Algorithmic Operations Research Vol.5 (2010) 65–74

Computing general static-arbitrage bounds for European basket options via Dantzig-Wolfe decomposition

Javier Peña [a] Xavier Saynac [b] Juan C. Vera [c] Luis F. Zuluaga [b]

[a]Tepper School of Business, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA, USA 15213
[b]Faculty of Business Administration, University of New Brunswick, Fredericton, NB, Canada E3B 5A3
[c]Tilburg School of Economics and Management, Tilburg University, Tilburg, The Netherlands

Abstract

We study the problem of computing general static-arbitrage bounds for European basket options; that is, computing bounds on the price of a basket option, given the only assumption of absence of arbitrage, and information about prices of other European basket options on the same underlying assets and with the same maturity. In particular, we provide a simple efficient way to compute this type of bounds by solving a large finite non-linear programming formulation of the problem. This is done via a suitable Dantzig-Wolfe decomposition that takes advantage of an integer programming formulation of the corresponding subproblems. Our computation method equally applies to both upper and lower arbitrage bounds, and provides a solution method for general instances of the problem. This constitutes a substantial contribution to the related literature, in which upper and lower bound problems need to be treated differently, and which provides efficient ways to solve particular static-arbitrage bounds for European basket options; namely, when the option prices information used to compute the bounds is limited to vanilla and/or forward options, or when the number of underlying assets is limited to two assets. Also, our computation method allows the inclusion of real-world characteristics of option prices into the arbitrage bounds problem, such as the presence of bid-ask spreads. We illustrate our results by computing upper and lower arbitrage bounds on gasoline/heating oil crack spread options.

Key words: option pricing, robust optimization, Dantzig-Wolfe decomposition, semiparametric bounds, large scale optimization.

1. Introduction

Computing bounds for option prices under incomplete market conditions or an incomplete knowledge of the distribution of the price of the underlying assets is a widely studied pricing technique, where in contrast to *parametric* pricing techniques, such as Monte Carlo simulations or "Black-Scholes pricing," strong assumptions about the underlying asset price distribution are not required. These type of *semiparametric* bounds provide a mechanism for checking consistency of prices, and to provide estimates for option prices in incomplete market conditions, or regardless of any model specifics. Also, these bounds are useful when the number of underlying assets makes the computation of parametric prices numerically challenging, or when the scarcity of data makes it difficult to make distributional assumptions about the future assets returns. For example, the first condition applies to the problem of pricing complex spread options (see, e.g., [2]) and index options. The latter condition is typical in Actuarial Science applications and some real option problems.

Here, we consider a particular class of semiparametric bounds. Specifically, we study the problem of computing general *static-arbitrage* bounds for European basket options; that is, computing bounds on the price of a basket option, given the only assumption of absence of arbitrage, and information about prices of other European basket options on the same underlying assets and with the same maturity. The problem of computing this type of static-arbitrage bounds has received a fair amount of attention in recent years. Of particular relevance to our work are the articles by [1,3,4,6,7,11,9,10,15] and [18]. (Throughout the article we will make more precise references to their work as

Email: Javier Peña [jfp@andrew.cmu.edu], Xavier Saynac [q79j1@unb.ca], Juan C. Vera [j.c.veralizcano@uvt.nl], Luis F. Zuluaga [lzuluaga@unb.ca].

[1] (Corresponding author)

we present our results.) Here, we provide a simple *efficient* way to compute general static-arbitrage bounds for European basket options by solving a *large* finite non-linear programming formulation of the problem. This is done via a suitable Dantzig-Wolfe decomposition that takes advantage of an integer programming formulation of the corresponding subproblems. Our computation method equally applies to both upper and lower arbitrage bounds, and provides a solution method for general instances of the problem. This constitutes a substantial contribution to the related literature, in which upper and lower bound problems need to be treated differently, and which provides efficient ways to solve particular static-arbitrage bounds for European basket options; namely, when the option prices information used to compute the bounds is limited to vanilla and/or forward options, or when the number of underlying assets is limited to two assets. Also, our computation method allows the inclusion of real-world characteristics of option prices into the arbitrage bounds problem, such as the presence of bid-ask spreads.

The reminder of the article is organized as follows. In Section 2., we formally introduce the problem of computing general static-arbitrage bounds for European basket options, and review how the problem can be formulated as a finite non-linear program. In Section 3., we show how this non-linear programming formulation can be solved via a suitable Dantzig-Wolfe decomposition. In Section 4., we discuss the solution of the related super or sub-replicating portfolio strategy. In Section 5., we illustrate the effectiveness of our solution approach by computing upper and lower arbitrage bounds on gasoline/heating oil crack spread options. We finish in Section 6., discussing some straight-forward extensions of the presented results, and future work.

2. Formulation of general static-arbitrage bound problem

Consider the problem of computing sharp lower and upper static-arbitrage bounds on the price of a European basket option, given information on the prices of other European basket options with the same maturity, without making any assumptions other than the absence of arbitrage. Finding the sharp lower static-arbitrage bound for the price of a basket option can be formulated as the following optimization problem (see, e.g., [3]):

$$\begin{aligned} \inf_{\pi}\ & \mathbb{E}_\pi[(w^0 \cdot X - K_0)^+] \\ \text{s.t. } & \mathbb{E}_\pi[1] = 1 \\ & \mathbb{E}_\pi[(w^j \cdot X - K_j)^+] = p_j,\ j = 1, ..., r \\ & \pi \text{ a distribution in } \mathcal{D}. \end{aligned} \tag{1}$$

The corresponding sharp upper static-arbitrage bound can be formulated as follows:

$$\begin{aligned} \sup_{\pi}\ & \mathbb{E}_\pi[(w^0 \cdot X - K_0)^+] \\ \text{s.t. } & \mathbb{E}_\pi[1] = 1 \\ & \mathbb{E}_\pi[(w^j \cdot X - K_j)^+] = p_j,\ j = 1, ..., r \\ & \pi \text{ a distribution in } \mathcal{D}. \end{aligned} \tag{2}$$

In (1) and (2), the multivariate random variable $X := (X_1, \dots, X_n)$ represents the prices at maturity of the n underlying assets in the basket of interest. The given vectors $w^j \in \mathbb{R}^n$, and constants $K_j \in \mathbb{R}$, $j = 0, 1, \dots, r$, respectively represent the weights of the underlying assets and the strike price of the basket options involved in the problem. The corresponding payoffs of the basket options are $(w^j \cdot x - K_j)^+ := \max\{0, \sum_{i=1}^n w_i^j x_i - K_j\}$, $j = 0, 1, \dots, r$, where we are using the conventional dot $(\cdot)$ product of vectors, and lower-case $x \in \mathbb{R}_+^n$ to represent an outcome of the random asset prices at maturity X. Problem (1) (problem (2)) minimizes (maximizes) the expected payoff of a target basket option – defined by weights w^0, and strike K_0 – over all underlying asset price distributions (π) with support set $\mathcal{D} \subseteq \mathbb{R}_+^n$ that *replicate* the given basket option's prices p_j, $j = 1, \dots, r$; that is, the discounted expected payoffs of the given options match the observed prices. Following [3], we implicitly assume that all the options have the same maturity, and that without loss of generality, the risk-free interest rate is zero; or equivalently, we compare the prices in the forward market (at maturity). The static-arbitrage bound problems (2) and (1) are feasible if the given basket option's prices are arbitrage free (for further details on feasibility see [15, Proposition 1]).

The problem of computing sharp static-arbitrage bounds has received a fair amount of attention in recent years. Typically, the problem is studied in the case when the support set $\mathcal{D} = \mathbb{R}_+^n$ (non-negative asset prices). In particular, [7, Section 4] derived a closed-form solution to the upper bound problem (2) in the special case when the weights of the basket of interest are non-negative (i.e., $w^0 \in \mathbb{R}_+^n$), and the given option prices are composed by forward and any number of vanilla options on each of the underlying assets (i.e., for $j = 1, \dots, r$, w^j has a single non-zero component equal to 1). This result was re-derived by [1, Section 3]. The latter results in turn generalize the results of

[3, Proposition 4], and [11] who derive a closed-form solution to problem (2) in the special case when the weights of the basket of interest are non-negative and the given option prices are composed of a forward and one vanilla options on each of the underlying assets. [9, Section 3] solve problem (2) for the special case in which the target option is a two-asset spread option (i.e., $n = 2$, and the weights defining target basket option are of opposite signs), and the given option prices are composed by forward and vanilla options on each of the underlying assets. Less general results have been derived for the lower bound problem (1). In particular, Laurence and Wang [11] provide closed-form solutions for problem (1) in the two-asset case given forward prices of the assets and prices of one vanilla call option for each asset. Hobson et al. [6] present a numerical procedure for the special case with only two assets, given information about prices for a continuum of vanilla call options for each asset. A. d'Aspermont and L. El Ghaoui [3] give an efficient linear programming formulation for the computation of the lower static arbitrage bound for the special case when only one vanilla call per asset is known. Laurence and Wang [10] solve problem (1) for the special case in which the target option is two-asset spread option, and the given option prices are composed by forward and vanilla options on each of the underlying assets. Pena at al. [15] generalize some of the latter results by giving tractable linear programming formulations for problem (1) in the special case of two assets and any number of given basket option prices (not necessarily vanilla options), as well as for the special n-asset case when only a forward and/or a vanilla call per asset is given.

In what follows we show how to efficiently solve the general arbitrage bounds problems (1) and (2) when the support set is given by *box* constraints:

$$\mathcal{D} := [0, u_1] \times [0, u_2] \times \cdots \times [0, u_n], \tag{3}$$

where $u_i > 0$, $i = 1, \ldots, n$ are given bounds on the asset prices at maturity. That is, instead of the typical choice in the related literature of considering that the underlying asset prices at maturity can take any non-negative value, here we consider that the asset prices at maturity have some given upper bound; that is, ruling out asset price distributions where prices are unbounded.

Problems (1) and (2) are semi-infinite programs (infinite dimensional variable with finite number of constraints). However, it is well-known that these problems can be reformulated using a finite (although in general very large) number of variables and constraints (for recent examples of these finite formulations, see e.g., [3,10,15]). In particular, when the support set $\mathcal{D}$ is defined by the box constraints (3), it is simple to show that problems (1) and (2) can be reformulated as finite non-linear programs by using the following remark.

Proposition 1 *Let I be an index set, and $R_i : i \in I$ be a partition of $\mathcal{D}$. For any (piece-wise linear) function $f : \mathcal{D} \to \mathbb{R}$ such that f restricted to R_i is linear for each $i \in I$, we have that*

$$\begin{aligned}\mathbb{E}_\pi[f(X)] &= \sum_{i\in I} \mathbb{E}_\pi[f(X)|X \in R_i]\pi(X \in R_i)\\ &= \sum_{i\in I} f[\mathbb{E}_\pi (X|X \in R_i)]\pi(X \in R_i).\end{aligned}$$

If R_i is convex and bounded then for each i we have $\mathbb{E}_\pi[X|X \in R_i] \in R_i$.

Remark 1 *Proposition 1 implies that given a suitable partition of the support set $\mathcal{D}$, one can, without loss of generality, assume that the underlying asset price distribution (π) in problems* (1) *and* (2) *is atomic, with one atom located in each of the sets defining the partition of $\mathcal{D}$. In other words, thanks to the piece-wise linearity of the basket option payoffs, the probability distribution in a region can be "concentrated" into a single point in the region.*

In order to continue our discussion, let us introduce the following notational conventions. Let $\mathcal{J} := \{J \subseteq \{0, 1, \ldots r\}\}$ be the set of all subsets of the index set $\{0, 1, \ldots, r\}$ (where r is the number of given basket option prices in problems (1) and (2)). Also, let W be the $(r + 1) \times n$ matrix whose j-th row is the vector w^j for all $j \in \{0, 1, \ldots, r\}$, $K := (K_0, K_1, \ldots, K_r)^T$, and $u := (u_0, u_1, \ldots, u_r)^T$. Given $v \in \mathbb{R}^{r+1}$, and $J \in \mathcal{J}$, let $v_J \in \mathbb{R}^J$ denote the vector formed by the entries v_j with $j \in J$. Likewise, for a matrix M with rows indexed by $\{0, 1, \ldots, r\}$ and $J \in \mathcal{J}$, let M_J denote the matrix formed by the rows of M indexed by J. Finally, For $J \in \mathcal{J}$, we shall write J^c as a shorthand for $\{0, 1, \ldots, r\} \setminus J$.

A suitable partition (recall Proposition 1) of the support set $\mathcal{D}$ in (3) can be obtained by considering regions of the possible asset price values at maturity (i.e, regions in $\mathcal{D}$), in which each of the given basket options, as well as the target option, are either always out-of-the-money or always in-the-money in the region. Note that in such regions the payoff of all the options in the problem are linear: either 0 if the option is out-of-the money, or $(w^j \cdot x - K_j)$ if the option is in-the-money. Specifically, we define the following partition for $\mathcal{D}$ in (3):

$$R_J := \left\{x \in \mathbb{R}^n : W_J x \geq K_J, W_{J^c} x \leq K_{J^c}, \vec{0} \leq x \leq u\right\}, \tag{4}$$

for all $J \in \mathcal{J}$. Note that in (4), J represents the set of options that are in-the-money in region R_J, and J^c represents the set of options that are out-of-the-money in region R_J. Clearly, R_J is convex and bounded for all $J \in \mathcal{J}$.

From now on, we will concentrate our discussion on the solution method for the lower bound problem (1); which has been the more challenging problem in the literature (as noted e.g., in [11,15]). The slight differences in the solution method for the upper bound problem (2) will be addressed in Section 3.1.. Using Remark 1, together with the partition (4), one can reformulate problem (1) as the following non-linear program:

$$\begin{aligned} \min \; & \sum_{J\in\mathcal{J}:0\in J} \left(w^0 \cdot x^J - K_0\right) t^J \\ \text{s.t.} \; & \sum_{J\in\mathcal{J}} t^J = 1 && \text{(5a)} \\ & \sum_{J\in\mathcal{J}:j\in J} \left(w^j \cdot x^J - K_j\right) t^J = p_j \quad j = 1,\dots,r && \text{(5b)} \\ & x^J \in R_J, \qquad J \in \mathcal{J} && \text{(5c)} \\ & t^J \geq 0, \qquad J \in \mathcal{J} && \text{(5d)} \end{aligned}$$

Above, for all $J \in \mathcal{J}$, x^J and t^J respectively represent the position of the atom and the probability of the atom in region R_J of the partition of $\mathcal{D}$.

The main difficulty in solving (5) is the size of the problem (the non-linearity could be tackled by defining new variables $u^J := x^J t^J$); namely, the problem has $(r+1+n)|\mathcal{J}| = \mathcal{O}((r+n)2^r)$ constraints, and $(n+1)|\mathcal{J}| = \mathcal{O}(n2^r)$ variables. To deal with the size of the problem we exploit the block structure of constraints (5c) and (5d) by solving the problem via a suitable Dantzig-Wolfe decomposition that is discussed in the next section.

3. Dantzig-Wolfe decomposition

In this section, we apply a Dantzig-Wolfe decomposition approach (see, e.g., [12, Section 3.9]) to solve the non-linear formulation of the lower bound problem (5). This is done by designating the constraints (5a), (5b) as the "hard" constraints that will define the *master problem*, and designating the constraints (5c), (5d) as the "easy" constraints that will define the *subproblems*.

In order to specifically define the master problem and corresponding subproblems we proceed in typical fashion. Namely, for each $J \in \mathcal{J}$ let $\{\hat{x}^{J,k} : k = 1,\dots,N_J\}$ be the set of extreme points of R_J (recall that from (4), R_J is a polytope). Then problem (5) can be written as

$$\begin{aligned} \min \; & \sum_{J\in\mathcal{J}:0\in J} \left(w^0 \cdot \left(\sum_{k=1}^{N_J} \lambda_{J,k}\hat{x}^{J,k}\right) - K_0\right) t^J \\ \text{s.t.} \; & \sum_{J\in\mathcal{J}} t^J = 1 \\ & \sum_{J\in\mathcal{J}:j\in J} \left(w^j \cdot \left(\sum_{k=1}^{N_J} \lambda_{J,k}\hat{x}^{J,k}\right) - K_j\right) t^J = p_j \\ & \qquad j = 1,\dots,r \\ & \sum_{k=1}^{N_J} \lambda_{J,k} = 1 \qquad J \in \mathcal{J} \\ & \lambda_{J,k} \geq 0,\ t^J \geq 0 \qquad J \in \mathcal{J}. \end{aligned} \tag{6}$$

Using $t^{J,k} := t^J \lambda_{J,k}$, the problem above is equivalent to:

$$\begin{aligned} \min \; & \sum_{J\in\mathcal{J}:0\in J} \sum_{k=1}^{N_J} \left(w^0 \cdot \hat{x}^{J,k} - K_0\right) t^{J,k} \\ \text{s.t.} \; & \sum_{J\in\mathcal{J}} \sum_{k=1}^{N_J} t^{J,k} = 1 \\ & \sum_{J\in\mathcal{J}:j\in J} \sum_{k=1}^{N_J} \left(w^j \cdot \hat{x}^{J,k} - K_j\right) t^{J,k} = p_j \\ & \qquad j = 1,\dots,r \\ & t^{J,k} \geq 0, \qquad J \in \mathcal{J}, k = 1,\dots,N_J. \end{aligned} \tag{7}$$

With the "decomposed" formulation (7) of the lower bound problem (5), we can now state the Dantzig-Wolfe decomposition solution algorithm.

Given a subset of the extreme points of R_J, for all $J \in \mathcal{J}$:

$$\mathcal{X} := \bigcup_{J\in\mathcal{J}} \{\tilde{x}^{J,k} : k = 1,\dots,M_J\} \subseteq \bigcup_{J\in\mathcal{J}} \{\hat{x}^{J,k} : k = 1,\dots,N_J\}$$

(where for some J, M_J could possibly be zero, indicating that no extreme points from region R_J are included in the subset of extreme points $\mathcal{X}$), define the *master problem*

$$
\begin{aligned}
\min\ & \sum_{J\in\mathcal{J}:0\in J}\sum_{k=1}^{M_J}\left(w^0\cdot\tilde{x}^{J,k}-K_0\right)t^{J,k} \\
\text{s.t.}\ & \sum_{J\in\mathcal{J}}\sum_{k=1}^{M_J} t^{J,k}=1 \\
& \sum_{J\in\mathcal{J}:j\in J}\sum_{k=1}^{M_J}\left(w^j\cdot\tilde{x}^{J,k}-K_j\right)t^{J,k}=p_j \\
& j=1,\ldots,r, \\
& t^{J,k}\geq 0, J\in\mathcal{J}, k=1,\ldots,N_J.
\end{aligned}
\qquad (M_{\mathcal{X}})
$$

Given τ, $\rho:=(\rho_1,\ldots,\rho_r)$ and $J\in\mathcal{J}$ define the *subproblem*

$$
\begin{aligned}
\min\ & \\
& (x\cdot w^0-K_0)\mathbb{I}_{0\in J}-\tau-\sum_{j\in J,j>0}\rho_j(x\cdot w^j-K_j) \\
\text{s.t.}\ & x\in R_J
\end{aligned}
\qquad (Q_J^{\tau,\rho})
$$

where the indicator function $\mathbb{I}_{0\in J}$ is 1 if $0\in J$, and 0 otherwise.

Note that both the master problem $(M_{\mathcal{X}})$, and the subproblems $(Q_J^{\tau,\rho})$ are linear programs for all $J\in\mathcal{J}$. Therefore, to solve (7) we use the following Dantzig-Wolfe (DW) algorithm, where $\epsilon_{\text{tol}}>0$ is the user provided error tolerance, and $\mathcal{X}_0$ is a subset of the extreme points of R_J, for all $J\in\mathcal{J}$, such that $(M_{\mathcal{X}_0})$ is feasible. (Such $\mathcal{X}_0$ can be found by running a *Phase I* version of the DW algorithm, where the objective in the master problem is to minimize the infeasibility in the price replicating constraints.)

Dantzig-Wolfe algorithm

(1) SET $\mathcal{X}:=\mathcal{X}_0$, and REPEAT = TRUE.
(2) WHILE REPEAT DO
  (2.1) SOLVE $(M_{\mathcal{X}})$, LET $\tau\in\mathbb{R}$ be the shadow price of the first equality constraint of $(M_{\mathcal{X}})$, and $\rho\in\mathbb{R}^r$ be the shadow prices of the r price replicating equality constraints of $(M_{\mathcal{X}})$.
  (2.2) LET $Q_J^{\tau,\rho*}$ be the optimal value of $(Q_J^{\tau,\rho})$. FIND $J^*\in\mathcal{J}$ such that $J^*=\arg\min\{Q_J^{\tau,\rho*}:J\in\mathcal{J}\}$.
  (2.3) IF $Q_J^{\tau,\rho*}\leq-\epsilon_{\text{tol}}$
      LET $\mathcal{X}:=\mathcal{X}\cup\{\arg Q_{J^*}^{\tau,\rho}\}$.
    ELSE
      REPEAT = FALSE
(3) RETURN the ϵ_{tol}-optimal value $M_{\mathcal{X}^*}$ of $(M_{\mathcal{X}})$

As shown in detail in Section 4. (see equation (12)), the DW algorithm above returns a value that is within the user provided error tolerance ϵ_{tol} of the optimal value of (1). However, since the size of $\mathcal{J}$ is generally exponential on r, it is prohibitively expensive to execute STEP (2.2) above by solving the $|\mathcal{J}|$ linear programming subproblems. Instead, we use a mixed integer programming formulation of the subproblems in order to efficiently execute STEP (2.2) of the DW algorithm. Specifically, consider the following mixed integer program related to the subproblems $(Q_J^{\tau,\rho})$, where $\rho_0:=-1$, and $M_j', M_j>0$, $j=0,1,\ldots,r$ are large enough given constants.

$$
\begin{aligned}
\min\ & -\tau-\textstyle\sum_{j=0}^r\rho_j(w^j\cdot z^j-y_jK_j) \\
\text{s.t.}\ & w^jx\geq K_j-M_j'(1-y_j)\quad j=0,1,\ldots,r \\
& w^jx\leq K_j+M_j'y_j\quad j=0,1,\ldots,r \\
& \\
& z_i^j\geq x_i-M_j(1-y_j) \\
& i\in\{1,\ldots,n\}, j\in\{0,1,\ldots,r\}:\rho_jw_i^j\leq 0 \\
& \\
& z_i^j\leq M_jy_j \\
& i\in\{1,\ldots,n\}, j\in\{0,1,\ldots,r\}:\rho_jw_i^j\geq 0 \\
& \\
& 0\leq z_i^j\leq x_i\leq u_i \\
& i=1,\ldots,n, j=0,1,\ldots,r \\
& \\
& y_j\in\{0,1\}\quad j=0,1,\ldots,r.
\end{aligned}
\qquad (P^\rho)
$$

Solving (P^ρ) is equivalent to executing STEP (2.2), provided that

$$M_j>\max\{u_1,\ldots,u_n\},$$

and

$$M_j'>\max\left\{\sum_{i\in\{1,\ldots,n\}:w_i^j>0}w_i^ju_i-K_j,\right.$$

$$\left.\left|\sum_{i\in\{1,\ldots,n\}:w_i^j<0}|w_i^j|u_i-K_j\right|\right\},$$

for $j=0,1,\ldots,r$. Specifically, if x^*,y_j^*,z_i^{j*} is the optimal solution of (P^ρ) with objective value $P^{\rho*}$, then in STEP (2.2) of the DW algorithm, $J^*=\{j\in$

$\{0,1,\dots,r\} : y_j^* = 1\}$, and $Q_J^{\tau,\rho *} = P^{\rho *}$, and in STEP (2.3) of the DW algorithm, $\arg Q_{J^*}^{\tau,\rho} = x^*$. This follows by noticing that the "big-M constraints" in (P^ρ) ensure that for any $J \in \mathcal{J}$, if one replaces $y_j = 1$ if $j \in J$, and $y_j = 0$ if $j \notin J$, for $j = 0,1,\dots,r$ in (P^ρ), then the resulting linear program is equivalent to ($Q_J^{\tau,\rho}$).

Loosely speaking, solving (P^ρ) is much more efficient than solving ($Q_J^{\tau,\rho}$) for all $J \in \mathcal{J}$, because doing the latter is equivalent to solving (P^ρ) by enumerating all the possible $0-1$ solutions in (P^ρ). In practice, current MIP solvers are typically able to solve MIP by enumerating only a *small* number of $0-1$ solutions (compared to the total number of possible $0-1$ solutions).

3.1. *Upper bound problem*

Thus far, we have concentrated our discussion on the lower bound arbitrage problem (1). This is because unlike related results in the literature, our computation method applies similarly for the upper bound arbitrage problem (1). In fact, in order to solve the upper bound problem (2), one only needs to change the discussed solution method for the lower bound arbitrage problem (1) as follows:

- In (5), (7), ($M_{\mathcal{X}}$), ($Q_J^{\tau,\rho}$) change $\min \mapsto \max$
- In STEP (2.2) change $\arg\min \to \arg\max$.
- In STEP (2.3) of the Dantzig-Wolfe algorithm change $\le -\epsilon_{\text{tol}} \ \mapsto \ \ge \epsilon_{\text{tol}}$
- In (P^ρ) change $\min \mapsto \max$, and in the first set of "big-M" constraints for the z_i^j variables, change $\rho_j w_i^j \le 0 \to \rho_j w_i^j \ge 0$, and in the second set of 'big-M" constraints for the z_i^j variables, change $\rho_j w_i^j \ge 0 \to \rho_j w_i^j \le 0$.

4. Super-replicating and sub-replicating portfolios

Problems (1) and (2) respectively have the following associated duals (see, [8]):

$$\begin{aligned}\sup_{\tau,\rho} \ & \tau + \sum_{j=1}^r p_j \rho_j \\ \text{s.t.} \ & \tau + \sum_{j=1}^r \rho_j (w^j \cdot x - K_j)^+ \le (w^0 \cdot x - K_0)^+ \\ & \text{for all } x \in \mathcal{D} \\ & \rho \in \mathbb{R}^r, \quad \tau \in \mathbb{R},\end{aligned} \tag{8}$$

and

$$\begin{aligned}\inf_{\tau,\rho} \ & \tau + \sum_{j=1}^r p_j \rho_j \\ \text{s.t.} \ & \tau + \sum_{j=1}^r \rho_j (w^j \cdot x - K_j)^+ \ge (w^0 \cdot x - K_0)^+ \\ & \text{for all } x \in \mathcal{D} \\ & \rho \in \mathbb{R}^r, \quad \tau \in \mathbb{R},\end{aligned} \tag{9}$$

The dual problem (8) (problem (9)) has a natural interpretation: it aims to find the most expensive (cheapest) portfolio of positions in cash (τ) and positions in the given basket options (ρ) with payoff $(w^j \cdot S - K_j)^+$, $j = 1,\dots,r$ that sub-replicates (super-replicates) the payoff of the basket option of interest with payoff $(w^0 \cdot S - K_0)^+$. It is easy to see that weak duality holds between (1) and (8), and between (2) and (9). Furthermore, thanks to the compactness of the support of the asset price distribution $\mathcal{D}$ considered here (see (3)), strong duality also holds between these problems.

Proposition 2 *Let* $\mathcal{D}$ *be as in* (3)*. Then the optimal values of* (1) *and* (8)*, and of* (2) *and* (9) *coincide.*

Proof. Follows from general convex duality results (see, e.g., [16,17]), as it is discussed in [19, Sec. 4, Proposition 4.2]. □

The choice of labels for the variables in (8) and (9) is not accidental. In fact, it is easy to see that from the values of τ and ρ obtained at the end of the DW algorithm discussed in Section 3., one can construct a feasible solution for the sub-replicating problem (8), whose objective value is within the user provided error tolerance ϵ_{tol} of the optimal value of (8). The same follows for the super-replicating problem (9) with the corresponding modified DW algorithm explained in Section 3.1.. To see that this is the case for the sub-replicating problem (8), let τ^* and ρ^* be the values of the shadow prices τ and ρ, of the master problem ($M_{\mathcal{X}^*}$) obtained at the end of the DW algorithm. From the stoping rule of the algorithm (STEP (2.3)) it follows that for all $J \in \mathcal{J}$, and for all $x \in R_J$

$$\begin{gathered} -\epsilon_{\text{tol}} \le (x \cdot w^0 - K_0)\mathbb{I}_{0\in J} - \tau^* - \\ \sum_{j \in J, j>0} \rho_j^* (x \cdot w^j - K_j) \\ \Updownarrow \\ -\epsilon_{\text{tol}} \le (x \cdot w^0 - K_0)^+ - \tau^* - \\ \sum_{j=1}^r \rho_j^* (x \cdot w^j - K_j)^+ \\ \Updownarrow \\ (\tau^* - \epsilon_{\text{tol}}) + \sum_{j=1}^r \rho_j^* (x \cdot w^j - K_j)^+ \\ \le (x \cdot w^0 - K_0)^+ \end{gathered} \tag{10}$$

that is, the pair $(\tau^* - \epsilon_{\rm tol}, \rho^*)$ is a feasible solution for (8) with objective value $(\tau^* - \epsilon_{\rm tol}) + \sum_{j=1}^r p_j \rho_j^*$. From this, and the fact that (τ^*, ρ^*) is the optimal solution of the linear programming dual of $(M_{\mathcal{X}^*})$ with objective value $\tau^* + \sum_{j=1}^r p_j \rho_j^*$, it follows that:

$$\begin{aligned} \tau^* + \sum_{j=1}^r p_j \rho_j^* = M^*_{\mathcal{X}^*} \geq \text{opt. value of (1)} = \\ \text{opt. value of (8)} \geq (\tau^* - \epsilon_{\rm tol}) + \sum_{j=1}^r p_j \rho_j^*, \end{aligned} \tag{11}$$

where $M^*_{\mathcal{X}^*}$ denotes the optimal value of $(M_{\mathcal{X}^*})$. From (10), (11), it follows that $(\tau^* - \epsilon_{\rm tol}, \rho^*)$ is a feasible solution for (8), whose objective value is within the user provided error tolerance $\epsilon_{\rm tol}$ of the optimal value of (8). From (11), it also follows that the DW algorithm returns a value $M^*_{\mathcal{X}^*}$ that is within the user provided error tolerance $\epsilon_{\rm tol}$ of the optimal value of (1). Specifically:

$$0 \leq M^*_{\mathcal{X}^*} - \text{optimal value of (1)} \leq \epsilon_{\rm tol}. \tag{12}$$

A similar "$\epsilon_{\rm tol}$-optimality" follows for the upper bound problem when the modified DW algorithm of Section 3.1. is used.

5. A simulated numerical experiment

We next present a simulated computational experiment that illustrates some of our results. The objective of the experiment is to highlight two of our main contributions. First, that unlike current results in the literature, our proposed method allows to efficiently compute upper arbitrage bounds for European basket options, when given prices of other Europen basket options that are not restricted to be forward, call, or put options. Second, that our approach allows the computation of the corresponding lower arbitrage bound. As will be seen from our results, the possibility to use given prices that are not restricted to be vanilla options; for example, exchange options, can result in much tighter arbitrage bounds.

Related numerical results are presented in [3,7], where the authors provide extensive numerical experiments comparing static-arbitrage pricing techniques and parametric pricing techniques (such as Monte Carlo simulations) for basket options.

Here, we compute upper and lower arbitrage bounds on the price of a *crack spread* option. Quoting [2, Section 2.3], "A crack spread is the simultaneous purchase or sale of crude against the sale or purchase of refined petroleum products... They were introduced in October 1994 by the NYMEX with the intent of offering a new risk management tool to oil refiners. These spreads are computed on the daily futures prices of crude oil, heating oil, and unleaded gasoline." In particular, we compute upper and lower arbitrage bounds on the price of a *3:2:1 crack spread* call option with strike K_0, whose payoff is given by (following [2, Section 2.3]):

$$\begin{aligned} &p_{3:2:1}(K_0) = \\ &\left(\tfrac{2}{3}[UG]_T + \tfrac{1}{3}[HO]_T - [CO]_T - K_0\right)^+, \end{aligned}$$

where $[UG]_T$, $[HO]_T$, and $[CO]_T$ denote the prices at maturity time T of a futures contract of unleaded gasoline, heating oil, and crude oil, respectively. First, we will compute upper and lower arbitrage bounds on the price of *3:2:1 crack spread* call options, when given information about unleaded gasoline, heating oil, and crude oil forward and call option prices. The payoffs of these options are given by:

$$\begin{aligned} p_{1:0:0}(K) &= ([UG]_T - K)^+ \\ p_{0:1:0}(K) &= ([HO]_T - K)^+ \\ p_{0:0:1}(K) &= ([CO]_T - K)^+. \end{aligned}$$

The forward option payoffs are obtained when $K = 0$. Second, we will compute upper and lower arbitrage bounds on the price of *3:2:1 crack spread* call options, when given information about unleaded gasoline, heating oil, and crude oil future and call option prices, as well as information about the price of a *1:1:0 gasoline crack spread* call option with strike 0 (i.e., a exchange option), whose payoff is given by (following [2, Section 2.3]):

$$p_{1:1:0} = ([UG]_T - [CO]_T)^+,$$

and information about the price of a *1:0:1 heating oil crack spread* call option with strike 0 (i.e., a exchange option), whose payoff is given by (following [2, Section 2.3]):

$$p_{1:0:1} = ([HO]_T - [CO]_T)^+.$$

To set up the upper and lower bound problems, we sample the given option price values, by assuming that the underlying commodity prices distribution follows a correlated multivariate lognormal distribution (see, e.g., eq. (15) in [3]). In particular we use a riskless interest rate $r = 0$; option maturity $T = 5$ months; current commodity prices in dollars per gallon $[UG]_0 = 1.7809$, $[HO]_T = 1.9544$, and $[CO]_T = 1.7105$; volatilities $\sigma_{UG} = 0.3532$, $\sigma_{HO} = 0.3364$, $\sigma_{CO} = 0.3376$; and correlations $\rho_{UG,CO} = 0.86$, $\rho_{HO,CO} = 0.88$. These values are based on July 2009 estimates for December 2009 future energy commodity prices. Thus, using Black-Scholes formula we obtain the given vanilla options prices shown in Table 1 that will be used to compute the arbitrage bounds for the *3:2:1 crack spread* call options. The strike values are obtained by multiplying the current commodity price by $0, 0.5, 0.8, 1.0, 1.2, 1.5$, to obtain for each commodity a forward option, two (2) out-of-money call options, one (1) at-the-money call option, and two (2) in-the-money call options.

Similarly, using Magrabe's formula (see [13]), we obtain the given spread options prices shown in Table 2 that will be used to compute the arbitrage bounds for the *3:2:1 crack spread* call options.

Finally, we set

$$u_{UG} = u_{HO} = u_{CO} = 3\max\{[UG]_0, [HO]_0, [CO]_0\},$$

and we let K_0 range between $[0.4K_0^*, 1.6K_0^*]$, where $K_0^* = \left(\frac{2}{3}[UG]_0 + \frac{1}{3}[HO]_0 - [CO]_0\right)$ is the at-the-money strike for the 3:2:1 crack spread call option. With this data, we use the DW algorithm presented in Section 3. to obtain the bounds in Figure 1. The DW algorithm was implemented in `MATLAB`, and uses `ILOG-CPLEX Callable Library` with its default settings to solve all the corresponding linear programs, and integer programs; and `TOMLAB` to interface with `ILOG-CPLEX` on a `INTEL CORE 2 DUO`, `3GB RAM` computer. The time necessary to compute any of the bounds is under 3 seconds.

As it can be seen from Figure 1, for the 3:2:1 crack spread call options, the ability to use given prices that are not restricted to be vanilla options; such as spread options, results in much tighter arbitrage bounds than when only given vanilla option prices are used. Furthermore, the computation time of around 3 seconds shows the efficiency of the DW algorithm. Note that the formulation given in (5) for this 3:2:1 crack spread call option problem, where the number of underlying assets is $n = 3$, the number of given option prices is $r = 20$, and $|\mathcal{J}| = 2^{(20+1)}$, would have $(20 + 1 + 3)2^{(20+1)} \approx 50$-million constraints, and $(3+1)2^{(20+1)} \approx 8$-million variables (recall the discussion at the end of Section 2.).

6. Concluding Remarks

In practice, among others due to the presence of transaction costs, instead of option prices being uniquely defined, they display a so-called bid-ask spread. The presence of bid-ask spreads can be taken into consideration by modifying the lower arbitrage bound problem (1) as follows:

$$\begin{array}{ll} \inf\limits_{\pi} & \mathbb{E}_\pi[(w^0 \cdot X - K_0)^+] \\ \text{s.t.} & \mathbb{E}_\pi[1] = 1 \\ & \mathbb{E}_\pi[(w^j \cdot X - K_j)^+] \le p_j^{\text{ask}},\ j = 1, ..., r \\ & \mathbb{E}_\pi[(w^j \cdot X - K_j)^+] \ge p_j^{\text{bid}},\ j = 1, ..., r \\ & \pi \text{ a distribution in } \mathcal{D}, \end{array} \tag{13}$$

where p_j^{ask} represents the ask (buying) price, and p_j^{bid} represents the bid (selling) price of the given options (satisfying $p_j^{\text{ask}} \ge p_j^{\text{bid}}$) for $j = 1, ..., r$. It is not difficult to see that following for problem (13) a similar procedure to the one outlined in Section 2. and Section 3., one obtains a DW algorithm to efficiently solve the arbitrage bound problem (13). Furthermore, the shadow prices obtained at the end of the DW algorithm will provide the optimal solution to the sub-replicating problem (dual) corresponding to (13); which is given by:

$$\begin{array}{lll} \sup\limits_{\tau,\rho,\rho^{\text{ask}},\rho^{\text{bid}}} & \tau + \sum_{j=1}^{r}(p_j^{\text{ask}}\rho_j^{\text{ask}} - p_j^{\text{bid}}\rho_j^{\text{bid}}) & \\ \text{s.t.} & \tau + \sum_{j=1}^{r}\rho_j(w^j \cdot x - K_j)^+ \le & \\ & (w^0 \cdot x - K_0)^+ & \text{for all } x \in \mathcal{D} \\ & \rho = \rho^{\text{ask}} - \rho^{\text{bid}} & \\ & \rho \in \mathbb{R}^r, \rho^{\text{ask}}, \rho^{\text{bid}} \in \mathbb{R}_+^r \ \ \tau \in \mathbb{R}, & \end{array} \tag{14}$$

where now ρ^{ask} (ρ^{bid}) indicates the amount of given options in which to have long (short) positions on the sub-replicating portfolio. Notice that it is not possible to know a priori whether a given option will have a short or long position. This extension applies in similar fashion to the corresponding upper arbitrage bound problem in the presence of bid-ask spreads.

Using bid-ask prices in the computation of the upper/lower arbitrage bounds (and super/sub-replicating strategies) gives a more practical value to the arbitrage pricing approach. In particular, this resolves a major limitation in previous approaches (see, e.g., [3,7]) that used mid-market prices (e.g., the average of the bid and ask prices) as the "nominal" option prices. Such approximation systematically underestimates the actual buying prices and overestimates the actual selling prices. It is then not surprising that the market data used in [5, 12] requires a fair amount of "cleaning" to rule out apparent arbitrage opportunities created by these estimates (see [7, Section 6.2]). By contrast, the model herein that takes into account bid-ask spreads does not suffer from this limitation.

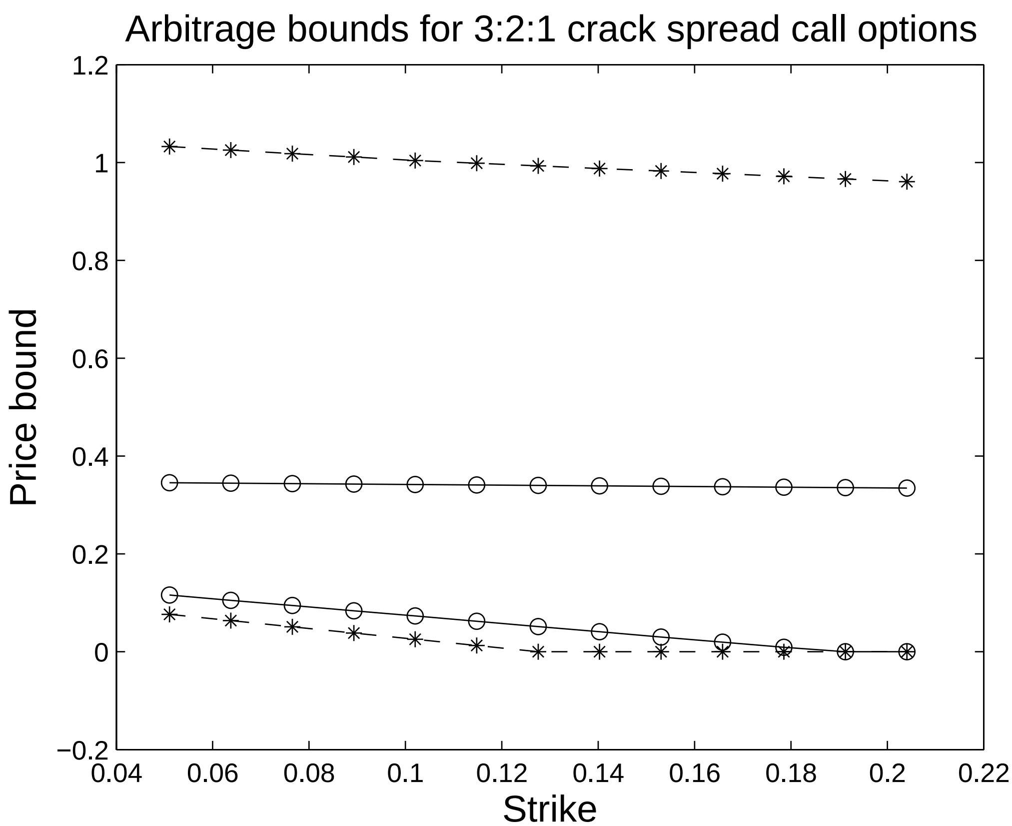
Table 1

Vanilla Options		
Unleaded Gasoline	Heating Oil	Crude Oil
$p_{1:0:0}(0) = 1.7809$	$p_{0:1:0}(0) = 1.9544$	$p_{0:0:1}(0) = 1.7112$
$p_{1:0:0}(0.8904) = 0.9897$	$p_{0:1:0}(0.9772) = 1.0733$	$p_{0:0:1}(0.8556) = 0.9405$
$p_{1:0:0}(1.4247) = 0.6887$	$p_{0:1:0}(1.5635) = 0.7325$	$p_{0:0:1}(1.3690) = 0.6428$
$p_{1:0:0}(1.7809) = 0.5469$	$p_{0:1:0}(1.9544) = 0.5730$	$p_{0:0:1}(1.7112) = 0.5034$
$p_{1:0:0}(2.1371) = 0.4386$	$p_{0:1:0}(2.3453) = 0.4523$	$p_{0:0:1}(2.0534) = 0.3978$
$p_{1:0:0}(2.6713) = 0.3206$	$p_{0:1:0}(2.9316) = 0.3229$	$p_{0:0:1}(2.5668) = 0.2845$

Table 2

Spread Options	
Unleaded Gasoline–Crude Oil	Heating Oil–Crude Oil
$p_{1:0:1} = 0.3198$	$p_{0:1:1} = 0.4070$

Fig. 1. Arbitrage bounds for 3:2:1 crack spread call options with strikes between $[0.05, 0.21]$: '$- * -$' indicates the bounds obtained when given the prices of the vanilla options in Table 1; '$-$o$-$' indicates the bounds obtained when also additionally given the prices of the spread options in Table 2.



A basic implementation of the DW algorithm presented here can effectively compute general upper/lower arbitrage bounds for European basket options in the commodity, energy, and currency markets – where basket options are most commonly traded – , thanks to the small number of assets involved in such basket options (typically less than ten assets). Currently, we are working on developing and testing a more sophisticated implementation of the DW algorithm presented here, in order to address the calculation of general upper/lower arbitrage bounds for index options where the number of assets involved in the baskets are of orders of magnitude larger than the basket options in the commodity, energy, and currency markets. In particular, we are working on the following enhancements for the basic implementation of the DW algorithm. First, notice that at the beginning of the DW algorithm's execution there is no need to solve (P^ρ) to optimality (i.e., finding a feasible solution with desirable reduced cost is enough). Second, it is not difficult to provide a warm-start feasible solution to speed up the solution time of (P^ρ). Third, and more importantly, thanks to the *block-ladder* structure of (P^ρ) (i.e., no constraint in (P^ρ) contains variables y_j and $y_{j'}$, or z^j and $z^{j'}$, with $j \neq j'$), a Benders decomposition algorithm (see, e.g., [14, Sections II.3.7 and II.5.4]) can be used to solve (P^ρ). Specifically, in (P^ρ) label the x variable as the *complicating* variables, and the z^j, y_j, $j = 1, \dots, r$ variables as the *non-complicating* variables to solve (P^ρ) via a Benders decomposition algorithm. Then, the Benders *subproblem* (obtained by fixing the values of the complicating variable x in (P^ρ)) can be decomposed into r problems (i.e., as many as given basket option prices), each with a single binary variable. Since each Benders subproblem has a single binary variable, a Benders decomposition algorithm similar to the one presented by [5, Section 4] can be used to solve (P^ρ), with a Benders *restricted master* problem that is a linear program on the complicating variable x.

References

[1] H. Albrecher, P. A. Mayer, and W. Schoutens. General lower bounds for arithmetic Asian option prices. *Applied Mathematical Finance*, 15(2):123–149, 2008.

[2] R. Carmona and V. Durrleman. Pricing and hedging spread options. *SIAM Review*, 45(4):627–685, 2003.

[3] A. d'Aspremont and L. El Ghaoui. Static arbitrage bounds on basket option prices. *Math. Program. A*, 106(3):467–489, 2006.

Received 12-12-2009; revised 20-8-2010; accepted 24-8-2010

[4] M. H. Davis and D. Hobson. The range of trading option prices. *Mathematical Finance*, 17(1):1–14, 2007.

[5] R. M. Freund. Benders? decomposition methods for structured optimization, including stochastic optimization. Technical report, Massachusetts Institute of Technology, 2004. available at `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.117.3003&rep=rep1&type=pdf`.

[6] D. Hobson, P. Laurence, and T. H. Wang. Static arbitrage optimal sub-replicating strategies for basket options. *Insurance Mathematics and Economics*, 37:553–575, 2005.

[7] D. Hobson, P. M. Laurence, and T. H. Wang. Static–arbitrage upper bounds for the prices of basket options. *Quant. Financ.*, 5(4):329–342, 2005.

[8] S. Karlin and W. Studden. *Tchebycheff Systems: with Applications in Analysis and Statistics*. Pure and Applied Mathematics Vol. XV, A Series of Texts and Monographs. Interscience Publishers, John Wiley and Sons, 1966.

[9] P. Laurence and T. H. Wang. Distribution-free upper bounds for spread options and market-implied antimonotonicity gap. *The European Journal of Finance*, 14(8):717–734, 2008.

[10] P. Laurence and T. H. Wang. Sharp distribution free lower bounds for spread options and the corresponding optimal subreplicating portfolios. *Insurance: Mathematics and Economics*, 44:35–47, 2009.

[11] P. M. Laurence and T. H. Wang. Sharp upper and lower bounds for basket options. *Applied Mathematical Finance*, 12(3):253–282, 2005.

[12] D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. third edition. Springer, 2008.

[13] W. Magrabe. The value of an option to exchange one asset for another. *Journal of Finance*, 3:177–186, 1978.

[14] G. L. Nemhauser and L. A. Wolsey. *Integer and combinatorial optimization*. Wiley, 1988.

[15] J. Peña, J. Vera, and L. Zuluaga. Static-arbitrage lower bounds on the prices of basket options via linear programming. *Quantitative Finance*, 10(8):819-827, 2010.

[16] J. Renegar. *A Mathematical View of Interior-Point Methods for Convex Optimization*. SIAM, 2001.

[17] T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1970.

[18] J. C. Vera, J. Peña, and L. F. Zuluaga. Static-arbitrage bounds on the prices of basket options via linear programming. Technical report, Carnegie Mellon University, 2006. available at `http://www.optimization-online.org/DB\_HTML/2006/07/1429.html`.

[19] L. F. Zuluaga and J. Peña. A conic programming approach to generalized Tchebycheff inequalities. *Math. Oper. Res.*, 30(2):369–388, 2005.