Towards Multi-Attribute Double Auctions for Financial Markets

HENNER GIMPEL AND JUHO MÄKIÖ

INTRODUCTION

Financial markets usually are double sided markets: They bring together the supply and demand of multiple sellers and buyers in a structured way and determine the price and allocation of financial products. Standardized products, like stocks, are frequently traded at exchanges and via electronic trading platforms. Most of these trading venues are organized as so called double auctions, i.e. auctions with competition on both sides of the market. The continuous-time double auction is a popular market mechanism for continuous trading – buyers and sellers can submit orders any time during the auction and they are matched instantaneously, if possible. Call markets are the other class of double auctions; bids and offers are gathered over time and cleared all at once when the auction closes. Call markets are oftentimes used to open or close a continuous market or when the market experienced a volatility shock.

Standardized products are traded on double auction-based markets in many cases. Financial derivatives like options and futures are, however, often traded off-exchange over the counter (OTC). One predominant reason is the lack of standardized commodities due to the inherent complexity of trading these products. In negotiating a deal, e.g. for an option, attributes like the type of option, underlying instrument, style, contract size, maturity and strike price have to be defined. Established double auction mechanisms are not capable of handling these multiple attributes. Hence, derivatives are not traded on centralized markets but, to date, trade bilaterally.

The present paper re-engineers widely established continuous-time double auctions and extends them to handling multi-attribute financial products. Additional attributes could include time to expiration for options and futures contracts, and credit rating or time to maturity for bonds. Implementing the presented multi-attribute double auctions offers stock exchanges the potential to attract trading volume from OTC markets. The paper outlines the employed methodology of financial market engineering, the class of multi-attribute double auction mechanisms and its design alternatives and, finally, the implementation in the meet2trade system (Weinhardt et al. 2005; http://www.meet2trade.com).

Financial market engineering

In recent years, economics has experienced a tendency to move

Abstract

Technological progress has led to electronic exchanges attracting a major part of transaction volume for standardized stocks. In the field of derivatives and bonds, electronic exchanges failed to prosper and so called over-the-counter (OTC) trading is predominant: possibly because multi-attribute assets are hard to trade in structured two-sided markets. The present paper aims at alleviating this shortcoming.

The present work follows the market engineering approach and presents a class of double auction mechanisms that is able to handle multiple attributes – the inherent potential of these mechanisms can be utilized in additional products to reduce search and coordination costs by equating supply and demand centrally. The mechanisms are individually rational, Pareto optimal, coalition proof and budget-balanced. As a proof of concept and to illustrate feasibility, the mechanisms are implemented in the meet2trade system.

Keywords: option trading, bond trading, financial exchange, electronic trading system, market engineering

Authors

Henner Gimpel (henner.gimpel@iwr.uni-karlsruhe.de) is a PhD student at the Institute of Information Systems and Management, University of Karlsruhe, Germany. His research focuses on market engineering. His specific interest is in experimental economics and behavioural aspects in negotiations.

Juho Mäkiö (juho.maekioe@iwr.uni-karlsruhe.de) is a PhD student at the Institute of Information Systems and Management, University of Karlsruhe, Germany. His research focuses on market engineering. Currently he is working on a generic market modelling language and the meet2trade system.
Trading financial derivatives

Double auctions serve as basic building block for the microstructure of the market to be designed for financial derivatives. Orders in a conventional double auction are price–quantity pairs submitted to the electronic auction system as bids (buy orders) and asks (sell orders). Upon arrival of an order, the system checks whether it can immediately be matched against orders waiting on the other side of the market; if so, it is executed instantaneously. If the order cannot be executed, it is put in an order book and waits to be matched against subsequent orders.

With continuous time, a double auction mechanically reduces to the matching of an incoming order against the order book on the other side of the market. An order is not a single point in the attribute-space; rather it is a subspace of acceptable attribute combinations. A trade is executed, if the space implied as acceptable by the incoming order overlaps with at least one order on the other market side. Orders on different sides of the order book can never overlap without a newly incoming or updated order; if they would overlap, they would already have been executed upon arrival.

Two major questions have to be answered by the auction mechanism: (1) if an incoming order overlaps with more than one opposing order, which of them is chosen for execution (matching) and (2) once a pair of two orders is matched unambiguously, there might still be several possible attribute combinations acceptable for both parties involved, but one single point in the attribute space has to be chosen (arbitration). Game theory helps in resolving both cases. Matching is done by augmenting orders with preference structures and optimizing individual or social objectives like utility or welfare. Arbitration uses concepts from cooperative game theory, like e.g. the Nash bargaining solution (Nash 1950).

The remainder of the paper is structured as follows: the next section outlines the concept of a multi-attribute double auction mechanism. Subsequently, the implementation in the meet2trade system is described and related work is surveyed. The final section concludes. The contents partially base on earlier work by Gimpel (2005) and Gimpel et al. (2005).

AUCTION MECHANISM

A multi-attribute double auction mechanism is presented in this section. First, four properties of a reasonable double auction mechanism are postulated. Then single-attribute double auctions are discussed to introduce the concepts of matching and arbitration before they are extended to the multi-attribute case. Finally, a discussion on discrete attributes follows.

Requirements

This section outlines four properties that a multi-attribute double auction should possess from an axiomatic viewpoint:
Budget-balance requires, in its weak form, that the market operator does not have to subsidize trading and the exchange does not run at a loss. This requirement is vital for a market operator’s business model.

Pareto optimality measures efficiency – a trade is Pareto optimal if there is no other trade that makes every trader at least as well off and at least one trader strictly better off.

Individual rationality requires that each individual trader prefers participating in the mechanism to not participating. If the auction would, for example, force a trader into a trade which is not desirable, the trader would be better off refraining from participation and trading via an outside venue. Therefore, each auction has to allow for traders to submit an indication on their preferences and must never impose an undesirable trade. In a conventional price double auction individual rationality requires that a buyer never buys for a price higher than the specified limit price and a seller never sells below the limit price.

Coalition proofness requires that there is no set of two or more traders which could form a coalition and trade via an outside venue with each member of the coalition being better off than participating in the auction. If such coalitions would exist, traders might refrain from participation and search for trading partners by means of other venues.

Generally coalition proofness is a desirable property. However, in a scenario with a closed group of participants who cannot leave the market, the requirement can be relaxed to achieve potentially desirable (social) objectives like welfare maximization. A government, for example, could in some domains enforce the usage of a single trading system and, therefore, refrain from requiring coalition proofness.

The overall objectives of these requirements are to encourage usage and order entry by market participants, to increase liquidity and to make the operation of the market profitable.

### Single-attribute double auctions

In a single-attribute continuous-time double auction with limit orders, there is a bid–ask spread and never more than one order overlapping with orders on the other side of the market. Figure 1 displays a potential order book with three bids (B1 to B3) and three asks (A1 to A3). Each order is characterized by its market side (bid or ask) and a limit price. The order volume is irrelevant in this context and therefore omitted. The trader who submitted A3, for example, is willing to sell at a price of €100 or above; for A2 it is €103 or above and so on.

Participants’ preferences over the price are monotonic: a seller would always accept a price higher than the one he posted in his order and a buyer would always prefer a lower price. The system implies this and actually treats an order price $p$ as an interval of acceptable prices ranging from zero to $p$ (buy) or from $p$ to infinity (sell). This idea can be extended to the multi-attribute case – an order is not a single point in the attribute-space, but a subspace. The direction of monotonicity of preferences on each attribute has either to be known implicitly or to be explicitly specified. At each time, at most one order – the newly entered or updated one – overlaps with the other market side. In Figure 1, this is bid $B_3$.

Matching is the process of selecting a single order which overlaps with the newly entered order. Order $B_3$ can be matched with either $A_2$ or $A_3$. A common matching rule in financial markets is to take the lowest ask or highest bid limit, respectively. If two or more orders specify the same limit price, a tie-breaking rule like price-time-preference has to be applied. In the example, $B_3$ is matched with $A_3$, i.e. the ask with the lowest limit price.

Arbitration is the process of determining the exact specification of the trade among two matched orders. Oftentimes, the limits of a matched pair do not coincide and there is more than one possible transaction price. The interval $P$ gives the potential prices. For orders $A_3$ and $B_3$, any price $p ∈ P=[100, 110]$ is individual rational as well as Pareto optimal and budget-balanced and can therefore be chosen. Common arbitration rules are to take either the limit of the order in the order book ($p=100$), the incoming order’s limit ($p=110$) or the midpoint of the arbitration interval ($p=105$). However, prices $p>103$ are not coalition proof; the traders who submitted $B_3$ and $A_2$ would both be better off with forming a coalition outside the mechanism and trading for a price of, for example, €103. Therefore, the requirements outlined before call for any price $p ∈ [100, 103]$ – which exact price is fair, just or best cannot be answered unambiguously but is up to the market operator’s decision. The presented concepts are now generalized to the multi-attribute case.

### Matching with multiple attributes

As in the single-attribute case, orders in the order-book can never be executed against one another. Therefore,
matching is, again, the process of selecting an order that overlaps with the newly entered order. In the multi-attribute case it is not always unambiguous to assign roles of buyers and sellers to the traders and to name orders as bids and asks. However, in the following the terms are used to simplify matters; the auction mechanism does not depend on this naming.

In selecting a matching order, the first step is to check which orders overlap with the newly entered one. Again, an order is not regarded as a single point (a single value on each attribute), but as a subspace of the space of possible agreements (an interval on each attribute). For doing so, attributes have to be defined on at least an ordinal scale and preferences have to be monotonic in each single attribute. The direction of monotonicity has to be common knowledge or given in the order. Making a monotonicity assumption is questionable, as traders might have non-monotonic preferences. A trader could, for example, prefer bonds maturing in 5 years over bonds maturing in either 3 or 7 years. In case of non-monotonic preferences, the order has to be split into several orders each of which is defined over a range where the preferences are monotonic.

Figure 2 exemplifies multi-attribute orders for a two-attribute example in an Edgeworth box. Obviously, other attributes like the type of option, the underlying instrument and so on would have to be specified in a real-world scenario. There are three bids \((B_1, B_2, \text{ and } B_3)\) in the order book when the ask \((A)\) enters the market. Each order in this representation is a rectangle, i.e. a two-dimensional subspace of the two-dimensional agreement space. Orders additionally contain utility values as discussed later. The trader who submitted order \(A\) is willing to sell at any price not lower than €100 and any option with a maturity between 30 and 105 days. The more on either value, the better for the seller, but anything within the rectangle to the upper right is acceptable. For bids \(B_1, B_2, \\text{ and } B_3\), it is exactly the other way round. These orders are specified by a single point and implicitly signal the acceptance of any trade at the lower left of this point – a lower price or shorter maturity is acceptable.

Obviously, traders have preferences over the different possible trades. In Figure 2, the three indifference curves indicate the sellers’ preferences; the higher the curve, the better for the seller. The assigned utility values are \(u_1, u_2, \text{ and } u_3\) and \(u_3 > u_2 > u_1\) holds. Note that well-behaved preferences are assumed at this point (Varian 1992).

As a practical matter it could be difficult for traders to determine these indifference points and they could change second by second as market conditions change. However with sophisticated software and analytical tools, it is assumed to be possible. The shape of the order, i.e. the range of acceptable trades, is restricted to rectangles here.

With a more complex bidding language it would be possible to shape orders in a way that more closely resembles the indifference curves. The lower left border of order \(A\) could, for example, be fitted to the lowest indifference curve displayed. However, for simplicity this is neither detailed here nor implemented in the prototype.

In the single-attribute case the incoming order was matched with the one potentially offering the highest utility. More specifically, the selling order with the lowest price was selected. Analogously, the bid which potentially offers the highest utility to the incoming order is chosen. In Figure 2, ask \(A\) is matched with \(B_2\) potentially offering a utility of \(u_3\).

Formally, matching is selecting a pair \((A, B^*)\) with \(B^* = \arg \max_B \{B_1, \ldots, B_n\} \max_x \in A \cap B \ u_B(x)\) where \(u_B\) is the multi-attribute utility function associated with order \(A\) and \(x\) is a point in the agreement space. Furthermore, \(A\) is the incoming order and \(B_1, \ldots, B_n\) are the orders in the book. At this point in the process, just the utility function of order \(A\), i.e., the single order that has bargaining power, is used. The utility function of \(B^*\) will subsequently be used in the arbitration phase and the utility functions of all other orders in the order book (except \(B^*\)) are irrelevant for this specific trade.

The system retrieves utility functions from each trader. Hence, the question arises whether this information is sufficient or whether it really has to be accompanied by an order giving a subspace in which the utility function is

![Figure 2. Matching in a two-attribute order-book](image-url)
applicable. Generally a utility function might be sufficient. However, defining the subspace of its applicability is convenient for two purposes: first of all the functional form and the values of the utility function might vary across the agreement space; limiting its applicability cases specifying it for a relatively small subspace. Furthermore, the order restriction serves to denote the agreements a trader is absolutely not willing to make, e.g. because he does not want to sell a product short. If, however, a trader wants to define his utility function over the entire agreement space, this is possible in principle.

Arbitration with multiple attributes

As in the single-attribute case, matching selected a pair of orders \((A, B^*)\) and the arbitration rule has to determine the exact specification for the trade. In the example from Figure 2, price and maturity have to be determined.

Trading any option with a maturity between 30 and 90 days at a price between €100 and €120 is individual rational and budget-balanced. However, most of these trades would not be coalition proof. With a maturity of 30 days and a price of €100, for example, the trader who submitted \(A\) would be better off with reaching an agreement with one of the two traders who submitted \(B_1\) and \(B_2\) outside the mechanism. Therefore, the final trade has to give \(A\) higher utility than a trade with another order could possibly do. All individual rational, budget-balanced and coalition proof possible trades are in the shaded area \(P\). Note that any trade in \(P\) is better for \(B_2\) than not trading with \(A\) at all. This is because \(A\) is the only sell order overlapping with \(B_2\) at this time. Otherwise, a trade would already have been executed.

For simplicity, an order volume of one is assumed here and ask \(A\) is just matched with a single order on the other market side. If \(A\) would ask for two or more units buy quantities could be aggregated. In this case, \(A\) would lose some bargaining power. It would, for example, additionally be matched with \(B_3\) and, thus, \(B_1\) would be the only remaining order that restricts coalition proofness. The area \(P\) for arbitrating among \(A\) and \(B_2\) would increase.

The two traders who submitted orders \(B_2\) and \(A\) have (partially) opposing preferences on the exact specification of the deal. The seller would prefer a higher price, whereas a buyer would prefer a lower price. In the arbitration phase the auction mechanism has to resolve this conflict. The area \(P\) can be mapped to the utility space in Figure 3. Obviously, the mechanism should restrict its choice to Pareto optimal solutions, i.e. the upper right boundary of the possible utility space. Which point to choose on the Pareto frontier is, as in the single-attribute case, not unambiguous.

![Arbitration in the utility space](image)

The auction takes the role of an arbitrator in a bilateral multi-attribute bargaining situation. Cooperative game theory provides different arbitration rules, each of which are supposed to find a "fair" solution. Note that these arbitration concepts require cardinal utility and interpersonal utility comparisons, while up to now ordinal utility was sufficient; see Kalai (1977) for a discussion of this matter.

The most prominent concept is the Nash bargaining solution (Nash 1950). The Nash solution – not to be confused with the Nash equilibrium in non-cooperative game theory – bases on the axioms symmetry, Pareto optimality, invariance to equivalent utility representation and independence of irrelevant alternatives. It turns out that these axioms uniquely determine a solution which Nash characterizes as fair. For identifying the Nash solution in Figure 3, one needs the concept of a conflict point, i.e. the outcome for both traders if they would not reach an agreement. The conflict point can be interpreted as no trade at all and can be assumed to yield zero utility for both traders. The Nash solution is the trade that maximizes the product of both traders’ utilities; it is displayed in Figure 3. Kalai and Smorodinsky (1975) replaced the independence of irrelevant alternatives axiom by individual monotonicity. While the Nash solution bases on the conflict point, the Kalai–Smorodinsky solution additionally requires an ideal point called utopia and defined as the point that would give both traders the highest utility they could expect from any solution. The Kalai–Smorodinsky solution is the point at which the straight line from the conflict outcome to the utopia intersects the Pareto frontier. It gives each trader the same share of the best outcome he could possibly expect; it is sketched...
in Figure 3. The Kalai–Smorodinsky solution is a special case of the Gupta and Livne (1988) solution. Yet another reasonable solution displayed in Figure 3 is to maximize the sum of utilities (cf. Harsanyi 1955).

All these solution concepts for the bilateral bargaining problem can be utilized as arbitration rule in a multi-attribute double auction. Furthermore, infinitely many other arbitration rules are possible. There is no definite best or fairest way to choose among arbitration rules – any rule that satisfies individual rationality, Pareto optimality, weak budget-balance and coalition proofness can be applied.

Formally, given a pair \((A, B^*)\) of matched orders, arbitration is selecting a trade specification \(x^*\) with the following properties:

- \(x^* \in A \cap B^*\) guarantees budget-balance (the exchange does not run at a loss) and individual rationality (traders prefer participating in the mechanism to not participating);
- \(u_A(x^*) = u_A(B^*)\) with \(B^*\) being the second best order for \(A\), i.e. \(B^* = \arg \max_B \{B_1, \ldots, B_n\} \max_x x \in A \cap B\) \(u_A(x)\), assures coalition-proofness (no coalition of at least two traders is better of by leaving the exchange and trading via an outside venue); and
- \(x^*\) is Pareto optimal (no other trade is strictly preferred by one party without being worse for the other party). This last requirement can be achieved by applying any of the bargaining solutions outlined above.

Note, that these properties do not preclude trade-toughs. If the order can be executed with more favourable terms at another exchange it can be routed there without being executed first. Comparing possible executions in different market systems is, however, far more complex with multiple attributes than in conventional price-only markets. Further note, that budget-balance is independent of traders’ utilities: the trade specification in the intersection of the traders’ orders \((x^* \in A \cap B^*)\) guarantees that one trader is willing to give away exactly what the other trader will receive – not in terms of utility, but in terms of the product specification.

How does the specific choice of an arbitration rule influence incentive compatibility? Achieving incentive compatibility, i.e. giving the participants an incentive to reveal truthfully their preferences, is impossible with any mechanism meeting the requirements outlined above (Myerson and Satterthwaite 1983). However, the degree of approximation might very well be different for different arbitration rules. The analysis of this interrelation is subject to future research. Note that Pareto optimality cannot be guaranteed if traders do not reveal their preferences truthfully. Hence, the discussion of Pareto optimality is restricted to stated preferences here.

**Example with discrete attributes**

Imagine a market for put options, where two discrete attributes can be specified: the underlying instrument (e.g. stocks from SAP or IBM) and the strike price (e.g. €100, €110, €120). Note that the underlying instrument is taken as an attribute here and not as a different product, as it is assumed that an option on the SAP stock partially substitutes for an option on the IBM stock and vice versa. Trader \(B\) wants to buy a put option and traders \(S\) and \(T\) are willing to sell put options. The traders’ preferences over the different specifications are given as utility values in Table 1 where a higher number indicates a higher utility from trading such an option and therefore a preferred combination.

The two sellers \(S\) and \(T\) might already have placed orders in the market and trader \(B\) then submits his order. In this scenario with discrete attributes, the region of acceptable agreements is given by placing a positive number on a combination of attribute values; i.e. the only acceptable trade for seller \(T\) is to sell a put option on the IBM stock with a strike price of €100. For seller \(S\), three combinations are acceptable and for buyer \(B\), all six combinations are acceptable, although a higher strike price is better and IBM is preferred to SAP.

Upon arrival of \(B\)’s order, the trading system checks whether it overlaps with the orders on the other market side. In this example it does: \(B\) could trade with either \(S\) or \(T\). Therefore, \(B\) has some bargaining power and the mechanism determines in the matching phase which of the sellers offers the highest utility to \(B\). Trading with \(S\) is best for \(B\), as \(B\) can get a utility of 0.64. Note that trading with \(T\) would maximize the sum of utilities. This is, however, not the aim here as it is not coalition proof.

The aim is to match the single order that has bargaining power with an order that maximizes utility of the incoming order.

Now the mechanism has to select a specific combination of attribute values to specify the put option that \(S\) sells to \(B\). The requirements outlined above restrict the set of possible trades:

<table>
<thead>
<tr>
<th>Trader</th>
<th>€100</th>
<th>€110</th>
<th>€120</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>0.16</td>
<td>0.48</td>
<td>0.80</td>
</tr>
<tr>
<td>(buy)</td>
<td>IBM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>0.32</td>
<td>0.64</td>
<td>0.96</td>
</tr>
<tr>
<td>(sell)</td>
<td>IBM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>0.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(sell)</td>
<td>SAP</td>
<td>0.16</td>
<td></td>
</tr>
<tr>
<td>(sell)</td>
<td>IBM</td>
<td>1.00</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Traders’ utility in a discrete example
Individual rationality rules out a strike price of €120 and an option on SAP for €110, as these combinations are not acceptable for S.

Pareto optimality and budget-balance are satisfied by all three remaining specifications (IBM for €100, IBM for €110 and SAP for €100).

Coalition proofness rules out an option on SAP for €100. This would be highly desirable for S, but B and T could form a coalition and agree to trade an option on IBM for €100. This would make both, B and T, better off.

The remaining two specifications – IBM for €100 and IBM for €110 – form the set P of possible trades. In the arbitration phase, the mechanism has to select a single specification for the trade from P. This can, for example, be done by selecting the Nash solution, i.e. the point that maximizes the product of utility values. This arbitration rule here results in trading a put option on IBM for a strike price of €100.

The result crucially depends on all three orders in the example. Assume T's order would be a little more aggressive – a put option on SAP for a strike price of €110 might be acceptable with a utility value of 0.01. Although T is not matched with B, the more aggressive order influences the trade. Now coalition proofness additionally rules out an option on IBM for €100 as B would then prefer trading with T. The final trade for S and B is a put option on IBM for a strike price of €110. Thus, the more aggressive order by T gives B more bargaining power and, consequently, B benefits. Would, on the other hand, S in the original example be less aggressive and not offer any option on the IBM stock, then B and T would be matched and trade an option on IBM for a strike price of €100.

IMPLEMENTATION

The outlined mechanism is implemented in the meet2trade system. Ranging from bilateral negotiations to auction mechanisms and more complex hybrid market protocols, the platform enables the automation of trading and negotiation processes.

As aforementioned, two questions have to be defined by the auction mechanism: namely matching and arbitration. From the technical point of view, these questions lead to further questions concerning implementation. The trading rules were defined according to economic and business requirements so far. The discussion now turns to a software engineering perspective on the IT infrastructure.

The meet2trade platform is a client-server-based generic trading system, implemented in Java (Weinhardt et al. 2005). On the client side orders are specified in an XML format and submitted via JMS (Java Messaging Service) to the server where the order specification is transformed into a Java object. Each order instance contains a preference object that represents all relevant individual preferences of the submitting agent and the current order respectively.

The preference object is generated and initialized from the XML-based order description during order transformation. It contains a table of computed preference values for each order. Each order object contains one preference object that in turn contains a table of preference values. Each value can be accessed by a preference key. Both preference table entries – the preference value and the preference key – are computed from the entries in the XML order specification when the order is inserted into the market. The preference key basically is the set of attribute values for a specific contract. Currently, the system supports additive utility functions. Additive utility functions are the most widely used heuristic for trading-off multiple objectives; see Keeny and Raiffa (1993) for details.

As outlined above, several different design options are available for a multi-attribute double auction. The following algorithm in pseudo-code constitutes one specific version; it computes welfare maximization, i.e. a closed market is assumed and the coalition proofness requirement is relaxed in this example. Note that welfare maximization requires knowledge of traders’ preferences – the preferences revealed in the orders are taken as substitutes for real preferences here, although incentive compatibility cannot be guaranteed (Myerson and Satterthwaite 1983). For the sake of simplicity, order volume and order disclosure are left out of this analysis. Furthermore, it is assumed that orders cannot be changed or withdrawn during runtime of the algorithm.

```
algorithm: matching for welfare maximisation
input (current order co)
output ([List of tradable pairs lt])
begin
    current max preference value = 0;
    lt = new list();
    for each preference key pk of co do
        ov = preference value of co for pk;
        for each order of the opposite side os do
            if ((ov + v) > current max preference value)
                insert the pair (pk, os) into lt;
            endif
        endfor
    endfor
    return lt;
end
```

The following example clarifies the algorithm above: an agent wants to buy a put option and submits an order X.
The order book contains three potential counter parts, namely the orders A, B and C, as outlined in Table 2. The negotiable attributes are the underlying instrument \((U)\), the strike price \((S)\) and the maturity \((M)\). For \(X\), the preferred underlying instrument is the IBM stock (0.51) but the SAP stock is almost as good (0.49). In this case the underlying is an attribute, IBM and SAP are characteristics and 0.51 and 0.49 are their weights. Attributes are traded-off against each other by another set of weights. The strike price could, for example, have a weight of 0.3 and the underlying of 0.5 and the maturity of 0.2. The attribute weights are omitted for the rest of the example, as they are irrelevant here.

Consider the potential trading pair \((X, A)\). If \(X\) is matched with \(A\) \((X\rightarrow A)\) then the attribute \(U\) would be set to the characteristic IBM for welfare maximization (cf. Table 2):

\[
\max((X_{SA} + A_{SA}), (X_{BA} + A_{BA})) = \\
\max((0.49+0.45), (0.51+0.55)) = 1.06.
\]

For each potential trading pair the sum of the maximums of the above sums is calculated over all attributes. For example: \(X\rightarrow A\Rightarrow \max((0.49+0.45), (0.51+0.55))\)+\(\max((0.20+0.38),(0.80+0.62))\)+\(\max((0.52+0.58), (0.48+0.42))\)

\[=1.06+1.42+1.1=3.58.\]

The algorithm searches the maximum of the sums in order to determine the trading pair. This leads to the following values and the matching terminates with trading pair \((X, C)\), as \(X\rightarrow A=3.58, X\rightarrow B=3.32\) and \(X\rightarrow C=3.95\). A put option with a maturity of 60 days, a strike price of €110 and the SAP stock as underlying is traded.

The arbitration rule determines the exact specification for the trade. The above algorithm fulfils this requirement when the solution is unambiguous. However, in conflict situations there exists more than one solution and an additional arbitration rule is necessary. If there is more than one potential (welfare maximizing) trading partner for the current order (i.e. the size of the \(\tau t > 1\)), then the order that entered the market earliest (welfare-time-priority) is taken for execution.

In multi-attribute double auctions, welfare maximization can be criticized as it does not accomplish coalition proofness. Furthermore, welfare maximization – defined as cumulated rents of all traders – requires knowledge on traders’ preferences which has to be approximated by taking the preferences revealed in the orders. Nevertheless, welfare maximization was outlined in this section, as it is a popular concept in single-attribute double auctions. The met2trade system implements a plethora of other matching and arbitration rules as well and the choice of the specific mechanism is up to the market engineer.

RELATED WORK

On the application side, the work of Weinhardt and Gomber (1998) relates to the presented approach. The authors propose an off-exchange system for bond trading based on software agents. A major difference is that there is no centralized auctioneer in their system but each order is represented by a software agent searching for others to trade with. The present work takes multi-attribute trade-offs across product attributes, and the search for bilateral Pareto improvements into account.

On the system side, the OptiMark system is related to the presented approach. OptiMark is an electronic financial trading system designed to allow traders to specify their preferences graphically across a range of sizes and prices and to facilitate cost-effective block trading (Clemons and Weber 1998, 2001). The congruence with the approach presented here is that orders are not point orders but describe a range of acceptable trades. Preferences are in both systems specified via a graphical user interface and the system ‘optimizes’ the matching of these trading preferences.

On the auction side, there is a substantial literature on one-sided multi-attribute auctions, see for example, Beil and Wein (2003), Bichler and Kalagnanam (2005); Parkes and Kalagnanam (2005) and Strecker and Seifert (2004). Key objects of investigation in this research branch are preference elicitation, the performance of different auction institutions, bidding languages, and preference revelation. All four objects of research on single-sided multi-attribute auctions relate to multi-attribute double auctions as well. Preference elicitation

<table>
<thead>
<tr>
<th>order</th>
<th>underlying (U)</th>
<th>strike price (S)</th>
<th>maturity (M)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SAP</td>
<td>IBM</td>
<td>€100</td>
</tr>
<tr>
<td>A</td>
<td>0.45</td>
<td>0.55</td>
<td>0.38</td>
</tr>
<tr>
<td>B</td>
<td>0.57</td>
<td>0.43</td>
<td>0.67</td>
</tr>
<tr>
<td>C</td>
<td>0.89</td>
<td>0.11</td>
<td>0.70</td>
</tr>
<tr>
<td>X</td>
<td>0.49</td>
<td>0.51</td>
<td>0.20</td>
</tr>
</tbody>
</table>
techniques, i.e. techniques helping a trader to externalize his preferences and trade-offs across multiple attributes, are a client-side support which can directly be utilized within the presented double auctions; an example concerning financial derivatives is given by Bichler (2000).

To date, there are neither comparisons on the performance of different multi-attribute double auctions, nor studies on bidding languages or the effect of preference revelation in this context. All of these are subjects of future research.

CONCLUSION

Financial derivatives and bonds are mainly traded over the counter to date. One reason can be seen in the fact that conventional double auction mechanisms cannot handle goods that are specified along several attributes. The bilateral search and negotiation in OTC markets has a lower transparency than a centralized market. The present paper presents a class of auction mechanisms – namely continuous-time multi-attribute double auctions – that are capable of centrally equating supply and demand for multi-attribute goods.

Electronic markets have to be engineered along three major building blocks: microstructure, IT infrastructure and business structure (Weinhardt et al. 2003). The latter one is mostly omitted in the present paper, except contributing the requirement that a market operator does not have to subsidize trading, i.e. the auction has to be budget-balanced.

The microstructure is derived by defining matching and arbitration for multi-attribute double auctions. Matching thereby is selecting an order from the order book to be executed against a newly incoming order. Orders are specified as subspaces of the multi-attribute agreement space. Arbitration is defining the specific conditions of the transaction for the matched orders. Concepts from cooperative game theory are employed here, most prominently the Nash bargaining solution and the Kalai–Smorodinsky bargaining solution. They select a single point in the agreement space which becomes the contract among traders. For this, orders are augmented by utility functions.

Objectives for the economic design of the mechanisms are Pareto optimality, individual rationality, coalition proofness and budget-balance. All of these serve to encourage market entry and therefore increase liquidity. The class of mechanisms presented satisfies these requirements.

The feasibility of implementing the IT infrastructure and operating a trading system based on the presented auction mechanisms is proven by a prototypical implementation in the meet2trade platform. The auction mechanisms bear the potential to facilitate electronic commerce in domains where most of the trades are negotiated manually up to date. By utilizing them, market operators can offer their customers, compared to bilateral trading, a reduction in costs for searching trading partners and negotiating multi-attribute deals. This can be applied but is not restricted to markets for financial derivatives and bonds. Other application areas comprise corporate and public procurement.

Market engineering suggests a process model for introducing new market mechanisms and platforms. Future work will involve proceeding along this model and especially employing computer simulations and laboratory experiments to test the strategic implications of the presented mechanisms (Bloomfield 1996, Schwartz and Weber 1997). From mechanism design theory, it is known that mechanisms satisfying the above requirements may not be incentive compatible, i.e. traders will not reveal their true preferences in equilibrium. However, the psychology involved in trading multi-attribute goods is to be studied in the lab and reasonable heuristics for (boundedly) rational agents are to be identified in simulations.

ACKNOWLEDGEMENTS

The authors wish to thank two anonymous reviewers for their helpful comments. This research was partially developed within the cooperation project electronic Financial Trading (e-FIT) as part of the research programme Innovative Financial Services funded by the German Federal Ministry of Education and Research (BMBF). The e-FIT project was co-financed by the BMBF, Boerse-Stuttgart AG, Reuters AG, trading fair AG and the Institute of Information Systems and Management, University of Karlsruhe.

References


