2007 and USEC 2007. LNCS, vol. 4886, pp. 163–178. Springer, Heidelberg (2007)

A Efficiency of Our Protocols

While we have observed that any number of cryptographic systems might support our protocols, we have conducted empirical tests using Paillier cryptography libraries written in C++ with the GMP multi-precision library; we wrote these libraries to test the practicality of cryptographic auctions in [15].

Notably, these tests included interval proofs, additions, and multiplications, all of which are required to operate a cryptographic combinatorial securities exchange. Our empirical tests demonstrate that our efficiency claims are realistic, namely, that each step of the protocol can proceed in a reasonable amount of time on cost-effective commodity hardware. As noted above, we expect our combinatorial exchange to clear high value baskets within hours; our tests meet this goal.

We assumed a universe of 3,000 securities in each basket. We assumed that quantities of securities are 32-bit values (up to approximately 4 billion). We used a 1536-bit Paillier key, a composite of two 768-bit primes that offers expected security for at least a few years. We assume all four processors are running in a quad-core Intel Xeon 2.0GHz processor. Obviously, cryptographic computations can be parallelized across many machines; this can offer even greater speed at additional hardware cost.

- Encrypting a basket: 48 seconds
- Decrypting a basket: 15 seconds
- Computing/Verifying the encrypted remainder: ¡ 1 second
- Interval proof on a 32-bit value: 1.25 seconds of required server precomputation; 0.25 seconds of real-time server computation; 1.25 seconds of client verification
- Performing additions: negligible
- Multiplication with a constant: 0.001 seconds

- Multiplication (proving an encrypted value represents the product of two other encrypted values): 4.3 server seconds; ¡ 1 second of client verification
- Proving a basket of 3,000 securities is "well-formed": 1 hour of required server precomputation; 12 minutes of real-time server computation; 1 hour of client verification

Using these values, we anticipate a typical risk analysis measurement would assume a basket already proven to be well-formed, and perhaps 10 interval proofs and 10 multiplications. This means that for a particular basket (say, the remainder), a risk analysis measurement, such as a breakdown into 10 market sectors, could be performed in less than 1 minute of server and client time. This puts our protocol well within the realm of practicality.

The majority of time spent using a Paillier cryptosystem is in modular exponentiation of random help values. Using a specialized cryptographic coprocessor could significantly reduce computation time. Moreover, in many cases these computations can be precomputed before the exchange clears, and fully verified in the hours after it clears – clearly, if the exchange can be found out to have cheated within a day, that is a significant enough deterrent so that the verification operations need not be carried out in real time.

B Allocating Liquidation Costs

The liquidity provider can be compensated in many ways; the simplest is for it to quote a brokerage commission that it accepts for executing the trades. A provider who perceives greater risk can charge a higher commission. Other pricing mechanisms are possible: if the cash value of the portfolio is revealed, the provider can quote a price based on that; if the skew is not revealed, then the provider can quote a price based on a discount factor or volume-weighted price after the transaction is agreed on. The institution can choose among the various providers' offers, and notify the winner. Once the transaction is complete, the liquidity provider accepting the basket will be able to verify that the information provided was correct when it receives the remainder portfolio — but we reiterate that an advantage of our protocol is that those that do not win still have convincing proof that the information was correct: the institution can't favor one bank over another.

Another interesting possibility is for the liquidity providers to publish deterministically verifiable valuation functions for their risk premium calculations. Using these, they can submit a representative portfolio to the exchange, obtain the changes in risk on their portfolio, then the exchange runs their calculations on the encrypted risk data and publishes a verifiable, encrypted result. These results would then be used to prove the payments correct, or could even be used in a verifiable sealed-bid auction to prove which of the liquidity providers' calculations yielded the most competitive bid for liquidating the remainder.

While total cost sharing is simple and convenient, we also consider a slightly more involved "pay for what you use" model: each institution pays its share of the commission based only on the benefit it derived from the securities provided by the liquidity providers. In this method, institutions that use more of the remainder (instead of the other institutions) to fill their trades pay a greater share of the commission. At the extremes, an institution that trades securities which do not appear in the remainder pays

nothing, while an institution who is the only one trading a particular security pays the entire share of the commission for that security.

We illustrate this method with an example which refers back to Table 1. For simplicity, we will assume that each security trades at a price of \$1, and the liquidity provider charged a commission of \$9000. The notional values of the four institutions' baskets are \$1800, \$1100, \$1500, and \$500, respectively; the remainder basket's value is \$900. The exchange operator then publishes the encrypted amounts of commission paid based on the pro rata notional value traded of each security: \$3000 for ABC, \$0 for DEF, \$2000 for GHI, \$1000 for JKL, and \$3000 for MNO. The operator proves that their sum is the (public) total commission.

Next, the exchange operator proves the total trading interest for each security by publishing encrypted sums of the absolute notional value of the orders in each basket: 700 for ABC, 1600 for DEF, 400 for GHI, 900 for JKL, and 1300 for MNO. Then, using the above methods, the exchange operator can publish an encrypted breakdown of the commission to be paid per share.[11] In this case, the commissions work out to \$429 per 100 shares of ABC, \$0 per 100 shares of DEF, \$500 per 100 shares of GHI, \$112 per 100 shares of JKL, and \$231 per 100 shares of MNO; this yields a total overcharge of \$14 due to rounding error.[12] The exchange proves that these encrypted prorated commissions are correct given the encrypted values already computed.

The exchange finally uses these encrypted prorated commissions to give each institution a verifiable share of its commission without revealing the magnitude of the securities traded by other institutions or the composition of the remainder basket. For example, Institution 1 would pay

$$(5 \times 429) + (3 \times 0) + (0 \times 500) + (2 \times 112) + (8 \times 231) = 4217.$$

The others would pay \$1358, \$3103, and \$336, respectively, for their share of the costs in liquidating the remainder.

We sketch a final, possibly fairer method inspired by the Vickrey auction, but we reserve a full treatment and analysis for later work. In this model, an institution's share of the commission would be based on its impact on the market versus the marginal economy without its basket. Thus, institutions who *improved* the market by submitting a basket with opposite interest from the remaining baskets would pay very little (or perhaps even be paid!). Institutions who made the market more unbalanced by submitting a basket with interest in the same direction the remaining baskets would pay a greater share of the commission, because its trades would only be filled by means of the liquidity providers.

[11] Since the numbers do not divide evenly, the exchange can simply round up to the nearest integer and prove that the result is within a small error, that is, the difference between the total commission and the reported commission is small.

[12] If verifiable operations over encrypted rationals are employed, even this rounding error can be (practically) eliminated at a constant factor of additional computation cost.