Algorithmic Finance 1 (2011) 13–16
DOI 10.3233/AF-2011-003
IOS Press

Binomial options pricing has no closed-form solution

Evangelos Georgiadis
M.I.T. Cambridge, MA 02139 USA
E-mail: egeorg@mit.edu

Abstract. We set a lower bound on the complexity of options pricing formulae in the lattice metric by proving that no general explicit or closed form (hypergeometric) expression for pricing vanilla European call and put options exists when employing the binomial lattice approach. Our proof follows from Gosper's algorithm.

The binomial model as heralded by Cox et al. (1979), has become a well-known approach to pricing options due to its simplicity as well as flexibility. Recently, Dai et al. (2008) demonstrated how combinatorial techniques can be applied to address the slow convergence issue in pricing options via the CRR approach. In particular, they provided a pricing algorithm that runs in linear time for a variety of options. Combinatorially, pricing options via this binomial lattice or binomial tree method can be viewed as an extended problem in lattice path counting. Costabile (2002) and Lyuu (1998)[1] demonstrated beautifully, successful applications of lattice path counting techniques in valuing options. In practice, enumerating lattice paths often leads to summation formulae involving binomial coefficients, factorials and rational functions. These types of summation formulae give rise to a large class of sums, namely, hypergeometric ones[2]. Hypergeometric sums, in turn, have been extensively studied in the mathematics literature and as we will see, some powerful tools have been developed for them. One such tool involves an algorithm that solves the problem of whether or not a given hypergeometric sum has a closed form. It turns out that our binomial options pricing formula can be expressed as such a sum and thereby analyzed in this setting. Thus, addressing the question of whether an explicit expression exists for our summation becomes natural not only from a computationally theoretic perspective that explores inherent limitations of the discrete metric but also from a practical point of view that strives for efficient computation.

To this end, Petkovšek et al. (1996)[3] developed a completely algorithmic solution. Practitioners can harness the power of PWZ's work, through a software implementation as proposed by Paule and Schorn (1994) or Zeilberger (1991).

Without loss of generality, our definition of closed-form or explicit expression is that of Petkovšek et al. (1996). A function $f(n)$ is said to be of (hypergeometric) closed form, if it is expressible as a linear combination of a fixed number, z, say, of hypergeometric terms in n. For instance, $\sum_{i=1}^{n} \frac{n+1}{i(n-i+1)}\binom{2i-2}{i-1}\binom{2n-2i}{n-i}$ (where $n \geq 1$) is not a closed form expression, but its

[1]Both authors incorrectly attribute the reflection principle to the French mathematician Désiré André. André's (1887) original solution to the famous ballot problem is based on a purely explicit counting argument without geometric insight. As a result, employing André's original argument for the development of the combinatorial identities at stake should be left as an exercise for the interested reader. Note that Renault (2008) provides an English translation of André's work.

[2]See section 4.4 of Petkovšek et al. (1996) for a precise definition of hypergeometric term.

[3]The interested reader should consult Chapter 5 in particular.

closed form exists, and can be derived with the use of Gosper's algorithm to be $\binom{2n}{n}$.[4]

Note that there seems to be no references in the finance literature that address the problem of formula complexity stemming from an inherently discrete approach. This paper helps to fill this gap.

In order to settle the question of whether a closed form expression exists for the vanilla European options when priced on the discrete binomial lattice, we first set the scene by establishing the formula in question. For simplicity, we consider options on stocks. A vanilla option gives the holder the right to buy or sell the underlying stock for price X defined in the option contract at the maturity date. A call option permits the owner of the option to buy the underlying stock for X dollars at time T, whereas a put options allows the owner of the option to sell the underlying stock for X at time T. The payoff functions as well as their pricing formulae are folklore. For convenience, we state the payoff function of a vanilla option as follows.

$$\max(\delta S(T) - \delta X, 0) \begin{cases} \text{for call option, we have } \delta = 1, \\ \text{for put option, we have } \delta = -1. \end{cases}$$

The theoretical option value is the expected value of the the payoffs discounted with the risk-free rate r, namely,

$$\exp(-rT)E(\text{payoff}).$$

Thus, the theoretical price of a vanilla option on an n-time step binomial lattice is,

$$\exp(-rT) \sum_{i=0}^{n} \binom{n}{i} p^{n-i}(1-p)^i \times \max(\delta S_0 u^{n-i} d^i - \delta X, 0), \tag{1}$$

where u and d denote the upward and downard multiplicative factors for the stock price, respectively. Correspondingly, p denotes the branching probability for an upward step whereas $(1-p)$ denotes the branching probability for a downward step. The numbers $\binom{n}{i}$ denote the binomial coefficients and are defined as follows

$$\binom{n}{i} = \begin{cases} \frac{n!}{i!(n-i)!} & \text{if } 0 \leq i \leq n, \\ 0 & \text{if } i > n \text{ or } i < 0, \end{cases}$$

for any $i, n \in \mathbb{Z}$ with $n \geq 0$.

Now, we are left to show that $\sum_{i=0}^{n} \binom{n}{i} p^{n-i}(1-p)^i \max(\delta S_0 u^{n-i} d^i - \delta X, 0)$ for $\delta \in \{-1, 1\}$ cannot yield a closed form. To do so, we consider the case when $\delta = 1$ (similar reasoning applies for $\delta = -1$) and employ Gosper's algorithm (1978). At this point, we should keep in mind that our sum for the European call option is expressible in terms of the complementary binomial distribution function, i.e. $\Phi[a; n, p]$, as presented by Cox et al. (1979). This notation induced formula however, does not constitute a closed form expression. In other words, the $\Phi[\cdot]$ notation merely disguises the underlying machinery involving explicit summation and dependency on n and thus does not help us to shortcut our computation. Gosper's algorithm deals with sums of the following type

$$s_n = \sum_{k=0}^{n-1} t_k, \tag{2}$$

where t_k is a hypergeometric term which does not depend on n. In other words, the consecutive term ratio

$$r(k) = \frac{t_{k+1}}{t_k}$$

is a rational function of k. Further note that,

$$s_{n+1} - s_n = t_n.$$

Gosper's algorithm provides an answer to the following question: Given a hypergeometric term t_n, is there a hypergeometric term z_n satisfying

$$z_{n+1} - z_n = t_n. \tag{3}$$

We quietly observe that any solution z_n of (3) will be of the form

$$z_n = z_{n-1} + t_{n-1} = z_{n-2} + t_{n-2} + t_{n-1} = \ldots = z_0 + \sum_{k=0}^{n-1} t_k = s_n + c,$$

where $c = z_0$ is a constant.

Thus if the algorithm generates an affirmative answer, then s_n can be expressed as a hypergeometric term plus a constant, and the algorithm outputs such a

[4] Alternatively, we could also establish this combinatorial identity in a less algorithmic manner, namely, with help of a (somewhat involved) lattice path argument.

term. On the other hand, if Gosper's algorithm returns a negative answer, then that proves that (3) has no hypergeometric solution.

Let us return to our sum,

$$s_n = \sum_{j=0}^{n} \binom{n}{j} p^{n-j}(1-p)^j \max\{Su^{n-j}d^j - X, 0\}. \quad (4)$$

We note that our sum, s_n, is not of hypergeometric form since it contains a max function. We can easily remedy this deficiency by re-expressing s_n as follows. Assume that $S, u, d, X > 0, d > u$ and $0 < p < 1$ (for other cases, an analogous argument can be made)[5]. Since

$$\max\{Su^{n-j}d^j - X, 0\} = \begin{cases} Su^{n-j}d^j - X, & \text{if } Su^{n-j}d^j \geq X, \\ 0, & \text{otherwise,} \end{cases}$$

and $Su^{n-j}d^j \geq X$ is equivalent to $j \geq j_0$ where

$$j_0 = \left\lceil \frac{\log \frac{X}{Su^n}}{\log \frac{d}{u}} \right\rceil,$$

we are able to re-write our sum, s_n, as follows.

$$\begin{aligned} s_n &= \sum_{j=j_0}^{n} \binom{n}{j} p^{n-j}(1-p)^j (Su^{n-j}d^j - X) \\ &= S \sum_{j=j_0}^{n} \binom{n}{j} p^{n-j}(1-p)^j u^{n-j} d^j \\ &\quad - X \sum_{j=j_0}^{n} \binom{n}{j} p^{n-j}(1-p)^j \\ &= S(pu)^n \sum_{j=j_0}^{n} \binom{n}{j} \left(\frac{(1-p)d}{pu}\right)^j \\ &\quad - Xp^n \sum_{j=j_0}^{n} \binom{n}{j} \left(\frac{1-p}{p}\right)^j \\ &= A\sigma_n(\alpha) - B\sigma_n(\beta), \end{aligned}$$

[5] We will ignore the uncompelling special case where p is zero or one and $u = d$.

where $A = S(pu)^n, B = Xp^n, \alpha = \frac{(1-p)d}{pu}, \beta = \frac{1-p}{p}$ are constants and

$$\sigma_n(x) = \sum_{j=j_0}^{n} \binom{n}{j} x^j = (1+x)^n - \sum_{j=0}^{j_0-1} \binom{n}{j} x^j \quad (5)$$

is a partial sum of the binomial series $(1+x)^n$. Our initial question is about the well-known sum (5). Our investigation concludes at the prospect of input $t(j) = \sum_{j=0}^{j_0-1} \binom{n}{j} x^j$ for which Gosper's algorithm fails. This part of the proof is computer assisted. This means there does not exist a solution over the field $\mathbb{Q}(n)$ for which $z_{j+1} - z_j = \binom{n}{j} x^j$. Put differently, the sum $t(j)$ is not expressible as a hypergeometric term over $\mathbb{Q}(n)$, plus a constant.[6] Thus, $\sigma_n(x)$ cannot be expressed as a hypergeometric term. As a result, our original sum, s_n, cannot yield a closed form expression.[7]

Acknowledgement

I cordially thank Dimitrios Kavvathas and Kostas Pantazopoulos of Goldman Sachs for a stimulating, inspiring and hospitable visit at the firm; David Page, for reminding me that PDEs can also compute; Brice Rosenzweig, for providing me with a succinct hands-on primer of the CRR model, and Lisa Opoku along with Katharina Koenig for their patience. In particular, I am grateful to Doron Zeilberger (2010, Private Communication) for his comments and source Gosper (1978), Marko Petkovšek for beautifying computation steps as well as Jean-Philippe Bouchaud, the Editor Phil Maymin and three anonymous referees for their careful comments that helped improve the overall exposition of this paper. Last but not least, I remain greatly indebted to Andrew Sutherland, Michael Sipser, John Tsitsiklis, Kiran Kedlaya and Denis Auroux of MIT.

[6] For a more in depth discussion, I encourage the reader to consult with Chapter 5 of Petkovšek et al. (1996), in particular p.88, which covers the case when $x = 1$.

[7] We note the trivial case that for certain assignments of the constants there might exist a closed-form expression, e.g. if $X \geq Su^{n-j}d^j$ for $j = 0, 1, \ldots, n$ then $s_n = 0$.
As a final remark, let us recall that the binomial option pricing formula converges in the limit to the Black-Scholes formula, which is known for its "explicit" or "closed-form solution".

References

André, D., 1887. Solution directed du problème résolu par M. Bertrand. Comptes Rendus de l'Académie des Sciences Paris 105, 436–437.

Costabile, M., 2002. A combinatorial approach for pricing Parisian options. Decisions in Economics and Finance 25 (2), 111–125.

Cox, J.C., Ross, S.A., Rubinstein, M., 1979. Option pricing: a simplified approach. Journal of Financial Economics 7, 229–263.

Dai, T.-S., Liu, L.-M., Lyuu, Y.-D., 2008. Linear-time option pricing algorithms by combinatorics. Computers and Mathematics with Applications 55, 2142–2157.

Gosper Jr., W.R., 1978. Decision procedure for indefinite hypergeometric summation. Proceedings of the National Academy of Sciences of the United States of America 75 (1) 40–42.

Lyuu, Y.-D., 1998. Very fast algorithms for barrier option pricing and the ballot problem. The Journal of Derivatives 5 (3), 68–79.

Petkovšek, M., Wilf, H.S., Zeilberger, D., 1996. A=B, A. K. Peters, Ltd., Wellesley, MA.

Paule, P., Schorn, M., 1995. A Mathematica© version of Zeilberger's algorithm for proving binomial coefficient identities. Journal of Symbolic Computation 20, 673–698.

Renault, M., 2008. Lost (and found) in translation: Andrés actual method and its application to the generalized ballot problem. American Mathematical Monthly 115, 358–363.

Zeilberger, D., 1991. A Maple© program for proving hypergeometric identities. SIGSAM. Bulletin, ACM Press, 25, 4–13.