An Efficient Implementation of Implied Binomial Trees[*]

Yisong S. Tian
Schulich School of Business
York University
4700 Keele Street
Toronto, Ontario
Canada, M3J 1P3
Email: ytian@schulich.yorku.ca
Phone: 416-736-2100 ext.77943

January 10, 2012

An Efficient Implementation of Implied Binomial Trees

Abstract

Implied binomial trees are constructed by fitting a risk-neutral density (in the form of ending nodal probabilities) to observed option prices. Since there are usually not enough options traded in the marketplace, a quadratic program is often used to extract the ending nodal probabilities from observed option prices. A problem with this commonly used approach is that the quadratic program is usually a high dimensional one and can be difficult or inefficient to solve. This is because numerical accuracy may require the construction of a large binomial tree, resulting in a high dimensional quadratic program as the number of unknowns (i.e., the number of ending nodal probabilities) is proportional to the number of binomial periods. In this paper, we propose a more efficient implementation of implied binomial trees by incorporating cubic spline smoothing in the quadratic program. Only a selected subset of ending nodal probabilities is treated as unknowns while the remainder is interpolated using cubic splines. This allows us to substantially reduce the dimension of the quadratic program and thus improve the efficiency of its numerical implementation. We demonstrate the effectiveness of the smoothed implied binomial tree method by evaluating its numerical accuracy in estimating the risk-neutral density and moments such as volatility, skewness and kurtosis.

I. Introduction

Implied binomial trees were developed and introduced to a wide audience by Rubinstein (1994). Implied binomial trees differ from their predecessors such as the Cox, Ross and Rubinstein (1979, CRR) binomial tree in that they are non-parametric in nature and designed to exactly match observed option prices. No assumption is made about the dynamics or distribution of the underlying stock price. The only requirement is that the constructed binomial tree must automatically price options correctly. This implied approach provides a viable alternative to traditional option pricing models in situations where it is unclear which parametric option pricing model (e.g., Black and Scholes (1973) and Merton (1973)) should be used. Unlike parametric models (and their underlying assumptions), model specification errors are ruled out from the outset in a non-parametric approach such as the implied binomial tree method.

The tradeoff or cost of this non-parametric approach is that implied binomial trees are much more difficult to construct than regular binomial trees (such as the CRR binomial tree). This is because an N-period implied binomial tree requires the estimation of $N+1$ unknown parameters while a regular binomial tree requires the estimation of only a single parameter (i.e., stock return volatility). The degree of difficulty escalates even further if an application requires a rather large implied binomial tree with many periods (e.g., $N = 200$) due to the high level of numerical accuracy demanded by the application. Solving a numerical problem with that many unknowns is clearly a daunting task, let alone the lack of confidence in any solution we may be lucky to find. In this paper, we propose a simple solution to this problem and develop a more efficient implementation method for implied binomial trees.

As discussed in Rubinstein (1994), an implied binomial tree is fully specified by the risk-neutral density of the underlying stock price distribution at option maturity. In other words, the construction of an implied binomial tree is equivalent to searching for a set of ending nodal probabilities (i.e., the discretized risk-neutral density) that exactly matches observed option prices. The ending nodal probabilities can be solved easily, and uniquely, if there are a sufficient number of strike prices available for traded options. To build an N-period implied binomial tree, we need to have option prices available for N-1 strike prices, ideally over a wide range of the support of the stock price distribution. In practice, however, this is almost never the case unless we build a small implied binomial tree with a small number of periods. As a result, the calibration of the ending nodal probabilities is typically set up as a quadratic program for

minimizing the sum of squared errors between model prices (calculated using the implied binomial tree given ending nodal probabilities) and market prices of the options (e.g., Jackwerth and Rubinstein (1996)). As the number of ending nodal probabilities on an *N*-period binomial tree is *N*+1, the quadratic program has *N*+1 unknowns to solve.[1] This is a high dimensional problem if the number of periods is large (e.g., *N* = 200). Solving such a high dimensional quadratic program is potentially problematic.

In this paper, we propose a more efficient implementation method for implied binomial trees by incorporating cubic spline smoothing in the quadratic program. The essence of the problem is that the entire set of ending nodal probabilities is treated as unrelated unknowns in the standard implementation of implied binomial trees. Although this is technically correct, it is not necessary in most practical applications. A slight deviation from this standard approach can lead to a much more efficient implementation of implied binomial trees. If the risk-neutral density is a smooth function (with continuous first or higher order derivatives), which is a reasonable assumption for most applications, ending nodal probabilities are not truly independent of one another. Nearby nodal probabilities are naturally connected due to smoothness. If a particular nodal probability is high (low), nearby nodal probabilities are likely to be high (low) as well since sudden jumps in the density function are ruled out by the smoothness assumption. This local connectedness of ending nodal probabilities thus provides a convenient way to reduce the number of unknowns in the quadratic program. By treating only a subset of ending nodal probabilities (e.g., one of every four nodal probabilities) as unknowns, the dimension of the quadratic program can be reduced substantially. The remaining ending nodal probabilities can be interpolated from the selected subset of ending nodal probabilities using cubic splines. In particular, cubic spline smoothing allows us to state each ending nodal probability as a linear combination of the ending nodal probabilities in the selected subset. With this linear transformation (facilitated by cubic spline smoothing), the original quadratic program, with the entire set of ending nodal probabilities as unknowns, is transformed into a new quadratic program with only the selected subset of ending nodal probabilities as unknowns. This reduction in dimensionality can drastically improve the efficiency of the implied binomial tree method.

---

[1] The number of unknowns is actually *N*-1 since the ending nodal probabilities must sum to 1 and the binomial tree must price the underlying stock correctly. If these two conditions are treated as constraints, then the number of unknowns remains *N*+1.

This modified implementation of the implied binomial tree method will be referred to as the *smoothed implied binomial tree method* or simply the *smoothing method*.

The important question to ask then is whether or not the smoothed implied binomial tree method is also as accurate as the standard implied binomial tree method is. Cubic spline smoothing may introduce errors or noise in the set of ending nodal probabilities since only a subset of ending nodal probabilities is used to match option prices. To evaluate the numerical performance of the smoothed implied binomial tree method, we implement the implied binomial trees using both the standard and smoothing methods. The extracted risk-neutral density is then used to calculate volatility, skewness and kurtosis of the risk-neutral distribution. An extensive analysis of the numerical results shows that the smoothed implied binomial tree method is as accurate as the standard implied binomial tree method. The efficiency gains are thus achieved without any sacrifice in numerical accuracy.

Alternatively, we can compare the numerical accuracy of the implied binomial trees when they are constructed using a quadratic program of identical dimension (i.e., the same number of unknowns) with either the standard or smoothed method. This is the case, for example, if an *N*-period implied binomial tree is constructed using the standard approach while a 4*N*-period implied binomial tree is constructed using the smoothing approach with on a subset of *N* ending nodal probabilities as unknowns. Both implied binomial trees require the solution of a quadratic program with *N* unknowns. The smoothing approach is a better method if the constructed implied binomial tree is more accurate than the matching implied binomial tree constructed using the standard method. As we show subsequently, this is indeed the case.

The rest of the paper proceeds as follows. Section II describes cubic spline smoothing and how it is incorporated in the implementation of implied binomial trees. Section III evaluates the numerical performance of the smoothed implied binomial tree method in comparison with the standard implied binomial tree method. Robustness analysis is performed to ensure that the results are sufficiently general in practical applications. The final section concludes.

II. Implied Binomial Trees with Cubic Spline Smoothing

We first describe the standard method for building implied binomial trees and the difficulties associated with its implementation. Cubic spline smoothing is then introduced into

the construction of implied binomial trees in order to reduce the dimension of the quadratic program.

A. The standard method for building implied binomial trees

As explained in Rubinstein (1994) and Jackwerth (1999), the first step in the construction of an implied binomial tree is the calibration of ending *nodal probabilities* to observed option prices. These are unconditional probabilities of reaching the stock price at a particular node at option maturity. After the calibration of ending nodal probabilities, the next step is to calculate the ending *path probabilities* which are unconditional probabilities of reaching the stock price at option maturity along a particular path. Assuming path probabilities are identical across all possible paths from the initial node to the ending node (see Rubinstein's (1994) Assumption 5 on p.778), the ending path probability is simply equal to the corresponding ending nodal probability divided by the number of possible paths reaching that node. A backward recursive procedure is then used to construct the implied binomial tree (see Section IV of Rubinstein (1994) on pp.789-790).

The key step is thus the calibration of ending nodal probabilities to observed option prices. Unlike the more commonly used CRR binomial tree, the jump size of the implied binomial tree is not explicitly specified. Instead, a grid for the ending stock prices is specified first to cover the support of the risk-neutral distribution. An evenly spaced grid (either in stock price or log stock price) is typically used, with the highest and lowest stock prices on the grid sufficiently spread out in order to cover the support of the stock price distribution in the interval $(0, +\infty)$. For most applications, a 10-standard deviation spread from the current stock price on either side is usually sufficient.

For an *N*-period implied binomial tree, there are *N*+1 ending nodes and thus *N*+1 ending nodal probabilities to calibrate. As ending nodal probabilities must sum to 1 and the binomial tree must price the underlying stock correctly, *N*–1 observed option prices (across different strike prices) are required to solve for a unique set of ending nodal probabilities. Note that the price of a European option can be directly calculated from the ending stock prices ($S_j, j = 0, 1, …, N$) and ending nodal probabilities ($P_j, j = 0, 1, …, N$):

$$V^{\text{model}} = \exp(-rT)\sum_{j=0}^{N} P_j \cdot g(S_j, K)$$

where K is the option's strike price, T is its maturity, r is the risk-free rate, g is the payoff function. The observed price of the option thus provides a constraint for the ending nodal probabilities:

$$V^{\text{mkt}} = \exp(-rT)\sum_{j=0}^{N} P_j \cdot g(S_j, K)$$

With N–1 observed option prices across different strike prices, together with the other two constraints (on probabilities and the underlying stock price), a unique set of ending nodal probabilities can be solved easily from the system of N+1 linear equations. The implied binomial tree is thus quite straightforward to build if there are a sufficient number of European options available.

In most practical applications, however, we do not have a sufficient number of traded options to extract a unique set of ending nodal probabilities (for any reasonably sized binomial tree, say with $N \geq 100$ time periods). In this case, multiple sets of ending nodal probabilities can match the observed option prices correctly. The question is thus how to distinguish among these feasible sets of ending nodal probabilities and choose one that is the most appropriate. Jackwerth and Rubinstein (1996) argue that the additional degrees of freedom should be used to search for a smooth implied risk-neutral density. Specifically, they propose the use of the following optimization problem for finding the ending nodal probabilities:[2]

$$\min_{P's} \sum_{j=1}^{N-1} (P_{j-1} - 2P_j + P_{j+1})^2 + \alpha \sum_{i=1}^{m} w_i (V_i^{\text{model}} - V_i^{\text{mkt}})^2 \tag{1}$$

Subject to:

$$P_j \geq 0, \text{ for all } j \ (0 \leq j \leq N)$$

$$\sum_{j=0}^{N} P_j = 1$$

$$S(0) = \exp[-(r-q)T]\sum_{j=0}^{N} P_j S_j$$

---

[2] This setup of the quadratic program for the construction of implied binomial trees follows the approach used in Tian (2011).

where $S(0)$ and V_i^{mkt}'s are the current market prices of the underlying stock and the traded options, respectively, V_i^{model}'s are model prices of the options calculated using the implied binomial tree (given the ending nodal probabilities P_j's), T is the common maturity of all options, q is the dividend yield of the underlying stock, S_j's are the ending stock prices of the binomial tree at option maturity, w_i's are the weights for pricing errors, and α is a positive parameter specifying the penalty for not matching option prices correctly. While the first part of the objective function in Eq. (1) is inversely related to the smoothness of the risk-neutral density, the second part is a constraint that model prices match correctly with the corresponding market prices. Given the estimated ending nodal probabilities, an implied binomial tree is straightforward to construct following Rubinstein's (1984) procedure.

Note that the implied binomial tree approach described above is slightly different from the one implemented by Jackwerth and Rubinstein (1996). In their implementation, the non-negative probability constraint is initially ignored in the optimization procedure. If optimization leads to negative probabilities, an ad hoc "clamping down" approach is then used to eliminate negative probabilities. This is done by rerunning the optimization program but with a reduced support for the density function. The clamping down continues until a set of non-negative probabilities is obtained. Our setup of the optimization problem in Eq. (1) does not require such clamping down adjustment and automatically solves for non-negative probabilities. Although this is a relatively minor change in the implementation of the binomial tree method, it avoids the unnecessary use of an ad hoc clamping down approach.

B. Problems with the standard method for building implied binomial trees

The optimization problem in Eq. (1) is essentially a quadratic program and relatively simple to solve as long as the number of unknowns is not too large. Unfortunately, the number of unknowns in the quadratic program, which is equal to the number of ending nodal probabilities ($N+1$), is directly tied to the number of binomial periods (N). This means that a 100-period implied binomial tree requires solving a quadratic program with 101 unknowns. It is conceivable that the accuracy demanded by most practical applications requires the use of binomial trees with at least 100 periods, especially with American options. Such high dimensional quadratic programs may be problematic to solve. Even if it is possible to solve such high-dimensional quadratic programs, it is certainly inefficient to construct implied binomial trees this way.

Another potential problem is overfitting when too many unknowns are used in the optimization. Options data inevitably contain noise (due to bid-ask spread and other microstructure concerns) that can cause deviations of observed option prices from "true" option prices. The use of too many unknowns may lead to an increased risk of overfitting the risk-neutral density to noisy option prices.

The essence of the problem with this standard implementation is that it treats ending nodal probabilities as a set of unrelated unknowns which directly leads to the problem of large dimensions and overfitting. Of course, these nodal probabilities are not entirely unrelated if we assume that the risk-neutral density is a smooth function (i.e., with continuous first or higher derivatives). Since the objective of the quadratic program in Eq. (1) is to maximize smoothness of the risk-neutral density, it is certainly consistent with the assumption of a smooth risk-neutral density. As a result, nearby nodal probabilities are naturally connected due to smoothness. If a particular nodal probability is high (low), nearby nodal probabilities are likely high (low) as well. A sharp jump in nearby nodal probabilities is inconsistent with a smooth risk-neutral density.

The local connectedness induced by smoothness thus provides an opportunity to reduce the number of unknowns in the quadratic program in Eq. (1). By grouping nodal probabilities into subsets of nearby bundles, we can select a single nodal probability in each bundle to represent that bundle. Only the representative nodal probabilities are estimated directly from option prices. Other nodal probabilities are then inferred from the representative nodal probabilities. The total number of unknowns is no longer the total number of nodal probabilities. Instead, it is equal to the total number of bundles.

To maintain the nonparametric spirit of the implied approach, we do not impose a parametric relationship between representative nodal probabilities and the other nodal probabilities. Rather, we use a spline smoothing approach to connect nearby nodal probabilities within each bundle. In particular, cubic splines are applied to the representative nodal probabilities in order to obtain a smooth risk-neutral density function. Although other spline fitting such as quadratic or exponential splines may be used as well, we choose cubic spline fitting for its simplicity, sufficiently high order of smoothness, and popularity in volatility smile applications (e.g., Campa, Chang and Reider (1998), Bliss and Panigirtzoglou (2002), Jiang and Tian (2005), and Figlewski (2010)).

C. Cubic spline smoothing

Before discussing how to reduce the dimensions of the quadratic program using cubic splines, we briefly explain how cubic splines work. Spline smoothing is appropriate when the objective is to estimate an unknown smooth function given a set of observed values of the function. Aside from the fact that the function to be estimated is smooth, no parametric form or other analytical properties of the function is known. This is precisely the situation we face with the estimation of risk-neutral density using implied binomial trees.

Suppose that we observe the value of a smooth function $f(x)$ (with continuous second-order derivatives) at the following $n+1$ points (known as the knot points):

$$x_0 < x_1 < x_2 < \cdots < x_n$$

Denote these observed values of the function as $y_i \equiv f(x_i)$ for $0 \leq i \leq n$. In each interval $[x_{i-1}, x_i]$ for $1 \leq i \leq n$, the function $f(x)$ is approximated by a cubic function of the form:[3]

$$g_i(x) = A_i(x)y_{i-1} + B_i(x)y_i + C_i(x)y_{i-1}'' + D_i(x)y_i'' \tag{2}$$

where

$$A_i(x) = \frac{x_i - x}{x_i - x_{i-1}} \tag{2a}$$

$$B_i(x) = 1 - A_i(x) \tag{2b}$$

$$C_i(x) = \frac{1}{6}\left[A_i^3(x) - A_i(x)\right](x_i - x_{i-1})^2 \tag{2c}$$

$$D_i(x) = \frac{1}{6}\left[B_i^3(x) - B_i(x)\right](x_i - x_{i-1})^2 \tag{2d}$$

for all $x_{i-1} \leq x \leq x_i$, and y_{i-1}'' and y_i'' are the second-order derivative of the function at the two knot points x_{i-1} and x_i, respectively:

$$y_i'' = f''(x_i) \text{ and } y_{i-1}'' = f''(x_{i-1}).$$

Since a different cubic spline is used for each interval $[x_{i-1}, x_i]$ (for $1 \leq i \leq n$), we must ensure that the adjacent splines are connected properly so that the resulting piecewise cubic function is smooth across the entire support of the function. This means that we need to ensure that the left

---

[3] See Stoer and Bulirsch (1980) and Press, Teukolsky, Vetterling and Flannery (1996) for further discussions on cubic splines.

and right derivatives match up exactly at each knot point. By construction, the second-order derivatives of the adjacent cubic splines automatically match up exactly at each knot point. This is straightforward to verify from Eq. (2):

$$g''(x_i) = y_i'' \text{ and } g''(x_{i-1}) = y_{i-1}''.$$

Thus, we only need to require that the first-order derivatives of the adjacent cubic splines match up properly. This requirement leads to the following n-1 equations:

$$\frac{x_i - x_{i-1}}{6} y_{i-1}'' + \frac{x_{i+1} - x_{i-1}}{3} y_i'' + \frac{x_{i+1} - x_i}{6} y_i'' = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \tag{2e}$$

for all $1 \leq i \leq n$-1.

Given the observed values of the function at the n+1 knot points, the cubic splines are fully specified by the n+1 second-order derivatives (i.e., y''''s). Since Eq. (2e) provides n-1 restrictions on these derivatives, we still need two additional conditions to fully specify the cubic splines. As the density function is most likely diminishing to zero at the two tails, we simply set the second-order derivative to zero at the two extreme points:

$$y_0'' = y_n'' = 0 \tag{2f}$$

Combining with the n-1 equations in (2e), we obtain the full set of second-order derivatives at the n+1 knot points in terms of the observed function values. These second-order derivatives in turn completely specify the cubic spline function in Eq. (2).

D. The smoothing method for building implied binomial trees

We now reconsider the calibration of an N-period binomial tree to a set of m observed option prices. The key step of this calibration is again the estimation of the risk-neutral density from the observed option prices. In the binomial tree setting, this means that we need to estimate the N+1 ending nodal probabilities. Without any restrictions, this leads to a quadratic program with N+1 unknowns and the estimation can be either difficult, inefficient or both. It is thus sensible to reduce the number of unknowns and solve an equivalent quadratic program with a lower dimension. In this paper, we use cubic spline smoothing to accomplish this task. Only a subset of the nodal probabilities is treated as unknowns to be estimated while the remainder is interpolated from this subset using cubic splines.

To incorporate cubic splines into the estimation of the risk-neutral density, we first specify a finite interval $[S_{\min}, S_{\max}]$ as the support of the risk-neutral density where $S_{\min}$ and $S_{\max}$ are the lowest and highest stock price that can be reached at option maturity. In an N-period binomial tree, there are $N+1$ ending nodes after N periods:

$$S_0 = S_{\min} < S_1 < S_2 < \ldots < S_N = S_{\max}.$$

This stock price grid is designed to span the entire support of the risk-neutral distribution. The stock price grid can be equally spaced between $S_{\min}$ and $S_{\max}$ or specified otherwise as required by the application. These stock prices specify the possible ending stock prices of the binomial tree after N periods (at the option maturity). Prior to maturity, a recursive procedure is used (e.g., Rubinstein (1984)) to determine the possible stock prices and conditional binomial probabilities in each period using the ending stock prices and ending nodal probabilities.

Next, we introduce a bandwidth parameter h (≥ 1) that plays a similar role that the width of bins plays in histograms. The ending nodes are divided into n bundles with h nodes in each bundle, $N = n \cdot h$. Only one nodal probability in each bundle is treated as unknown while others are indirectly specified using cubic splines. The nodes selected for the unknown nodal probabilities are spread out evenly across the vertical span of the binomial tree:

$$S_0 = S_{\min}, S_h, S_{2h}, \ldots, S_{nh} = S_{\max}$$

These nodes will be referred to as the *knot points* or *knot nodes*. The standard implied binomial tree is a special case when there is only one nodal probability in each bundle (i.e., $h = 1$).

Between two adjacent knot nodes, $S_{(i-1)h}$ and S_{ih}, the nodal probability at node S_j (for all $(i-1)h \leq j \leq ih$) is interpolated from the nodal probabilities at the two knot nodes using cubic splines. Applying the cubic spline specification in Eq. (2), these nodal probabilities are determined as follows:

$$P_j = A_i(S_j)P_{(i-1)h} + B_i(S_j)P_{ih} + C_i(S_j)P^{''}_{(i-1)h} + D_i(S_j)P^{''}_{ih} \tag{3}$$

where

$$A_i(S_j) = \frac{S_{ih} - S_j}{S_{ih} - S_{(i-1)h}} \tag{3a}$$

$$B_i(S_j) = 1 - A_i(S_j) \tag{3b}$$

$$C_i(S_j) = \frac{1}{6}\left[A_i^3(S_j) - A_i(S_j)\right]\left(S_{ih} - S_{(i-1)h}\right)^2 \tag{3c}$$

$$D_i(S_j) = \frac{1}{6}\left[B_i^3(S_j) - B_i(S_j)\right]\left(S_{ih} - S_{(i-1)h}\right)^2 \tag{3d}$$

for all $(i-1)h \leq j \leq ih$, and $P_{(i-1)h}^{''}$ and $P_{ih}^{''}$ are the second-order derivative of the cubic spline function at the two knot nodes, $S_{(i-1)h}$ and S_{ih}, respectively. To ensure the splines are connected properly (i.e., with continuous second-order derivatives), the following condition must be satisfied by any two adjacent splines:

$$\frac{S_{ih} - S_{(i-1)h}}{6} P_{(i-1)h}^{''} + \frac{S_{(i+1)h} - S_{(i-1)h}}{3} P_{ih}^{''} + \frac{S_{(i+1)h} - S_{ih}}{6} P_{(i+1)h}^{''} = \frac{P_{(i+1)h} - P_{ih}}{S_{(i+1)h} - S_{ih}} - \frac{P_{ih} - P_{(i-1)h}}{S_{ih} - S_{(i-1)h}} \tag{3e}$$

for all $1 \leq i \leq n$-1. Together with the simplifying assumption that the second-order derivative of the cubic spline function diminishes at both the lowest and highest stock price nodes (i.e., $P_0^{''} = P_n^{''} = 0$), the second-order derivatives at all the knot nodes are fully specified by the n-1 linear equations in (3e) in terms of nodal probabilities at the knot nodes. Write down this linear relationship as:

$$P_{ih}^{''} = \sum_{k=0}^{n} u_{i,k} P_{kh}, \quad \text{for all } 0 \leq i \leq n \tag{4}$$

where the $u_{i,k}$'s are the elements of the following matrix:

$$\begin{bmatrix} 0, 0, \cdots, 0 \\ (\Sigma_0)^{-1} \Sigma_1 \\ 0, 0, \cdots, 0 \end{bmatrix}_{(n+1)\times(n+1)}$$

and

$$\Sigma_0 = \begin{pmatrix} \dfrac{\Delta S_{2h} + \Delta S_h}{3} & \dfrac{\Delta S_{2h}}{6} & 0 & \cdots & \cdots & \cdots & 0 \\ \dfrac{\Delta S_{2h}}{6} & \dfrac{\Delta S_{2h} + \Delta S_{3h}}{3} & \dfrac{\Delta S_{3h}}{6} & 0 & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & \dfrac{\Delta S_{(n-2)h}}{6} & \dfrac{\Delta S_{(n-2)h} + \Delta S_{(n-1)h}}{3} & \dfrac{\Delta S_{(n-1)h}}{6} \\ 0 & \cdots & \cdots & \cdots & 0 & \dfrac{\Delta S_{(n-1)h}}{6} & \dfrac{\Delta S_{(n-1)h} + \Delta S_{nh}}{3} \end{pmatrix}$$

$$\Sigma_1 = \begin{pmatrix} \dfrac{1}{\Delta S_h} & -\dfrac{1}{\Delta S_h} - \dfrac{1}{\Delta S_{2h}} & \dfrac{1}{\Delta S_{2h}} & 0 & \cdots & \cdots & 0 \\ 0 & \dfrac{1}{\Delta S_{2h}} & -\dfrac{1}{\Delta S_{2h}} - \dfrac{1}{\Delta S_{3h}} & \dfrac{1}{\Delta S_{3h}} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \dfrac{1}{\Delta S_{(n-2)h}} & -\dfrac{1}{\Delta S_{(n-2)h}} - \dfrac{1}{\Delta S_{(n-1)h}} & \dfrac{1}{\Delta S_{(n-1)h}} & 0 \\ 0 & \cdots & \cdots & 0 & \dfrac{1}{\Delta S_{(n-1)h}} & -\dfrac{1}{\Delta S_{(n-1)h}} - \dfrac{1}{\Delta S_{nh}} & \dfrac{1}{\Delta S_{nh}} \end{pmatrix}$$

$$\Delta S_{ih} = S_{ih} - S_{(i-1)h}$$

Given these second-order derivatives at the knot points, the cubic spline specified in Eq. (3) is now stated without any explicit reference to the second-order derivatives:

$$P_j = A_i(S_j)P_{(i-1)h} + B_i(S_j)P_{ih} + C_i(S_j)\sum_{k=0}^{n} u_{i-1,k}P_{kh} + D_i(S_j)\sum_{k=0}^{n} u_{i,k}P_{kh} \qquad (5)$$

for all $0 \le j \le N$ and i is the smallest integer not exceeding j/h (i.e., $(i\text{-}1)h < j \le ih$). We then restate the quadratic program in Eq. (1) as a problem for nodal probabilities at the knot points only:

$$\min_{P_{jh}\text{'s}} \sum_{j=1}^{N-1}(P_{j-1} - 2P_j + P_{j+1})^2 + \alpha\sum_{i=1}^{m} w_i(V_i^{\text{model}} - V_i^{\text{mkt}})^2 \qquad (6)$$

Subject to:

$$P_j \ge 0, \text{ for all } j \; (0 \le j \le N)$$

$$\sum_{j=0}^{N} P_j = 1$$

$$S(0) = \exp[-(r-q)T]\sum_{j=0}^{N} P_j S_j$$

where

$$P_j = A_i(S_j)P_{(i-1)h} + B_i(S_j)P_{ih} + C_i(S_j)\sum_{k=0}^{n} u_{i-1,k}P_{kh} + D_i(S_j)\sum_{k=0}^{n} u_{i,k}P_{kh}$$

for all $0 \le j \le N$ and i is the smallest integer not exceeding j/h (i.e., $(i\text{-}1)h < j \le ih$).

Although the quadratic program in Eq. (6) appears to be similar to the one in Eq. (1), only $n+1$ of the total $nh+1$ nodal probabilities are treated as unknowns. In contrast, all $nh+1$ nodal

probabilities are treated as unknowns in Eq. (1). The dimension of the quadratic program is thus reduced by a factor of h ($\geq$ 1). Depending on the choice of this bandwidth parameter h, the potential gains in efficiency can be substantial. Consider, for example, a 100-period implied binomial tree (i.e., N = 100). With the standard setup of the quadratic program in Eq. (1), we need to solve 101 unknown nodal probabilities in order to calibrate the binomial tree to option prices. Such a quadratic program requires the use of a 101×101 matrix. With a moderate choice of the bandwidth parameter of h = 4, we only need to solve 26 unknown nodal probabilities. The dimension of the matrix used in the quadratic program is reduced to 26×26. The potential improvement in efficiency is thus very clear.

III. Numerical Performance

Improved implementation and numerical efficiency may not be meaningful unless they are also accompanied by a maintained or improved level of numerical accuracy. We now perform numerical analysis to evaluate the numerical accuracy of the smoothed implied binomial tree method in comparison with the standard implied binomial tree method. Option prices are first calculated using a given option pricing model for a discrete set of strike prices (with identical maturity). The smoothed implied binomial tree is then calibrated to these option prices. The risk-neutral density and its moments (e.g., volatility, skewness and kurtosis) are then extracted from the fitted binomial tree and compared to the true risk-neutral density and its moments of the underlying stock price distribution. Numerical accuracy is evaluated on the basis of how closely these extracted measures match those of the corresponding accurate measures of the underlying stock price distribution. A similar analysis is then carried out for the standard implied binomial tree method. A comparison of the numerical performance of the two implied binomial tree methods provides insight on the relative numerical accuracy of these methods.

To perform the required numerical analysis, we assume that the dynamics of the underlying stock price are specified by the following stochastic volatility with jumps (SVJ) model (under the risk-neutral probability measure):

$$\frac{dS(t)}{S(t)} = \left(r - \lambda\mu_J\right)dt + \sqrt{V(t)}dz_S(t) + J(t)dq(t) \tag{7}$$

$$dV(t) = [\theta_v - k_v V(t)]dt + \sigma_v\sqrt{V(t)}dz_v(t) \tag{8}$$

13

where t is the time index, $S(t)$ is the stock price, $V(t)$ is the diffusion component of return variance, $z_s(t)$ and $z_v(t)$ are correlated standard Brownian motions:

$$Cov[dz_S(t), dz_v(t)] = \rho dt,$$

$J(t)$ is the percentage jump size that is lognormally, identically and independently distributed over time with unconditional mean μ_J and standard deviation σ_J:

$$\ln[1 + J(t)] \sim N(\ln(1 + \mu_J) - \frac{1}{2}\sigma_J^2, \sigma_J^2)$$

$q(t)$ is a Poisson jump counter with intensity λ (i.e., frequency of jumps per year):

$$\Pr\{dq(t) = 1\} = \lambda dt,$$

and r is the risk-free rate. Jump frequency and size are uncorrelated with each other or with the two diffusion components. We select a base set of model parameters for the SVJ model based on empirical estimates from previous studies (e.g., Bakshi, Cao, and Chen (1997)): $k_v = 1$, $\sigma_v = 0.25$, $\rho = 0$, $\lambda = 0.5$, $\mu_J = -0.15\sigma_I$, $\sigma_J = 0.15\sigma_I$, $\theta_v = 0.85k_v\sigma_I^2$, $V_0 = \theta_v/k_v$ and $\sigma_I = 0.2$ is the annualized integrated volatility. Both the risk-free rate and dividend yield are assumed to be zero. With this base set of parameters, the diffusion (jump) portion of the volatility accounts for 85% (15%) of the total volatility. We do perform robustness tests subsequently using alternative sets of model parameters that deviate from the base set of parameters and the results are not materially different.

A. Measures of numerical accuracy

If the constructed implied binomial tree provides an accurate estimate of the risk-neutral density, it should also provide accurate estimates of option prices. We thus focus on the accuracy of the estimated risk-neutral density when we evaluate the numerical performance of an implied binomial tree method. In particular, we evaluate whether or not the constructed implied binomial tree provides accurate estimate of volatility, skewness and kurtosis of the risk-neutral density. These three moments capture key aspects of the risk-neutral density and are widely used in empirical applications such as volatility forecasting, variance risk premium, empirical asset pricing, and portfolio selection (e.g., Jiang and Tian (2005), Ang, Hodrick, Xing, and Zhang (2006), Conrad, Dittmar, and Ghysels (2008), Bollerslev, Tauchen, and Zhou (2009), Carr and Wu (2009), Chang, Christoffersen and Jacobs (2009), DeMiguel, Plyakha, Uppal and Vilkov (2009), and Feunou, Fontaine and Tedongap (2009)).

There are two commonly used methods for calculating these higher moments. The first one is based on numerical integration of option prices (NIP). As derived by Bakshi, Kapadia and Madan (2003), the volatility (i.e., standard deviation), skewness and kurtosis of the risk-neutral density are given by the following expressions, respectively:

$$\sigma = \sqrt{e^{rT}H_2 - e^{2rT}H_1^2} \tag{9}$$

$$\text{skew} = \frac{e^{rT}H_3 - 3e^{2rT}H_1H_2 + 2e^{3rT}H_1^3}{(e^{rT}H_2 - e^{2rT}H_1^2)^{3/2}} \tag{10}$$

$$\text{kurt} = \frac{e^{rT}H_4 - 4e^{2rT}H_1H_3 + 6e^{3rT}H_1^2H_2 - 3e^{4rT}H_1^4}{(e^{rT}H_2 - e^{2rT}H_1^2)^2} \tag{11}$$

where

$$H_1 = -e^{-rT} + (S_0/S^*)e^{-qT} - \int_{S^*}^{\infty} \frac{V_{call}(K)}{K^2} dK - \int_0^{S^*} \frac{V_{put}(K)}{K^2} dK$$

$$H_2 = \int_{S^*}^{\infty} \frac{2[1 - \ln(K/S^*)]}{K^2} V_{call}(K)dK - \int_0^{S^*} \frac{2[1 - \ln(K/S^*)]}{K^2} V_{put}(K)dK$$

$$H_3 = \int_{S^*}^{\infty} \frac{3\ln(K/S^*)[2 - \ln(K/S^*)]}{K^2} V_{call}(K)dK - \int_0^{S^*} \frac{3\ln(K/S^*)[2 - \ln(K/S^*)]}{K^2} V_{put}(K)dK$$

$$H_4 = \int_{S^*}^{\infty} \frac{4\ln^2(K/S^*)[3 - \ln(K/S^*)]}{K^2} V_{call}(K)dK - \int_0^{S^*} \frac{4\ln^2(K/S^*)[3 - \ln(K/S^*)]}{K^2} V_{put}(K)dK$$

V_{call} and V_{put} are call and put prices, K and T are option strike price and maturity, respectively, and S^* is an arbitrary level of the stock price which is usually chosen as the current stock price or forward stock price. The required option prices can be either calculated using a specified option pricing model or observed directly in the market. For this approach to be accurate, we need prices of options with strike prices across the entire support of the risk-neutral density at a reasonably small increment. Numerical integration may not work well if the increment in stock price is too large or the available strike prices do not cover a sufficient range of the support of the risk-neutral density.

Another approach is to calculate the higher moments from numerical integration of the risk-neutral density (NID). Given the risk-neutral density (either theoretically or estimated), the j-th order non-central moments can be calculated directly as integrals of the risk-neutral density:

$$m_j = \int_{-\infty}^{\infty} x^j f(x)dx, \quad \text{for } j = 1, 2, 3, ... \tag{12}$$

15

With these moments, the volatility, skewness and kurtosis of the risk-neutral density are straightforward to calculate:

$$\sigma = \sqrt{m_2 - m_1^2} \tag{13}$$

$$\text{skew} = \frac{m_3 - 3m_1 m_2 + 2m_1^3}{(m_2 - m_1^2)^{3/2}} \tag{14}$$

$$\text{kurt} = \frac{m_4 - 4m_1 m_3 + 6m_1^2 m_2 - 3m_1^4}{(m_2 - m_1^2)^2} \tag{15}$$

Theoretically, these two methods are equivalent since the risk-neutral density determines a unique price for an option. Conversely, a continuum of option prices across all strike prices determines a unique risk-neutral density function. The relevant question is then how these methods actually perform when they are used to estimate the higher moments. To evaluate their numerical performance, we apply these two methods to the SVJ model with the base set of parameter values described above. The initial stock price is normalized at 100. Option maturities vary from 0.5 to 12 months. The support of the risk-neutral distribution is approximated by an interval [S_{min}, S_{max}] covering a 6-standard deviation spread from the current stock price on either side. The support is then divided into 600 subintervals to facilitate numerical integration. For integration over option prices, we calculate 601 option prices at the end point of each subinterval using the SVJ model. For integration over the density function, we calculate the 601 values of the density function directly from the SVJ model. The results are reported in Table 1. As a benchmark for comparison purposes, we also report the higher moments calculated from the moment generating function (MGF) of the risk-neutral distribution.

Table 1 about here

As reported in Table 1, the two numerical integration methods provide virtually identical values for all three moments (i.e., volatility, skewness and kurtosis) across all maturities. These values are also virtually identical to those calculated directly from the moment generating function. The differences are so small that it is barely noticeable. We also perform robustness analysis using alternative sets of model parameter values. Untabulated results show that both methods remain very accurate and there are virtually no differences at all between moments

16

calculated from these methods. We thus conclude that both methods are very accurate and either one may be used to calculate moments of the risk-neutral density. The choice between the two is essentially a matter of convenience. If option prices are already available, numerical integration of option prices (NIP) is a better choice. Using the density-based approach requires a further step of estimating the risk-neutral density from option prices. On the other hand, numerical integration of risk-neutral density (NID) is a better choice if the risk-neutral density is already known or estimated. In our applications below, the construction of the implied binomial tree automatically provides an estimate of the risk-neutral density function. We thus will use the NID method to calculate the risk-neutral moments.

B. Numerical performance of implied binomial trees

To evaluate the numerical performance of implied binomial trees, we calculate a set of option prices using the SVJ model with the base set of parameter values as described above. The initial stock price is normalized at 100. Option maturity varies from 0.5 to 12 months. A set of strike prices is chosen to cover an interval of 3 standard deviations from either side of the initial stock price. The strike price interval is then divided into 20 subperiods, with 21 equally spaced strike prices selected to cover the end points of the subintervals.

An *N*-period implied binomial tree is then constructed to match these SVJ option prices. For the standard implied binomial tree (BIN) approach, we use $N = 120$ periods and the penalty parameter α for pricing errors is equal to 1.[4] For comparison purposes, we also use $N = 120$ periods for our cubic spline smoothed implied binomial tree (csBIN) method. To improve the efficiency of the implied binomial tree implementation, we vary the bandwidth parameter (*h*) from 1 to 4. For $h = 1$, the smoothed implied binomial tree method is identical to the standard implied binomial tree method. For $h > 1$, the number of unknowns in the quadratic program is reduced by a factor of *h*, thus making the implied binomial tree method more efficient. For example, only 30 unknowns are solved in the quadratic program if $h = 4$ in comparison to 120 unknowns for the standard implied binomial tree method. More importantly for efficiency gains,

---

[4] Note that our choice of the penalty parameter (α) value is quite different from those in Jackwerth and Rubinstein (1996). This is because our penalty parameter is applied to the weighted average squared pricing errors instead of the sum of squared pricing errors. Applying the penalty parameter this way minimizes the impact of the number of options used in the construction of the implied binomial tree. Otherwise, a larger (smaller) α value may have to be used when the number of options is smaller (greater).

we no longer need to solve a quadratic program with a 120×120 matrix. Instead, we only need to solve one with a 30×30 matrix.

Solving the quadratic program (either Eq. (1) or (6)) provides the ending nodal probabilities, P_j's, for the implied binomial tree. These ending nodal probabilities are then converted into values of the risk-neutral density observed at the ending stock prices:

$$p(R_j) = \frac{P_j}{\Delta R_j}, \quad \text{for } 0 \le j \le N,$$ (16)

where $R_j = \ln[S_j/S(0)]$ is continuously compounded stock return, and $\Delta R_j = (R_{j+1} - R_{j-1})/2$ is the subinterval covered by the nodal probability. The density function is then used to calculate the risk neutral moments using numerical integration of the density function (NID) as discussed before. The results are reported in Table 2 for both implementation methods for implied binomial trees.

<div style="border:1px solid black; display:inline-block; padding:10px;">Table 2 about here</div>

Consider first the results from the standard implied binomial tree method. These are reported in Table 2 in the column labeled "csBIN" with bandwidth parameter $h = 1$. The standard implied binomial tree is a special case of the smoothed implied binomial tree when the bandwidth parameter (h) is equal to 1. We thus do not separately report the results for the standard implied binomial tree method. Comparing with the accurate values calculated using the moment generating function (MGF), the risk-neutral moments estimated using the standard implied binomial tree method are virtually identical in every case. This is true for volatility, skewness and kurtosis across all maturities. These results suggest that the implied binomial tree method does a very good job extracting risk-neutral moments from option prices.

More importantly, the numerical accuracy of the smoothed implied binomial tree method (csBIN) is virtually identical to that of the standard implied binomial tree method. In fact, as the bandwidth parameter h increases from 1 to 4, there is no virtually no change at all in the estimated value of the risk-neutral volatility or skewness. As expected, we do observe a slight increase in estimation error for the fourth moment, the risk-neutral kurtosis, especially for longer maturities. Even in that case, the difference is still so small and clearly immaterial for any practical applications. The smoothed implied binomial tree method thus maintains virtually the

same level of numerical accuracy as the standard implied binomial tree and yet provides a substantial improvement in computational efficiency. With bandwidth parameter h as high as 4, the smoothed method still achieves the same level of numerical accuracy by replacing a quadratic program with a 120×120 matrix with a much lower dimensional quadratic program with a 30×30 matrix. The improvement in efficiency is thus quite clear.

Of course, the moment measures may not be sufficient in gauging the numerical accuracy of the implied binomial tree method since not all features of the risk-neutral distribution are captured by these moments. We thus plot the entire risk-neutral density estimated using the implied binomial tree method, in comparison with the true density from the SVJ model. The results are presented in Figure 1 for the case with a 3-month maturity and a bandwidth parameter of 4. The results for other cases are similar and thus omitted. As shown in Figure 1, the density plot is virtually indistinguishable between the implied binomial tree method and the true SVJ density. This is true for both the standard and the smoothed implied binomial tree methods. These results provide further support for the numerical accuracy of the implied binomial tree method in extracting the risk-neutral density from option prices.

<div style="border:1px solid black; padding:10px; text-align:center;">
Figure 1 about here
</div>

To ensure that the results reported in Table 2 and Figure 1 are not confined to that specific setting, we perform several robustness tests. First, a commonly encountered problem in practice is the lack of available strike prices to span the support of the risk-neutral distribution. In the analysis so far, we assume that we have prices of options across 21 strike prices that collectively cover 3 standard deviations on either side of the current stock price. How does the implied binomial tree method perform if there are fewer (e.g., only 11) strike prices? With fewer strike prices available, the gap between two adjacent strike prices becomes larger and may thus affect the numerical accuracy of the implied binomial tree method. With a reduced number of strike prices, we recalibrate the implied binomial trees and then recalculate the three risk-neutral moments using the extracted risk-neutral density. The new results are reported in Table 3, with the number of strike prices ranging from 7 to 21. Option maturity is fixed at 3 months. The results for other maturities are similar and thus omitted for brevity. Although the estimate values of the risk-neutral moments deviate a bit more from the accurate values of the corresponding

moments, the differences are still too small to have any material impact on practical applications. Even when there are only 7 or 9 strike prices available, the implied binomial tree method remains quite accurate.

<div style="border:1px solid black; display:inline-block; padding:10px 40px;">Table 3 about here</div>

Next, the strike prices may not be evenly spaced over a given range of stock prices. More realistically, there are usually more strike prices around the at-the-money region and fewer strike prices away from the at-the-money region. In other words, the gap between two adjacent strike prices tends to increase as we move away from the at-the-money strike price (on either side). To see if the implied binomial tree method still works well in this case, we let the gap between two adjacent strike price increases by 25% each time we move away from the at-the-money region by one strike price. This is roughly how strike prices on the S&P 500 index options are spaced. We still have 21 strike prices available, covering 3 standard deviations around either side of the current stock price. The difference is that these strike prices are no longer evenly spaced. We recalibrate the implied binomial trees using the new set of strike prices and reproduce the moment results in Table 2. The new results are reported in Table 4.

<div style="border:1px solid black; display:inline-block; padding:10px 40px;">Table 4 about here</div>

As shown in Table 4, both implied binomial tree methods continue to work well when the strike prices are spread out unevenly, with decreased availability in strike prices as we move away from the at-the-money strike price in either direction. The numerical accuracy is comparable to the level reported in Table 2 where the strike prices are evenly spread over the same span in stock price. This is particularly true for volatility and skewness. For kurtosis, numerical accuracy is slightly worse than the level reported in Table 2. The errors are nonetheless quite small and are most likely immaterial in any practical applications.

Finally, we consider the robustness of numerical performance in case of greater volatility and higher jump risk. All results reported so far are obtained for option prices from an SVJ model with a 20% total volatility and with diffusion and jumps accounting for 85% and 15% of the total volatility, respectively. With such moderate levels of total volatility and jumps, the risk-

neutral density has fairly low levels of skewness and kurtosis (as reported in Table 2). To further examine robustness, we elevate the level of total volatility to 30% and let jumps (diffusion) to account for 25% (75%) of the total volatility. To do so, we specify the parameters of the SVJ model as follows: $k_v = 1$, $\sigma_v = 0.25$, $\rho = 0$, $\lambda = 0.25$, $\mu_J = -0.15\sigma_I$, $\sigma_J = 0.25\sigma_I$, $\theta_v = 0.75k_v\sigma_I^2$, $V_0 = \theta_v/k_v$ and $\sigma_I = 0.3$. Both skewness and kurtosis are now much greater (as reported subsequently in Table 5). Given these new parameters of the SVJ model, we then recalibrate the implied binomial tree to option prices for each maturity and bandwidth combination used in Table 2. The extracted risk-neutral density is then used to calculate the three moments, with the results reported in Table 5.[5]

<div style="border:1px solid black; padding:10px; text-align:center; width:50%; margin:auto;">
Table 5 about here
</div>

As shown in Table 5, the elevated levels of volatility and jumps indeed lead to higher levels of skewness and kurtosis. As expected, numerical accuracy is not as precise as that is reported in Table 2 when the base set of parameters of the SVJ model is used. For risk-neutral volatility, both the standard and smoothed implied binomial tree methods remain very accurate with numerical errors virtually zero in every case. The elevated levels of total volatility and jumps do not have any impact on the numerical accuracy of the estimated volatility at all. This is no longer the case for skewness and kurtosis. The errors are clearly larger than those reported in Table 2. For skewness, the largest percentage errors are found in the case of options with 0.5 month to maturity, ranging from 5.1% to 5.7%. In comparison, the largest error in the estimated skewness reported in Table 2 is only 3.8% (in the case of options with 6 months to maturity and a bandwidth parameter of 4). The numerical accuracy for kurtosis deteriorates even more. The largest errors are also found in the case of options with 0.5 month to maturity. The percentage errors in the estimated kurtosis range from 8.2% to 9.0%. In comparison, the largest corresponding error reported in Table 2 is only 0.4%. Numerical accuracy is clearly impacted by the higher levels of total volatility and jumps. Nevertheless, the estimation errors are still relatively small and will probably not have any material impact in practical applications.

---

[5] Because of the increased level of jump risk (25% as opposed to 15% of total volatility), the tails of the risk-neutral density become more important. We thus increase the coverage of the strike prices to 4 standard deviations on either side of the current stock price (instead of the 3 standard deviations used before).

IV. Conclusions

In this paper, we have developed a more efficient method for implementing implied binomial trees. A key innovation in our approach is the use of cubic spline smoothing for reducing the dimensionality of the quadratic program. Instead of treating all ending nodal probabilities as unrelated unknowns, only a selected subset of ending nodal probabilities is treated as such. The remaining ending nodal probabilities are interpolated from the selected subset using natural cubic splines. This allows for the transformation of the original quadratic program, with all ending nodal probabilities as unknowns, into a lower dimensional quadratic program with only the selected subset of ending nodal probabilities as unknowns. The reduction of dimensionality substantially improves the numerical efficiency of the implied binomial tree.

More importantly, the efficiency gains are achieved without any sacrifice in numerical accuracy. An extensive numerical analysis demonstrates that the smoothed implied binomial tree method is as accurate as the standard implied binomial tree method as long as the bandwidth parameter is not too large (e.g., $h \leq 4$).[6] Both methods are designed to extract a smooth risk-neutral density from option prices. The extracted risk-neutral density also provides very accurate estimates for the risk-neutral volatility, skewness and kurtosis.

The efficiency gains of the smoothed implied binomial tree method can be very helpful in practical applications. Faster implementation is an obvious advantage. A more important advantage is the ability to handle very large binomial trees. The standard implied binomial tree method runs into difficulties when the implied binomial tree is constructed for a large number of periods (e.g., $N = 400$). With so many unknowns, an optimization routine such as a quadratic program is most likely difficult, if not impossible, to implement. This can be a problem if the application demands the use of a large binomial tree due to the level of accuracy required. The smoothed implied binomial tree method provides a simple solution to this problem.

---

[6] As the selected subset of ending nodal probabilities shrinks in size (or equivalently, as the bandwidth parameter h increases), the numerical accuracy of the smoothed implied binomial tree method may deteriorate eventually. For most practical applications, however, a bandwidth of 4 is usually sufficient. The tradeoff between numerical efficiency and accuracy is thus quite favorable for the smoothed implied binomial tree method in most cases.

References

Ang, Andrew, Robert J. Hodrick, Yuhang Xing, and Xiaoyan Zhang, 2006, "The cross-section of volatility and expected returns," *Journal of Finance* 61, 259-299.

Bakshi, Gurdip, Charles Cao, and Zhiwu Chen, 1997, "Empirical performance of alternative option pricing models," *Journal of Finance* 52, 2003-2049.

Bakshi, Gurdip, Nikunj Kapadia, and Dilip Madan, 2003, "Stock return characteristics, skew laws, and the differential pricing of individual equity options," *Review of Financial Studies* 16, 101-143.

Black, Fischer, and Myron Scholes, 1973, "The pricing of options and corporate liabilities," *Journal of Political Economy* 81, 637-659.

Bliss, Robert R., and Nikolaos Panigirtzoglou, 2002, "Testing the stability of implied probability density functions," *Journal of Banking and Finance* 26, 381-422.

Bollerslev, Tim, George Tauchen, and Hao Zhou, 2009, "Expected stock returns and variance risk premia," *Review of Financial Studies* 22, 4463-4492.

Campa, Jose M., P.H. Kevin Chang, and Robert L. Reider, 1998, "Implied exchange rate distributions: Evidence from OTC option markets," *Journal of International Money and Finance* 17, 117-160.

Carr, Peter, and Liuren Wu, 2009, "Variance Risk Premia," *Review of Financial Studies* 22, 1311-1341.

Chang, Bo Young, Peter Christoffersen, and Kris Jacobs, 2009, "Market skewness risk and the cross-section of stock returns," Working, *McGill University*.

Conrad, Jennifer, Robert F. Dittmar, and Eric Ghysels, 2008, "Ex-ante skewness and expected stock returns," Working paper, *University of North Carolina*, Chapel Hill.

Cox, John, Stephen Ross, and Mark Rubinstein, 1979, "Option pricing: A simplified approach," *Journal of Financial Economics* 7, 229-263.

DeMiguel, Victor, Yuliya Plyakha, Raman Uppal, and Grigory Vilkov, 2009, "Improving portfolio selection using option-implied volatility and skewness," Working paper, *London Business School*.

Feunou, Bruno, Jean-Sébastien Fontaine, and Roméo Tedongap, 2009, "The equity premium and the volatility spread: The role of risk-neutral skewness," Working paper, *Duke University*.

Figlewski, Stephen, 2010, "Estimating the implied risk neutral density for the U.S. market portfolio." In: *Volatility and Time Series Econometrics: Essays in Honor of Robert F. Engle*, eds. Tim Bollerslev, Jeffrey R. Russell and Mark Watson, Oxford, UK: Oxford University Press.

Jackwerth, Jens C., 1999, "Option implied risk-neutral distribution and implied binomial trees: A literature review," *Journal of Derivatives* 7, 66-82.

Jackwerth, Jens C., and Mark Rubinstein, 1996, "Recovering probability distributions from option prices," *Journal of Finance* 51, 1611-1631.

Jiang, George J., and Yisong S. Tian, 2005, "Model-free implied volatility and its information content," *Review of Financial Studies* 18, 1305-1342.

Merton, Robert C., 1973, "Theory of rational option pricing," *Bell Journal of Economics and Management Science* 4, 141-183.

Press, William H., Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery, 1996, *Numerical Recipes in Fortran 77: The Art of Scientific Computing*, 2nd Ed., Cambridge, U.K.: Cambridge University Press.

Rubinstein, Mark, 1994, "Implied binomial trees," *Journal of Finance* 49, 771-818.

Stoer, J., and R. Bulirsch, 1980, *Introduction to Numerical Analysis*, New York: Springer-Verlag.

Tian, Yisong S., 2011, "Extracting risk-neutral density and its moments from American option prices," *Journal of Derivatives* 18(3), 17-34.

Table 1

Risk-neutral moments by numerical integration

This table evaluates the numerical performance of two alternative methods for risk-neutral volatility, skewness and kurtosis calculations – the numerical integration of option prices (NIP) method based on numerical integration of option prices (e.g., Bakshi, Kapadia and Madan (2003)) and the numerical integration of risk-neutral density (NID) method based on numerical integration of the risk-neutral density function. The moment generating function (MGF) is also used to calculate a benchmark value for each higher moment for evaluation purposes. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs. (7) and (8) with the following parameters: $k_v = 1$, $\sigma_v = 0.25$, $\rho = 0$, $\lambda = 0.5$, $\mu_J = -0.15\sigma_I$, $\sigma_J = 0.15\sigma_I$, $\theta_v = 0.85k_v\sigma_I^2$, $V_0 = \theta_v/k_v$ and $\sigma_I = 0.2$ is the annualized integrated volatility. The risk-free rate is assumed to be zero for convenience. With this base set of parameters, the diffusion (jump) portion of the volatility accounts for 15% (85%) of the total volatility.

Maturity	Volatility			Skewness			Kurtosis		
(month)	NIP	NID	MGF	NIP	NID	MGF	NIP	NID	MGF
0.5	0.038	0.038	0.038	-0.044	-0.044	-0.044	3.170	3.126	3.154
1	0.054	0.054	0.054	-0.034	-0.034	-0.034	3.178	3.179	3.178
2	0.076	0.076	0.076	-0.031	-0.031	-0.031	3.277	3.277	3.278
3	0.094	0.094	0.094	-0.034	-0.034	-0.034	3.376	3.376	3.377
6	0.132	0.132	0.132	-0.053	-0.052	-0.052	3.625	3.614	3.617
12	0.187	0.187	0.187	-0.091	-0.090	-0.091	3.889	3.877	3.888

Table 2
Numerical performance of implied binomial trees

This table evaluates the numerical performance of two alternative methods for constructing implied binomial tree – the standard implied binomial tree (BIN) method and the cubic spline smoothed implied binomial tree (csBIN) method. All binomial trees are constructed with 120 periods, matched to 21 option prices with strike prices equally spaced over an interval covering 3 standard deviations on either side the initial stock price. The risk-neutral density from the implied binomial tree is then used to calculate volatility, skewness and kurtosis of the risk-neutral density. For comparison purposes, the corresponding accurate moments are calculated using the moment generating function (MGF) of the theoretical density function of the assumed SVJ model. Estimates from the standard implied binomial tree method are not separately reported in the table since they are identical the csBIN estimates when the bandwidth parameter h is equal to 1. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs. (7) and (8) with the same set of parameters as in Table 1.

Maturity (month)	Bandwidth (h)	Volatility		Skewness		Kurtosis	
		csBIN	MGF	csBIN	MGF	csBIN	MGF
0.5	1	0.038	0.038	-0.045	-0.044	3.158	3.154
0.5	2	0.038	0.038	-0.045	-0.044	3.158	3.154
0.5	3	0.038	0.038	-0.045	-0.044	3.158	3.154
0.5	4	0.038	0.038	-0.045	-0.044	3.158	3.154
1	1	0.054	0.054	-0.034	-0.034	3.177	3.178
1	2	0.054	0.054	-0.034	-0.034	3.177	3.178
1	3	0.054	0.054	-0.034	-0.034	3.177	3.178
1	4	0.054	0.054	-0.034	-0.034	3.177	3.178
2	1	0.076	0.076	-0.030	-0.031	3.275	3.278
2	2	0.076	0.076	-0.030	-0.031	3.275	3.278
2	3	0.076	0.076	-0.030	-0.031	3.275	3.278
2	4	0.076	0.076	-0.030	-0.031	3.275	3.278
3	1	0.093	0.093	-0.034	-0.034	3.369	3.377
3	2	0.093	0.093	-0.034	-0.034	3.369	3.377
3	3	0.093	0.093	-0.034	-0.034	3.369	3.377
3	4	0.093	0.093	-0.034	-0.034	3.368	3.377
6	1	0.132	0.132	-0.054	-0.052	3.615	3.617
6	2	0.132	0.132	-0.054	-0.052	3.615	3.617
6	3	0.132	0.132	-0.054	-0.052	3.615	3.617
6	4	0.132	0.132	-0.054	-0.052	3.617	3.617
12	1	0.187	0.187	-0.093	-0.091	3.904	3.888
12	2	0.187	0.187	-0.093	-0.091	3.904	3.888
12	3	0.187	0.187	-0.093	-0.091	3.903	3.888
12	4	0.187	0.187	-0.094	-0.091	3.896	3.888

Table 3
Robustness: Number of strike prices

This table evaluates the impact of available number of strike prices on the numerical performance of the implied binomial tree method. All binomial trees are constructed with 120 periods. Option maturity is 3 months and the strike prices are equally spaced over an interval covering 3 standard deviations on either side the initial stock price. The risk-neutral density from the implied binomial tree is then used to calculate volatility, skewness and kurtosis of the risk-neutral density. For comparison purposes, the corresponding accurate moments are calculated using the moment generating function (MGF) of the theoretical density function of the assumed SVJ model. Estimates from the standard implied binomial tree method are not separately reported in the table since they are identical the csBIN estimates when the bandwidth parameter h is equal to 1. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs. (7) and (8) with the same set of parameters as in Table 1.

Number of strikes	Bandwidth (h)	Volatility		Skewness		Kurtosis	
		csBIN	MGF	csBIN	MGF	csBIN	MGF
21	1	0.093	0.093	-0.034	-0.034	3.365	3.376
21	2	0.093	0.093	-0.034	-0.034	3.365	3.376
21	3	0.093	0.093	-0.034	-0.034	3.365	3.376
21	4	0.093	0.093	-0.034	-0.034	3.364	3.376
17	1	0.093	0.093	-0.034	-0.034	3.370	3.376
17	2	0.093	0.093	-0.034	-0.034	3.371	3.376
17	3	0.093	0.093	-0.034	-0.034	3.370	3.376
17	4	0.093	0.093	-0.033	-0.034	3.371	3.376
13	1	0.093	0.093	-0.034	-0.034	3.363	3.376
13	2	0.093	0.093	-0.034	-0.034	3.363	3.376
13	3	0.093	0.093	-0.034	-0.034	3.363	3.376
13	4	0.093	0.093	-0.034	-0.034	3.363	3.376
11	1	0.093	0.093	-0.034	-0.034	3.368	3.376
11	2	0.093	0.093	-0.035	-0.034	3.368	3.376
11	3	0.093	0.093	-0.035	-0.034	3.368	3.376
11	4	0.093	0.093	-0.035	-0.034	3.367	3.376
9	1	0.093	0.093	-0.040	-0.034	3.419	3.376
9	2	0.093	0.093	-0.040	-0.034	3.419	3.376
9	3	0.093	0.093	-0.040	-0.034	3.419	3.376
9	4	0.093	0.093	-0.040	-0.034	3.420	3.376
7	1	0.093	0.093	-0.025	-0.034	3.473	3.376
7	2	0.093	0.093	-0.025	-0.034	3.472	3.376
7	3	0.093	0.093	-0.025	-0.034	3.473	3.376
7	4	0.093	0.093	-0.025	-0.034	3.473	3.376

Table 4
Robustness: Uneven increment in strike prices

This table evaluates how uneven increment in strike prices affects the numerical performance of the implied binomial tree method. All binomial trees are constructed with 120 periods, matched to 21 option prices associated with strike prices covering 3 standard deviations around the initial stock price. From the at-the-money strike price, the gap between two adjacent strike prices increases by 25% each time as we move away from the money. The risk-neutral density from the implied binomial tree is then used to calculate volatility, skewness and kurtosis of the risk-neutral density. For comparison purposes, the corresponding accurate moments are calculated using the moment generating function (MGF) of the theoretical density function of the assumed SVJ model. Estimates from the standard implied binomial tree method are not separately reported in the table since they are identical the csBIN estimates when the bandwidth parameter h is equal to 1. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs. (7) and (8) with the same set of parameters as in Table 1.

Maturity (month)	Bandwidth (h)	Volatility		Skewness		Kurtosis	
		csBIN	MGF	csBIN	MGF	csBIN	MGF
0.5	1	0.038	0.038	-0.043	-0.044	3.134	3.154
0.5	2	0.038	0.038	-0.043	-0.044	3.134	3.154
0.5	3	0.038	0.038	-0.043	-0.044	3.133	3.154
0.5	4	0.038	0.038	-0.043	-0.044	3.133	3.154
1	1	0.054	0.054	-0.035	-0.034	3.173	3.178
1	2	0.054	0.054	-0.035	-0.034	3.173	3.178
1	3	0.054	0.054	-0.035	-0.034	3.173	3.178
1	4	0.054	0.054	-0.035	-0.034	3.173	3.178
2	1	0.076	0.076	-0.035	-0.031	3.290	3.278
2	2	0.076	0.076	-0.035	-0.031	3.290	3.278
2	3	0.076	0.076	-0.035	-0.031	3.290	3.278
2	4	0.076	0.076	-0.035	-0.031	3.295	3.278
3	1	0.093	0.093	-0.039	-0.034	3.398	3.377
3	2	0.093	0.093	-0.039	-0.034	3.398	3.377
3	3	0.093	0.093	-0.039	-0.034	3.398	3.377
3	4	0.093	0.093	-0.040	-0.034	3.401	3.377
6	1	0.132	0.132	-0.054	-0.052	3.666	3.617
6	2	0.132	0.132	-0.054	-0.052	3.667	3.617
6	3	0.132	0.132	-0.054	-0.052	3.671	3.617
6	4	0.132	0.132	-0.050	-0.052	3.690	3.617
12	1	0.187	0.187	-0.090	-0.091	3.850	3.888
12	2	0.187	0.187	-0.090	-0.091	3.850	3.888
12	3	0.187	0.187	-0.090	-0.091	3.850	3.888
12	4	0.187	0.187	-0.090	-0.091	3.849	3.888

Table 5
Robustness: Higher volatility and greater jump risk

This table evaluates how higher volatility and greater jump risk affect the numerical performance of the implied binomial tree method. All binomial trees are constructed with 120 periods, matched to 21 option prices with strike prices equally spaced over an interval covering 4 standard deviations on either side the initial stock price. The risk-neutral density from the implied binomial tree is then used to calculate volatility, skewness and kurtosis of the risk-neutral density. For comparison purposes, the corresponding accurate moments are calculated using the moment generating function (MGF) of the theoretical density function of the assumed SVJ model. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs. (7) and (8) with the following parameters: $k_v = 1$, $\sigma_v = 0.25$, $\rho = 0$, $\lambda = 0.25$, $\mu_J = -0.15\sigma_I$, $\sigma_J = 0.25\sigma_I$, $\theta_v = 0.75k_v\sigma_I^2$, $V_0 = \theta_v/k_v$ and $\sigma_I = 0.3$ is the annualized integrated volatility. The risk-free rate is assumed to be zero for convenience. With this base set of parameters, the diffusion (jump) portion of the volatility accounts for 25% (75%) of the total volatility.

Maturity (month)	Bandwidth (h)	Volatility		Skewness		Kurtosis	
		csBIN	MGF	csBIN	MGF	csBIN	MGF
0.5	1	0.058	0.058	-0.861	-0.908	8.286	9.025
0.5	2	0.058	0.058	-0.862	-0.908	8.287	9.025
0.5	3	0.058	0.058	-0.860	-0.908	8.267	9.025
0.5	4	0.058	0.058	-0.856	-0.908	8.214	9.025
1	1	0.082	0.082	-0.633	-0.644	5.895	6.050
1	2	0.082	0.082	-0.633	-0.644	5.895	6.050
1	3	0.082	0.082	-0.633	-0.644	5.894	6.050
1	4	0.082	0.082	-0.633	-0.644	5.891	6.050
2	1	0.116	0.116	-0.457	-0.459	4.552	4.595
2	2	0.116	0.116	-0.457	-0.459	4.552	4.595
2	3	0.116	0.116	-0.457	-0.459	4.553	4.595
2	4	0.116	0.116	-0.461	-0.459	4.600	4.595
3	1	0.142	0.142	-0.384	-0.380	4.180	4.134
3	2	0.142	0.142	-0.384	-0.380	4.181	4.134
3	3	0.142	0.142	-0.385	-0.380	4.196	4.134
3	4	0.142	0.142	-0.389	-0.380	4.248	4.134
6	1	0.201	0.201	-0.284	-0.284	3.728	3.726
6	2	0.201	0.201	-0.284	-0.284	3.728	3.726
6	3	0.201	0.201	-0.284	-0.284	3.730	3.726
6	4	0.201	0.201	-0.282	-0.284	3.729	3.726
12	1	0.285	0.284	-0.236	-0.231	3.647	3.578
12	2	0.285	0.284	-0.236	-0.231	3.648	3.578
12	3	0.285	0.284	-0.236	-0.231	3.646	3.578
12	4	0.285	0.284	-0.231	-0.231	3.585	3.578

Figure 1
Risk-neutral density

This figure illustrates the risk-neutral density extracted from option prices using the implied binomial tree methods. Both the standard implied binomial tree (BIN) method and the cubic spline smoothed implied binomial tree (csBIN) method are used to estimate the density function. Both binomial trees are constructed with 120 periods, matched to 21 option prices with strike prices equally spaced over an interval covering 3 standard deviations on either side the initial stock price. The bandwidth (h) used for the csBIN is 4. The underlying stock price is assumed to follow a stochastic volatility with jumps (SVJ) model specified in Eqs (7) and (8) with the following parameters: $k_v = 1$, $\sigma_v = 0.25$, $\rho = 0$, $\lambda = 0.5$, $\mu_J = -0.15\sigma_I$, $\sigma_J = 0.15\sigma_I$, $\theta_v = 0.85k_v\sigma_I^2$, $V_0 = \theta_v/k_v$ and $\sigma_I = 0.2$ is the annualized integrated volatility. The risk-free rate is assumed to be zero for convenience. With this base set of parameters, the diffusion (jump) portion of the volatility accounts for 15% (85%) of the total volatility.