Direct Exchange Mechanisms for Option Pricing

Sarvar Abdullaev(✉), Peter McBurney, and Katarzyna Musial

Department of Informatics, King's College London, London, UK
{sarvar.abdullaev,peter.mcburney,katarzyna.musial}@kcl.ac.uk

Abstract. This paper presents the design and simulation of direct exchange mechanisms for pricing European options. It extends McAfee's single-unit double auction to multi-unit format, and then applies it for pricing options through aggregating agent predictions of future asset prices. We will also propose the design of a combinatorial exchange for the simulation of agents using option trading strategies. We present several option trading strategies that are commonly used in real option markets to minimise the risk of future loss, and assume that agents can submit them as a combinatorial bid to the market maker. We provide simulation results for proposed mechanisms, and compare them with existing Black-Scholes model mostly used for option pricing. The simulation also tests the effect of supply and demand changes on option prices. It also takes into account agents with different implied volatility. We also observe how option prices are affected by the agents' choices of option trading strategies.

Keywords: Mechanism design · Option pricing · Double auctions · Combinatorial exchanges · Prediction markets

1 Introduction

Standard financial theory provides a number of methods for calculating option prices based on the market performance of an underlying asset. But there are few models that take into account strategic agents playing in this market, and their role in forming the prices. It is commonly assumed that an individual trader is mostly a price-taker and therefore her influence to the market is insignificant. But in reality, traders with their aggregated utilities form the market prices. Although it is almost impossible to know how each individual agent would evaluate the risk in the market, we can still model them with reasonable properties such as rationality, strategic behaviour and risk-neutrality. This would provide a testable environment where various market mechanisms and trading behaviours can be simulated and used for taking analytical decisions.

There has been a growing interest in the research of markets as complex game-theoretic systems since Myerson coined *mechanism design* as a framework for strategic interactions between self-interested agents [1]. A new discipline of *auction theory* emerged as a part of mechanism design, and it found

N. Bulling (Ed.): EUMAS 2014, LNAI 8953, pp. 269–284, 2015.
DOI: 10.1007/978-3-319-17130-2_18

its applications in solving many of well-known problems such as resource allocation, scheduling, supply chain optimization, operations control and multi-agent system implementation [5]. The ultimate goal of any auction is the allocation of scarce resources to agents. The space of auction types is limitless, because they may vary in their initial settings, bidding rules, market clearing methods etc. Parsons describes more than 30 variations of auctions based on properties such as dimensionality, quantity and heterogeneity of traded items; direction, sidedness, openness of accepted bids; and kth order prices in determining winners [4].

There have been a number of researches accomplished in applying auction mechanisms to model prediction markets. One of the famous examples of such mechanisms is *Iowa Electronic Markets* used for aggregating predictions on political elections [12]. DeMarzo *et al.* have used regret minimisation of agent decisions on compiling a replicating portfolio which is equivalent to European option value [11]. King *et al.* has described a multi-agent model for derivatives market which used Gaia methodology [9] to match and coordinate agents. Espinosa has implemented a multi-agent system which uses options to allocate scarce resources through a market-like model [6].

We will focus more on Double Auctions (DA) and Combinatorial Exchanges (CE) in this paper. DA is an auction mechanism which involves sellers and buyers trading identical goods using single-item bids. McAfee laid the foundation of DA specifying the direct implementation of a DSIC mechanism which could match bids and asks efficiently [13]. CE is the generalisation of DA where traders are allowed to submit bids and asks as a bundle for heterogeneous goods. We will use DA and CE as prediction markets to evaluate option prices.

The paper is organised as follows. Section 2 provides the basic framework within which we will construct our mechanisms. We will define fundamental concepts used in auction theory and review the main aspects of option pricing. In Sect. 3 we will talk about how traders are going to produce bids and asks, and select option trading strategies (OTS). Then we will walk through the design of multi-unit DA and consequently the CE mechanisms. Section 4 provides experimental results obtained from the simulation of both proposed mechanisms. Finally, in Sect. 5 we conclude highlighting the important aspects of our work.

2 Preliminaries

In this section, we explain some of the key concepts that we use throughout this paper. This involves the basic framework for mechanism design and some brief overview of options and their intrinsic values.

2.1 Designing Mechanisms

The very idea of designing mechanisms imply making rules for given game settings that incentivise truthful revelation of agent types. In terms of auctions, it can be seen as the truthful bidding of agents. Myerson proved that if the allocation rule of the auction is monotone, then there is a unique and explicit payment rule which makes the mechanism *dominant strategy incentive compatible* (DSIC) [3]. This payment rule should include the *critical values* of each agent who has been allocated with goods. Critical value of an agent is the value that the agent needs to beat in order to get the good. For example, in terms of single-item auction, the payment rule corresponds to the second price, because agents must beat the second price to be the winner of the auction. In a continuous domain, Myerson's payment rule can be defined as follows:

Definition 1. For an auction with a monotone allocation rule $\chi(\mathbf{b})$, the *Myerson's payment rule* is

$$\rho_i(b_i, \mathbf{b_{-i}}) = \int_0^{b_i} z\chi_i'(z, \mathbf{b_{-i}})\mathrm{d}z \tag{1}$$

where b_i denote the agent i's bid, $\mathbf{b_{-i}}$ the bids of the rest of the agents, and χ_i' is the marginal allocation rule for bidder i.

Hence it is clear that we can calculate DSIC payments for agents given that we have a monotone allocation rule which never decreases as the bidder increases her bid. One economically fair way of allocating goods is giving it to the highest bidder, or in other words, maximise the social surplus. Indeed, *surplus maximisation* (SM) rule is monotone, because whenever bidder increases her bid, if it beats the other bids, the surplus maximising algorithm will select this bid and thus will increase the number of allocated items to this bidder. In case if it does not exceed the other bids, the bidder's allocation will remain unchanged. For this reason, we will use SM as our main allocation rule in our simulation model.

In double auction and exchange environments, the SM involves the maximisation of the utilities of buyers and sellers. We can define the utility for the agent as follows:

Definition 2. For given agent i, her *ex-post quasilinear utility* is

$$u_i(\mathbf{q}_i) = v_i(\mathbf{q}_i) - \mathbf{p}^\mathsf{T}\mathbf{q}_i \tag{2}$$

where v_i is the valuation function, $\mathbf{q}_i$ is the allocation result of a bidder i, and $\mathbf{p}$ represents the anonymous prices.

Thus utility function requires two types of outcomes from given mechanism: $\mathbf{q}_i$ the quantities allocated to agent i and $\mathbf{p}$ anonymous clearing prices. The agent i will buy (sell) the item j if $q_{ij} \in \mathbf{q}_i$ is positive (negative). So the quantities for a pure seller will be all negative, and for a pure buyer positive. We will assume that the valuation function $v_i(\mathbf{q}_i)$ will also reflect this relationship. Quasilinear utility assumption also implies that agents are risk-neutral as it changes linearly with no budget constraints. However, risk-neutrality in the context of option pricing must not be confused. We will later assume that every agent will have her own forecast on future price of an underlying asset (which might not be a risk-neutral estimate) and evaluate her own option price based on this factor.

2.2 Options

In this section, we will provide some basic notions about European options and how they are priced. An option is a financial contract which provides to its holder the right of buying or selling certain assets at an agreed future price (i.e. strike price). The one who sells (writes) them takes the liability to fulfil buy or sell requests in exchange for the premium he receives. European options are exercised upon their maturity date. An option allowing its holder to buy is named a *call option*, and allowing to sell is a *put option* [8]. Depending on the present value of its strike price K and the current price of its underlying asset S_t, options can be classified into *Out-of-The-Money* (OTM), *At-The-Money* (ATM) and *In-The-Money* (ITM) options. The table below illustrates the types of options that are traded in exchanges. We can also define the upper and lower boundaries for option valuation in Eqs. (3) and (4).

Option Types

	OTM	ATM	ITM
CALL	$K > S_t$	$K = S_t$	$K < S_t$
PUT	$K < S_t$	$K = S_t$	$K > S_t$

$$\max(S_t - K, 0) \leq c \leq S_t \quad (3)$$

$$\max(K - S_t, 0) \leq p \leq K \quad (4)$$

For simplicity reasons, we will assume that the risk-free interest rate is zero, so money has no time value. Also there is no friction in the market, so options can be sold and bought at the same price without any transaction costs involved.

There is an established relationship between put and call options with the same strike price and maturity date. This relationship results from the possibility of buying the one and selling the other. Consider a case, when trader buys a call option at K strike price, and at the same time sells a put option with K strike price, and both have the same maturity T. In some sense, it seems that trader can compensate the cost of a call option he bought for with the premium he received for selling put. So on maturity date, S_T turns out to be higher than strike price K, so the trader can benefit profit as a difference of $S_T - K$. However if S_T appears to be less than K, then trader has a liability to fulfil the put option that he sold, so he incurs a loss of $K - S_T$. This market position actually simulates a forward contract which could be obtained for free. This type of contract is free because it involves future possible liability or profit at the same time, so the risk for both parties is even. Once the combination of put and call options can replicate the liabilities of a forward contract, the prices for put and call options must hold the *put-call parity* relationship: $(c + K = p + S_T)$ [7]. Using the put-call parity relationship, we can easily convert call prices to put prices, and vice versa.

3 Design of Exchange Mechanisms

In this section, we will propose design of a multi-unit multi-type direct DA auction for pricing options and provide some future perspectives on its implementation using CE settings. We will start with McAfee's *Single-Unit Single-Type Double Auction* and gradually reduce it to an option pricing DA and CE mechanisms. In both mechanisms, we comply with the Myerson's lemma to make them DSIC, and thus the agent bids are equal to actual valuations ($\mathbf{b}_i = \mathbf{v}_i$). Vector $\mathbf{v}_i$ will represent the valuation of OTM, ATM and ITM call options by trader i. We will describe how trader valuations are drawn from the distribution, and used to determine the future forecast. We will also show an algorithm for selecting *option trading strategy* (OTS) based on agent's valuation vector. This will determine how demand and supply quantities are formed in the market.

3.1 Valuations and Bidding

In DA mechanism, we will be running a two-sided auction where traders can submit bids and asks to trade one type of option in multiple quantities. We can run several DAs in parallel to determine the pricing of different types of options independently from each other. Agents must represent their orders in terms of two matrices: $V = \{v_{ij} \in \mathbb{R}^+; \forall i \in N, \forall j \in G\}$ for valuations, and $Q = \{q_{ij} \in \mathbb{Z}; \forall i \in N, \forall j \in G\}$ for quantities requested.

Valuations are obtained from agent forecasts. We define agent i's forecast on the future price of underlying asset as a geometric Brownian motion without drift. We have already made an assumption that risk-free rate is zero which frees us from adjusting the prices for their time values. Below geometric process defines how agents obtain their forecasts.

$$S_{i,T} = S_0 e^{(-\frac{1}{2}\sigma^2 T + \sigma W_{i,T})} \quad \forall i \in N \tag{5}$$

Every agent calculates her own values for call options, and also translates those valuations to put options using put-call parity relationship mentioned earlier. Call options will be calculated for different strike prices K_j. It will form a valuation matrix $V = \{v_{ij} = (S_{i,T} - K_j)^+; \forall i \in N, \forall j \in G\}$.

For determining the quantities to be ordered for different types of options, first we need to consider option trading strategies (OTS). These are common combinations of options to be bought or sold in order to minimise the risk of loss. OTSs are frequently, if not every time, used by traders in major real-world option exchanges such as CBOT[1], Eurex[2], etc. Therefore we will assume that our virtual traders will use the same strategies while trading in the market. OTS can be represented as $\mathbf{q}_i \in Q$ for agent i, as it shows the units of options to be bought or sold. Some of the major OTSs, but not all of them, are listed in Table 1 where the quantity of option type to buy or sell is specified in positive or negative numbers respectively. The table also tells about the forecast direction of each OTS, so agents can choose OTS based on their forecast. For example if agent's forecast is in between some $S_0 - \epsilon \leq S_{i,T} \leq S_0 + \epsilon$, then agent will choose *neutral strategy*. If $S_{i,T} > S_0 + \epsilon$, then agent will choose *bullish strategy*. And finally if $S_{i,T} < S_0 - \epsilon$, the agent will choose bearish strategy. Traders pick random strategy among strategies with same direction. However some OTSs can be both bullish and bearish such as Long Straddle, so both bullish and bearish

[1] Chicago Board of Trade, http://www.cmegroup.com/company/cbot.html.
[2] Eurex Group, http://www.eurexchange.com/exchange-en/.

traders can be interested in this OTS. It is also possible that OTS is more bullish, than bearish, and vice versa. For example, Strip generates greater payoff when prices go up. Therefore there is a biased chance for a bearish trader to choose Strip among other bearish OTSs because it is less bullish.

Table 1. Option trading strategies

Name	c_{ATM}	p_{ATM}	c_{OTM}	p_{OTM}	c_{ITM}	p_{ITM}	Direction
Long Call	1	0	0	0	0	0	bullish
Long Put	0	1	0	0	0	0	bearish
Bull Call Spread	0	0	-1	0	1	0	bullish
Butterfly Put Spread	0	-2	0	1	0	1	neutral
Long Call Ladder	-1	0	-1	0	1	1	neutral
Long Put Ladder	0	-1	0	-1	0	1	neutral
Iron Butterfly	-1	-1	1	1	0	0	neutral
Long Straddle	1	1	0	0	0	0	bearish and bullish
Long Strangle	0	0	1	1	0	0	bearish and bullish
Strip	1	2	0	0	0	0	bullish > bearish
Strap	2	1	0	0	0	0	bearish > bullish

Also it is worth noting that we will regard option as ATM option if its strike price K_j is in ϵ vicinity of current asset price S_0. By definition of ATM option, its strike price must be equal to the current asset price, but because we only have discrete K_js in price line, we have to take this assumption. Strike prices beyond $[S_0 - \epsilon, S_0 + \epsilon]$ are either considered OTM or ITM.

We name the algorithm for selecting OTS as a $Strat(S_0, S_{i,T})$ function which returns $\mathbf{q}_i$ quantities to buy and sell. Thus the quantities matrix can be formed $Q = \{\mathbf{q}_i = Strat(S_0, S_{i,T}), \forall i \in N\}$. $Strat$ algorithm is defined below in Algorithm 1.

Algorithm 1. OTS Selection Algorithm

Require: $S_0, S_{i,T}, \epsilon$
  if $S_0 - \epsilon \leq S_{i,T} \leq S_0 + \epsilon$ **then**
    return random neutral OTS
  else if $S_{i,T} < S_0 - \epsilon$ **then**
    return random bearish OTS
  else if $S_{i,T} > S_0 + \epsilon$ **then**
    return random bullish OTS
  end if

3.2 Multi-Unit DA

In this section we will gradually extend McAfee's DA to a multi-unit auction, and apply it for option pricing using OTSs. McAfee's matching rule can be written as a greedy algorithm which sorts bids $b_{(1)} \geq b_{(2)} \geq \cdots \geq b_{(m)}$ and asks

$a_{(1)} \leq a_{(2)} \leq \cdots \leq a_{(n)}$ to satisfy $k \leq \min(m, n)$ such that $b_{(k)} \geq a_{(k)}$ and $b_{(k+1)} < a_{(k+1)}$ [13]. This rule assumes that bids and asks are for a single-unit of item. We can reformulate this rule to a LP problem defined below:

Definition 3. *For a given vector of valuations* $\mathbf{v}$, *McAfee's SM allocation rule for DA is*

$$\max_{\lambda} \sum_{i} v_i q_i \lambda_i \tag{6}$$

$$s.t. \quad \lambda_i \in \{0, 1\} \quad \forall i \tag{7}$$

$$\sum_{i} q_i \lambda_i = 0 \tag{8}$$

where $q_i \in \{-1, 1\}$ *represents sell/buy action by trader* i, λ_i *is an allocation decision variable.*

Theorem 1. *Allocation rule* (6) *generates exactly same number of* k *efficient trades as McAfee's greedy matching rule.*

Proof. Given that the the supply and demand is matched in constraint (8), we can assume that the number of trades is $m = (\sum_i \lambda_i)/2$, hence we have to prove $m = k$. Let's assume that $m < k$, then it means that there is $b_{(m+1)} - a_{(m+1)} > 0$ and SM solver could add this difference to result greater surplus. So it is not the maximum surplus, and there is a contradiction. Let's assume that $m > k$, then it would mean that $b_{(m)} - a_{(m)} < 0$, and SM solver would be better off not including this match into allocation, as it decreases the objective. Hence there is a contradiction in this case too. Therefore $m = k$. □

We will use McAfee's pricing rule to determine the clearing prices which conform with Myerson's payments.

Definition 4. *For a given vector of valuations* $\mathbf{v}$, *McAfee's DSIC payment rule is:*

$$p = \begin{cases} p^0 & \text{if } p^0 \in [a_{(k)}, b_{(k)}] \\ p^1 & \text{otherwise} \end{cases}$$

where

$p^0 = (b_{(k+1)} + a_{(k+1)})/2$, $\quad p^1 = (b_{(k)} + a_{(k)})/2$

$b_{(k+1)} = \sup(v_i; \lambda_i = 0, q_i = 1, \forall i)$ $\quad a_{(k+1)} = \inf(v_i; \lambda_i = 0, q_i = -1, \forall i)$

$b_{(k)} = \inf(v_i; \lambda_i = 1, q_i = 1, \forall i)$ $\quad a_{(k)} = \sup(v_i; \lambda_i = 1, q_i = -1, \forall i)$

McAfee mechanism rejects $b_{(k)}$ and $a_{(k)}$ match when the clearing price is p^1, and thus it looses one efficient trade. This efficient trade makes the least portion of the overall surplus. However this makes the mechanism DSIC, individual rational and budget-balanced for single-unit single-type bids and asks. It is individual rational because whenever the p^0 exceeds the boundary of $[a_{(k)}, b_{(k)}]$, it uses p^1 price which is always in between the winning bid-ask spread. It is budget-balanced because it uses anonymous prices to clear the market along with fully matched supply and demand.

From the valuation of options (see the definition of matrix V), we know that traders value call options in monotonic strictly increasing function of their private prediction $S_{i,T}$. Then we can use *Revelation Principle* to convert McAfee's DSIC mechanism to another DSIC mechanism, let us name it *Predictions Matching* (PM) mechanism where traders disclose their private predictions to the mechanism designer, instead of submitting their valuations for every option. Hence we can find the aggregated predictions of asset price S_T at the maturity of option. Mechanism then can use this aggregated prediction to determine the price for any type of option, and clear the market. Also this would restore the lost information about predictions on the valuation of OTM calls, as they are valued zero if agent's prediction is below option's strike price.

Now let us extend McAfee's mechanism to multi-unit mechanism. For simplicity sake, we will assume that bids (asks) $b_i = S_{i,T}$ are agent i's prediction of asset prices at T. Then consider multi-unit bid as a tuple (b_i, q_i). We can split this tuple into set of same-valued bids $\mathbf{b}_i = \bigcup_{t=1}^{q_i} b_{i,t}$ where $b_{i,t} = b_{i,t'}, \forall = t, t'$. This can be done to asks as well. Then we will have complete set of bids $\mathbf{b} = \bigcup_{i=1}^{n} \mathbf{b}_i$ and asks $\mathbf{a} = \bigcup_{i=1}^{n} \mathbf{a}_i$. We can use single-unit McAfee's mechanism mentioned above to find SM allocation, and DSIC payments. However, we can observe below that not all bids/asks can be fully satisfied. If the bids are atomic then the mechanism will loose the efficiency from discarding partially satisfied bids. In case of OTS based bids, we assume that bids are indivisible, so the mechanism has to either satisfy fully or discard the bid. Moreover, by discarding the partially satisfied bids, mechanism also incurs into the cost of covering the exposed asks which has been matched to discarded bids. So it will make the mechanism not budget-balanced.

Lemma 1. *In extended multi-unit McAfee's mechanism, there exists at most one multi-unit bid/ask which is partially satisfied, and the remaining winning bids/asks are fully satisfied.*

Proof. Let us assume that we use McAfee's single-unit DA matching rule for expanded set of bids $\mathbf{b}$ and asks $\mathbf{a}$. Then we should have some k such that $b_{(k)} \geq a_{(k)}$ and $b_{(k+1)} < a_{(k+1)}$. We can also claim, without loss of generality, that there is a bid $\mathbf{b}_i$ such that $b_{(k)}, b_{(k+1)} \in \mathbf{b}_i$. This would imply that $b_{(k)} = b_{(k+1)}$. However, it cannot be $a_{(k)} = a_{(k+1)}$ because it contradicts $b_{(k+1)} < a_{(k+1)}$. Hence, $a_{(k)}$ and $a_{(k+1)}$ belong to different asks, and it must be the case that the multi-unit ask which owns $a_{(k)}$ is fully satisfied, and so do other preceding winning multi-unit bids and asks. □

Using Lemma 1, we can formulate an LP problem for SM allocation of multi-unit bids and asks. This would involve changing decision variable from binary to continuous $\lambda_i \in [0, 1]$. Below is the definition.

Definition 5. *For given vectors of predictions and quantities* $(\mathbf{S}_T, \mathbf{q})$, *SM allocation rule for Multi-Unit DA is*

$$\max_{\lambda} \sum_i q_i \lambda_i S_{i,T} \tag{9}$$

$$s.t. \quad \lambda_i \in [0,1] \quad \forall i \tag{10}$$

$$\sum_i q_i \lambda_i = 0 \tag{11}$$

where $q_i \in \mathbb{Z}$ represents quantities, $S_{i,T}$ is the agent's prediction, λ_i is an allocation decision variable.

Given that the bids and asks are atomic, mechanism discards the partially satisfied bid/ask and covers the cost of exposed winning ask/bid. In this way, mechanism looses one partial multi-unit efficient bid. Without loss of generality, let us set $\mathbf{b}_l$ as the partially satisfied bid, and this implies the fact $b_{(k)} = b_{(k+1)}$ shown in Lemma 1. We can use p^0 defined in Definition 4 as long as it is within the bounds of winning bid-ask spread. However, mechanism has to reject both least winning multi-unit bid and ask, if p^0 exceeds the bounds. This may expose preceding bid $\mathbf{b}_{l-1}$ to be partially satisfied. Then mechanism will cover the cost of fully satisfying $\mathbf{b}_{l-1}$. The key difference in between the actions of mechanism for partially satisfied bids $\mathbf{b}_l$ and $\mathbf{b}_{l-1}$ is that it rejects the former, and covers the latter. In other words, the mechanism is responsible for covering the costs of rejecting the efficient trades. Rejecting both bid and ask at the edge would allow us to use them to compute the clearing price p^1 defined in Definition 4. As a matter of note, by p^0 and p^1 we mean not the price of an option, but the estimated predications $\hat{S}_T$ which can be used to determine the intrinsic value of any option.

Theorem 2. *Multi-Unit DA is DSIC, individual rational and at most looses one efficient multi-unit trade.*

Proof. Mechanism is DSIC is because it follows the Myerson's lemma, as it has monotonic SM allocation rule and its payment is the critical value of winning bids and asks. It is individual rational because it uses p^0 when it does not exceed the winning bounds. It discards efficient trade and use its prices to obtain p^1 when individual rationality bounds exceeded. There is only one case when mechanism discards both efficiently matched bid and ask, and this case is when the average of offsetting bid and ask is not individual rational. Therefore it approximates the efficiency of the mechanism up to a single efficient multi-unit trade.

As we have already mentioned, mechanism is not budget-balanced, and it may generate negative cash flow. However, we will closely examine how it progresses over the time if mechanism is allowed to keep record of its cash flows and inventory. For example, if mechanism discards partial multi-unit bid, then it will need to satisfy the exposed ask by buying out the remaining options. Mechanism then can use these bought options to satisfy exposed bids later in the time frame. For instance, when mechanism ends with over-supply having multi-unit ask partially satisfied. Mechanism will take options from its inventory to satisfy exposed bids. Below Algorithm 2 summarises the Multi-Unit DA:

Algorithm 2. Multi-Unit DA

Require: $\mathbf{S}_T, \mathbf{q}$
  Determine SM λ
  Discard λ_l bid or ask
  Calculate p^0 and p^1
  if $p^0 \in [b_k, a_k]$ **then**
    Clear the market with p^0
  else
    Discard remaining bid/ask at the edge
    Clear the market with p^1
  end if
  if Has inventory(cash) to cover exposed bid(ask) **then**
    Cover the exposed bid(ask) from inventory(cash)
  else
    Cover the exposed bid(ask) at mechanisms cost, update inventory(cash).
  end if

3.3 Multi-item Multi-Unit DA

In this section, we will extend our Multi-Unit DA further to accommodate multi-item bids and asks as well. In options case, traders would be interested in taking OTSs and this would involve different types of options such as OTM call, ATM put etc. We have defined several commonly used OTSs as a potential candidates for multi-item bids in Sect. 3.1. We will model a mechanism which is based on our previous multi-unit DA which also allows trading multiple heterogeneous items.

We will consider 2 cases of markets: multi-unit multi-item DA where traders can disclose their linear valuations of options to market maker; and CE where traders only disclose their valuation for the bundle. In both cases, traders would want to have their bids satisfied fully. In this multi-unit multi-item DA setup, we will have valuation and quantities matrices (V, Q) to represent the trader preferences. And in CE setting, traders will define their preferences as a tuple of valuation vector and quantities matrix $(\mathbf{v}, Q)$. We will again use Revelation Principle to turn option valuations into predictions in both cases, once we assert that both mechanisms are DSIC for valuations.

Let us consider several multi-unit DAs run in parallel for different items. Traders can simultaneously participate in all of them. In such setup, the overall SM outcome can be viewed as the sum of SM outcome for each DA. So let us construct LP allocation rule for this mechanism:

Definition 6. *For given valuations and quantities* (V, Q)*, SM allocation rule for multi-unit multi-item DA is*

$$\max_{\lambda} \sum_i \sum_j v_{ij} q_{ij} \lambda_{ij} \tag{12}$$

$$s.t. \quad \lambda_{ij} \in [0, 1] \quad \forall i \in N, \forall j \in G \tag{13}$$

$$\sum_i q_{ij} \lambda_{ij} = 0, \quad \forall j \in G \tag{14}$$

where λ_{ij} *determines the allocation of each option to each trader.*

However it follows from Lemma 1 that there will be at most G number of partially satisfied bids/asks and the mechanism has to discard those multi-unit multi-item bids/asks in order to avoid traders partially executing their corresponding OTSs. It can use the same pricing method per option type as it has been described for multi-unit DA. Hence it can also inherit DSIC and individual rationality from multi-unit DA.

In CE mechanism, the valuations come for bundles and are usually expressed through indirect means such as bidding languages because communicating the valuation for all combination of possible bundles is exponentially large amount of data which requires much memory and computing resources to process. Nisan provides a good analysis of existing bidding languages [10] used in combinatorial auctions. But in order to avoid this complexity, we will assume that the trader's combinatorial bid space is a predefined list of OTSs and the trader can only choose one of them to participate in CE. Also, like in previous cases, trader want his OTS fully satisfied. In this way, we can represent traders preferences using one valuation vector $\mathbf{v} = \{v_i \in \mathbb{R}; \forall i \in N\}$, and one quantities matrix $Q = \{q_{ij} \in \mathbb{Z}; \forall i \in N, \forall j \in G\}$. If bidders use linear valuations for combinatorial bids, the CE problem can be reduced to multiple DAs. In case of options, the value of OTS is calculated through summing up the elements of the OTS. Moreover, every options' intrinsic value is dependent only on agent i's prediction $S_{i,T}$. Hence, mechanism designer can determine agent's prediction from the OTS value and quantities she submits. Below is the formula for calculating the value of OTS:

Definition 7. *If odd j represents call option, and even j represents put option, the linear value of OTS for agent i is*

$$v_i = \sum_j ((-1)^{j+1}(S_{i,T} - K_j))^+ q_{ij} \tag{15}$$

Given all variables except $S_{i,T}$, the mechanism designer can numerically solve the Eq. (15), and find corresponding $S_{i,T}$ for every bidder. Then mechanism designer can use Revelation Principle to find estimated prediction $\hat{S}_T$ for calculating the individual prices of options. Below is the transformation of Definition 6 to a PM mechanism:

Definition 8. *For given predictions and quantities $(\mathbf{S}_T, Q)$, SM allocation rule for multi-unit multi-item DA and CE is*

$$\max_{\lambda} \sum_i \sum_j q_{ij} \lambda_{ij} S_{i,T} \tag{16}$$

$$s.t. \quad \lambda_{ij} \in [0, 1] \quad \forall i \in N, \forall j \in G \tag{17}$$

$$\sum_i q_{ij} \lambda_{ij} = 0, \quad \forall j \in G \tag{18}$$

where λ_{ij} determines the allocation of each option to each trader.

It can be noted that allocation of G options will result at most G number of partially satisfied bids/asks. This would mean that we will have at most G estimated predictions $\hat{S}_T$ for every type of option. In order find the clearing estimated prediction we can calculate the weighted average of $\hat{S}_T$ by trade volume. In other words, we can have $\hat{S}_T$ adjusted based on the bullishness and bearishness of traders. This is the key part of our experiment, to observe how the use of OTSs may result in the change of overall estimated predication. We will provide series of experimental results to test this effect.

As a matter of caveat, we also admit that OTS can be valued in non-linear fashion, and options can be substitutes or compliments. In case if they are substitutes, then it has been shown by Roughgarden [2] that the above mechanism will beat the surplus produced from substitutes, and hence can be used to determine the SM allocation for combinatorial bids with substitute goods. However it is much more complex task and out of the scope of this paper to design a mechanism where goods are compliments, as it would require iterative rounds of price discovery and package bidding. Also it is important to note that options can be compliments and there is enough evidence to assert this assumption. There is an established phenomenon called *volatility smile* which exhibits abnormally higher prices for OTM options in major derivatives markets, whereas their intrinsic value is zero [8]. They can even be valued higher than ATM options. This forms a convex parabola for *implied volatility* as the strike price increases. Implied volatility can be calculated through finding the root of Black-Scholes formula for volatility σ using the resulted option price and other known parameters.

4 Experimental Setup and Results

We will conduct series of experiments to see how estimated predictions, and consequently the option prices change in multi-unit DA and CE mechanisms. In first set of experiments with multi-unit DA, we will simulate asset prices as Brownian process, and then use it for different market settings defined below:

- *Vol = Vol, Supply = Demand:* In this setting the real asset price volatility, and the implied volatility for agents are the same. Also supply and demand scalers are taken from a random variable $\lfloor 15 * z \rfloor$ where $z \sim \mathcal{N}(0, 1)$. This balances the supply and demand the market around zero.
- *Vol = Vol, Supply > Demand* : The same as above, except supply and demand scalers are taken from $\lfloor 15 * z - 5 \rfloor$ where $z \sim \mathcal{N}(0, 1)$. This balances the market around 5 oversupply.
- *Vol = Vol, Supply < Demand* : The same as above, except supply and demand scalers are taken from $\lfloor 15 * z + 5 \rfloor$ where $z \sim \mathcal{N}(0, 1)$. This balances the market around 5 overdemand.
- *Vol ≠ Vol, Supply = Demand:* The same as above, except implied volatility of traders differ around real asset price volatility with lognormal standard deviation of 0.5.

We fix several other parameters for the experiment. For example, options have constant strike prices throughout the timeline. This means that agents will trade

Table 2. Parameters of the experiment

Name	Value	Name	Value
Initial Asset Price	$S_0 = 100$	Random Quantities Scaler Range	[-15, 15]
Strike Price	$K = 100$	Shift in Supply/Demand per Agent	5
Deviation from Strike price	$\epsilon = 10$	Random Implied Volatility Mean	0
Asset Price Volatility	$\sigma = 0.05$	Random Implied Volatility St.D	0.5
Risk-free rate	$r = 0$	Number of agents	$N = 100$
Time to maturity	$T = 100$	Number of option types	$G = 6$
Number of tests per mechanism	$w = 30$		

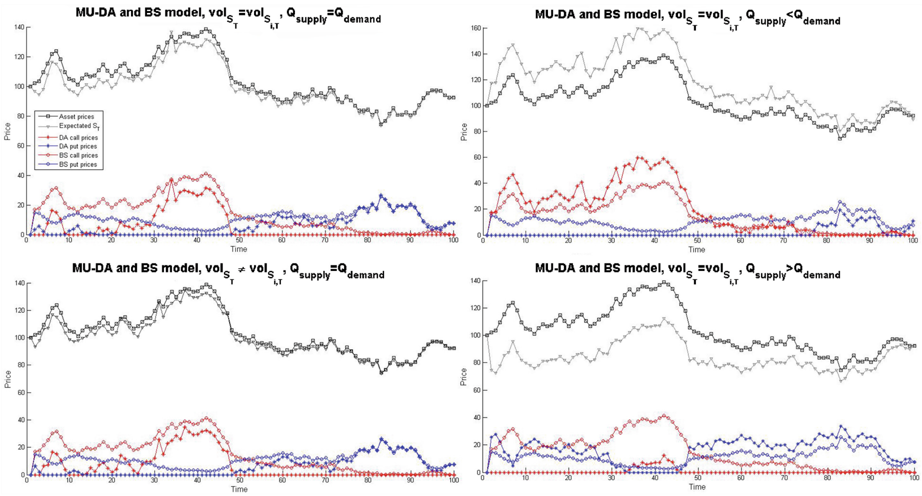

Fig. 1. Call and put prices from multi-unit DA mechanism and Black-Scholes model

only with predefined set of options at the beginning of the simulation, and no new type of option with new strike price will enter the market. Also option maturity date will be constant, and it will approach its maturity date through the timeline of the simulation. Asset price volatility will also be fixed (Table 2).

In Fig. 1 we can see estimated predictions change when implied volatility, supply and demand are different. It illustrates that multi DA mechanism can effectively simulate Black-Scholes prices, as long as the implied volatility is the same as the asset price volatility, and supply and demand are equal. However we can see that call prices drop blow Black-Scholes model when the supply exceeds demand, and vice versa. We can also observe that randomised implied volatility around real asset price volatility can better approximate option prices.

Another set of experiments reveals the key aspect of the research exhibiting the effect of OTSs on estimated predictions. In this experiment we simulate CE mechanism, and calculate the estimated predictions as weighted average of estimated predictions obtained for different option types through simultaneously executed multi-unit DAs. In this set, we consider following cases:

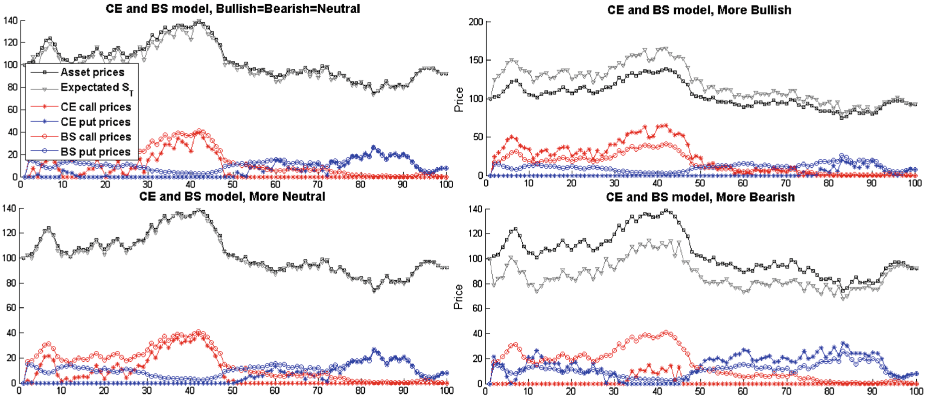

Fig. 2. Call and put prices from CE mechanism and Black-Scholes model

- *Balanced Bullish, Bearish and Neutral Traders*: In this setup, traders use OTSs equally having balanced quantities for every OTS.
- *More Bullish Traders*: Traders use more bullish OTSs compared to other OTSs.
- *More Bearish Traders*: Traders use more bearish OTSs compared to other OTSs.
- *More Neutral Traders*: Traders use more neutral OTSs compared to other OTSs.

Figure 2 illustrates the estimated predictions obtained from simulating CE mechanism where traders use OTSs to interpret their predictions. It also shows the corresponding option prices compared to Black-Scholes model. As it was expected, we can observe that estimated predictions are higher when traders are more bullish, and lower if they are more bearish. Also we can see that estimated predictions stick up well with the asset prices when traders are more neutral. This clearly shows that option prices are affected by the choice of OTSs in the market, although OTS is not purely a buy/ask order, but it is mixed combination of bids and asks for particular options.

As we have already mentioned, proposed mechanisms are not budget-balanced and it is worthwhile to view how they yield loss and profit from covering the partial bids/asks of rejected traders. Figure 3 shows the accumulated cost and revenue for multi-unit DA and CE mechanisms.

It can be seen from the Fig. 3 that in cases of oversupply in multi-unit DA or more bullish traders in CE the revenue of the mechanism is soaring, because there are fewer bids than asks, and the mechanism always ends up partially satisfying some seller. As a result it rejects that seller, and takes its role of selling options to exposed bidder. Hence it increase its revenue day after day. The opposite phenomena happens when there are more bids than asks, and mechanism has to spend money on behalf of rejected bidder to buy out exposed ask. Mechanisms are somewhat stabilised around zero when the balance of supply and demand is maintained. Also it is interesting to observe that in CE, the mechanism revenue/cost is more volatile and enormous because the volumes of options traded are at least G times bigger.

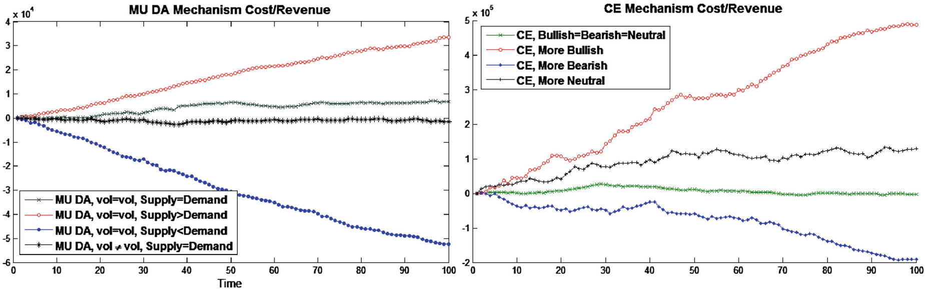

Fig. 3. Cost/Revenue for multi-unit DA and CE mechanisms

5 Concluding Remarks

In this paper, we have gradually designed two important mechanisms (multi-unit DA and CE) based on McAfee's description of single-unit DA. Although designed mechanisms are not budget-balanced, we have proved that they are DSIC, individual rational and approximately efficient. We have used these mechanisms to price options where traders not only bid in price and quantities, but also apply various commonly used OTSs to minimise their risks. The experiments gave us results where demand and supply can also affect the option prices, and more importantly, we saw that the OTSs have a considerable impact in forming option prices. We have also highlighted the revenue and cost of the mechanisms under various scenarios, and found out that mechanism is stable as long as the supply and demand in the market are balanced.

References

1. Satterthwaite, M.A., Myerson, R.B.: Efficient mechanisms for biliteral trading. J. Econ. Theor. **28**, 265–281 (1983)
2. Roughgarden, T.: Lectures on Combinatorial Auctions. Lecture Notes on Topics in Algorithmic Game Theory, Stanford (2008)
3. Myerson, R.: Optimal auction design. Mathematics of Operations Research **6**(1), 5873 (1981)
4. Parsons, S., Rodriguez-Aguilar, J.A., Klein, M.: Auctions and bidding. ACM Comput. Surv. **43**(2), 1–59 (2011). doi:10.1145/1883612.1883617
5. Shoham, Y., Crampton, P., Steinberg, R.: Combinatorial Auctions. MIT Press, Cambridge (2010)
6. Baqueiro Espinosa, O.: Agent Risk Management in Electronic Markets using Option Derivatives. PhD thesis, Department of Computer Science, University of Liverpool (2008)
7. Higham, D.: An Introduction to Financial Option Valuation: Mathematics, Stochastics and Computation. Cambridge University Press, Cambridge (2004)
8. Hull, J.C.: Options, Futures and Other Derivatives, 5th edn. Prentice Hall, Englewood cliffs (2001)

9. King, A.J., Streltchenko, O., Yesha, Y.: Using multi-agent simulation to understand trading dynamics of a derivatives market. Ann. Math. Artif. Intell. **44–3**, 233–253 (2005)
10. Nisan, N.: Bidding languages for combinatorial auctions. In: Crampton, P., Shoham, Y., Steinberg, R. (eds.) Combinatorial Auctions. The MIT Press (2006)
11. DeMarzo, P., Kremer, I., Mansour, Y.: Online trading algorithms and robust option pricing. In: Proceedings of the 38th Annual ACM Symposium on Theory of Computing, pp. 477–486. ACM, New York (2006)
12. Berg, J.E., Forsythe, R., Nelson, F.D., Rietz, T.A.: Results from a dozen years of election futures markets research. In: Plott, C.A., Smith, V.L. (eds.) Handbook of Experimental Economic Results, vol. 1, pp. 742–751. Elsevier, Amsterdam (2008)
13. McAfee, R.P.: A dominant strategy double auction. J. Econ. Theor. **56**, 434–450 (1992)
14. Shoham, Y., Leyton-Brown, K.: Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, Cambridge (2008)