Received: 29 September 2017 | Accepted: 4 February 2018
DOI: 10.1002/fut.21911

RESEARCH ARTICLE

WILEY

A simple iteration algorithm to price perpetual Bermudan options under the lognormal jump-diffusion-ruin process

San-Lin Chung[1] | Jr-Yan Wang[2]

[1] Department of Finance, National Taiwan University, Taipei, Taiwan

[2] Department of International Business, National Taiwan University, Taipei, Taiwan

Correspondence
Jr-Yan Wang, Department of International Business, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei 106, Taiwan.
Email: jryanwang@ntu.edu.tw

Funding information
Ministry of Science and Technology, Taiwan

We propose an analytical-form framework for pricing perpetual Bermudan options (PBOs) under the lognormal jump-diffusion-ruin model of Merton (1976). We first analytically derive the holding and early exercise values of PBOs. The optimal exercise boundary of the PBO, determined by equating the holding and early exercise values, is then solved using an iteration algorithm. We finally evaluate the PBO by taking the expectation of the option prices at the subsequent exercisable date and discounting it at the risk-free rate. The numerical results indicate that our method is far more efficient than the competing methods in the literature for pricing PBOs.

KEYWORDS
analytical form solution, jump-diffusion process, jump-to-ruin model, optimal exercise boundary, perpetual bermudan option

JEL CLASSIFICATION
G13

1 | INTRODUCTION

Bermudan options are non-standard American options that can be exercised only on a number of specified dates during their life. Perpetual Bermudan options (PBOs) are one special case of the general Bermudan option where the inter-exercise time is constant and the time to maturity is increased to infinity. In the financial literature, many corporate finance decisions are analyzed using the framework of perpetual options. For example, Mcdonald and Siegel (1986), Boyle and Guthrie (2003), Guthrie (2007), and Sundaresan and Wang (2007), among others, assume that a firm has the perpetual rights to invest in a project, Quigg (1993) considers that a landholder holds a perpetual option to construct a building, Leland (1994) supposes that stockholders have a perpetual American option to default, and Lambrecht and Myers (2007) analyze acquirers' takeover option as a perpetual put option. As discussed in Chung and Shackleton (2007), although previous studies usually adopt perpetual American options for their analyses, many corporate finance decisions are actually made discretely with potentially infinite time horizons and thus are similar to the cases of PBOs. Moreover, in capital markets, perpetual contingent convertible bonds are important debt instruments embedded with the feature of perpetual Bermudan puts (contingent on issuers' capital values).[1] When applying the

[1]Perpetual contingent convertible bonds (PCC bonds) can be treated as the capital buffer to aid banks to meet the capital requirements regulated in Basel III. Holders of PCC bonds agree to take equity in exchange for the debt at a pre-specified conversion ratio when the issuer's Tier-1 capital ratio, usually observed quarterly, falls below a certain level. In addition, PCC bonds are usually callable. To price PCC bonds exactly, one needs to model the stochastic processes of the capital ratio, equity prices, and interest rates, and tackle the infinite-maturity feature simultaneously. The valuation of PCC bonds is very complicated and beyond the scope of this paper. Nevertheless, the proposed method provides a possible starting point for developing pricing methods for PCC bonds.

PBO framework to study the above financial issues in an indefinite horizon, it seems essential to take the default risk of the underlying firms into account. To cope with this important feature, our paper offers an analytical-form framework for the valuation of PBOs under the lognormal jump-diffusion-ruin model of Merton (1976).[2]

Perpetual Bermudan options are no easier to price than the American options.[3] To the best of our knowledge, only a few studies have considered the valuation problem of PBOs. Boyarchenko and Levendorskii (2002) solve the pricing problem using the Wiener–Hopf factorization technique and derive approximate formula for certain underlying processes such as normal inverse Gaussian processes. Alobaidi, Mansi, and Mallier (2014) discretize the integrals in the Wiener–Hopf method to obtain a linear system and solve the system by the value-matching condition. Ma and Luo (2012) express the expected payoff from holding a PBO as a sum of iterative integrals, since holders of the PBO will eventually earn the exercise value at one of the future exercisable time points. Lattice-based solutions, such as binomial tree and explicit finite difference models, have been proposed by Lin and Liang (2007) and Muroi and Yamada (2008), respectively.

Except for Boyarchenko and Levendorskii (2002), all the above-mentioned models evaluate PBOs under the Pure Diffusion (PD) process. However, many empirical studies have confirmed that the jump behavior can be observed in the prices of almost all types of assets. Furthermore, the default risk cannot be ignored when pricing any asset with a long life, not to mention an indefinite life for PBOs. To combine the above two features, we propose using the lognormal jump-diffusion-ruin model of Merton (1976) for the valuation of PBOs. To the best our knowledge, this paper is the first one that can evaluate perpetual Bermudan and American options under the lognormal jump-diffusion-ruin processes.

It is well known that PBO prices follow a periodic property, that is, $V(S,t) = V(S,t+\tau)$, where $V(S,t)$ is the PBO price at time t with the stock price equaling S, and τ is the time interval between two neighboring exercisable time points of the PBO. Using this periodic property, this paper proposes a simple iteration algorithm to determine the optimal exercise boundary[4] by pricing PBOs at the exercisable time points. Then PBO prices at the non-exercisable time points can be evaluated by simply taking the expectation of the option prices at the subsequent exercisable time point and then discounting it at the risk-free rate. Moreover, we develop a regression-based extrapolation approach to evaluate perpetual American options based on prices of PBOs given different inter-exercise time intervals. Since a PBO is more difficult to be evaluated than a Bermudan option with a finite maturity, we contribute to the literature by proposing the most efficient pricing method so far for PBOs under the lognormal jump-diffusion-ruin process. It should be noted that the proposed algorithm in this paper is designed specifically for PBOs because it exploits the periodic property of PBOs. It is not our intention to develop a general pricing method for Bermudan options with different times to maturity. In fact, when the time to maturity is finite, many lattice models are available to achieve accurate pricing for Bermudan options.

Specifically, the proposed method can be applied to the pricing of PBOs under not only the PD process in Black and Scholes (1973) model but also the Lognormal Jump-Diffusion (LJD) process and the Lognormal Jump-Diffusion-Ruin (LJDR) process. Even under the assumption of the PD process, our method is far more efficient than the explicit finite difference method for evaluating PBOs. For example, by controlling the same required computational time, Figure 1 indicates that the percentage pricing errors of the proposed method are less than one hundredth of those of the explicit finite difference method. When applying the proposed method to pricing PBOs under even the most complicated LJDR processes, similar degrees of accuracy can be achieved as those under the PD process, although more time is unavoidably consumed.

The rest of the paper is organized as follows. Section 2 describes the pricing model and the proposed iteration algorithm for the valuation of PBOs. Section 3 discusses the numerical results of the benchmark method and the proposed option pricing method. Section 4 concludes the paper.

[2]The lognormal jump-diffusion-ruin model is a combination of two special cases, the lognormal jump-diffusion process and the jump-to-ruin diffusion process, of Merton (1976). Please refer to Equation (1) of this paper for the detailed specification.

[3]One exception is to price perpetual American options under the pure-diffusion stock price process. Due to the time-homogeneous nature of the valuation problem, the Black–Scholes–Merton's partial differential equation is thus simplified to an ordinary differential equation, which can be solved analytically, as shown in Merton (1973). However, this approach does not work for lognormal jump-diffusion-ruin processes that we consider in this paper.

[4]Note that the optimal exercise boundary is identical at all exercise time points due to the periodic property. To the best of knowledge, our paper is the first one to take advantage of this property to efficiently solve the optimal exercise boundary and obtain PBO prices.

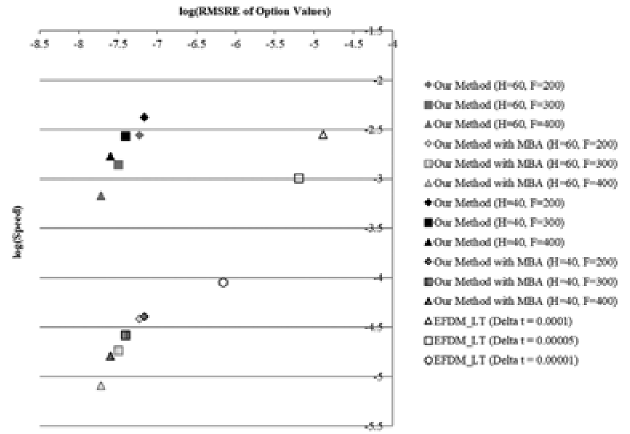

FIGURE 1 Speed-Accuracy Analysis for Different Methods under the PD process: Figure 1 compares the accuracy and speed of the proposed method and the explicit finite difference method with logarithmic transformation (EFDM_LT) for pricing perpetual Bermudan options under the PD process ($\lambda_1 = \lambda_2 = 0$). We employ the speed (number of options priced per second) and the root mean squared relative errors (RMSREs) of different methods for the 150 option contracts in Appendix A to plot this figure. In our method, Newton's method is used to determine the critical stock price S^*. When the moving boundary approach (MBA) proposed by Muthuraman (2008) is used to find S^*, it is denoted as our method with MBA

2 | THE MODEL AND THE METHODOLOGY

Without loss of generality, we consider the pricing of a perpetual Bermudan put option with a strike price X at an exercisable time point t. Of course, it is just as straightforward to apply our method to pricing perpetual Bermudan call options. Extending Merton's (1976) model, we assume that the underlying asset price S_t under the risk-neutral measure follows the lognormal jump-diffusion-ruin process, that is,[5]

$$\frac{dS_t}{S_t} = (r - q - \lambda_1 K + \lambda_2)dt + \sigma dW_t + dP_{1t} + dP_{2t}, \tag{1}$$

where W_t is a standard Wiener process, P_{1t} and P_{2t} are two individual Poisson processes with the jump intensities to be λ_1 and λ_2, respectively, and W_t, P_{1t}, and P_{2t} are mutually independent. If the Poisson event P_{1t} occurs, $J - 1$ is the random percentage change in the underlying asset price, and the Poisson process P_{2t} represents the default event of the issuer of the underlying asset. To maintain the martingale property of the underlying asset price, the adjustment term $(-\lambda_1 K + \lambda_2)$ is introduced in the drift term of the S_t process, where $K \equiv E[J - 1]$ and the reason to add λ_2 is because there is a -100% percentage change in the underlying asset price when S_t jumps to zero in the event of default. In addition, the volatility σ, the risk-free rate r, and the dividend yield rate q are all constant.

Applying the Itô's Lemma to Equation (1), we derive:

$$\begin{cases} S_{t+\tau} = 0 \quad \text{if the default occurs in } (t, t+\tau] \\ \ln S_{t+\tau} = \ln S_t + \left(r - q - \dfrac{\sigma^2}{2} - \lambda_1 K + \lambda_2\right)\tau + \sigma(W_{t+\tau} - W_t) + J(P_1(\tau)) \text{ o/w} \end{cases}, \tag{2}$$

where τ is the time interval between two exercisable time points, $P_1(\tau)$ is the number of the Poisson jumps occurring in the interval of $(t, t+\tau]$, and $J(P_1(\tau)) = 0$ if $P_1(\tau)$is zero; $J(P_1(\tau)) = \sum_{m=1}^{P_1(\tau)} \ln J_m$ for $P_1(\tau) \geq 1$, where the jump size J_m follows an independently and identically lognormal distribution, that is, $\ln J_m \, N(\mu_J, \sigma_J^2)$. Therefore, the variable K in the drift should be

[5]We thank the referee to point out that pricing options under Equation (1) may not be arbitrage free. However, as an application, we consider this lognormal jump-diffusion-ruin process because it can nest the PD process in Black and Scholes (1973) and LJD in Merton (1976) as special cases such that we can compare our pricing algorithm to existing methods based on these two processes. In fact, the PD and LJD processes have been used widely for decades under the presumption that the corresponding option pricing methods have been verified to be arbitrage free.

$e^{\gamma}-1$, where $\gamma \equiv \mu_J + \sigma_J^2/2$. Last, based on the results in Merton (1976), the transition density function of the stock price under the proposed model is expressed as follows:

$$\begin{cases} \phi(S_{t+\tau}=0|\ln S_t) = 1-e^{-\lambda_2\tau} \quad \text{if the default occurs in } (t,t+\tau] \\ \phi(\ln S_{t+\tau}|\ln S_t) = e^{-\lambda_2\tau}\Sigma_{m=0}^{\infty}\dfrac{e^{-\lambda_1\tau}(\lambda_1\tau)^m}{m!}\dfrac{1}{\sqrt{2\pi v_m^2\tau}}e^{-\dfrac{\left[\ln S_{t+\tau}-\ln S_t-\left(r_m-q-v_m^2/2\right)\tau\right]^2}{2v_m^2\tau}}, \quad \text{o/w} \end{cases} \tag{3}$$

where $v_m^2 \equiv \sigma^2 + m\sigma_J^2/\tau$ and $r_m \equiv r - \lambda_1 K + \lambda_2 + m\gamma/\tau$, conditional on knowing that there are exactly m Poisson jumps in the interval of $(t,t+\tau]$.

The periodic property implies that critical stock prices, which separate the exercise and holding regions, are all the same at each exercisable time point and thus are time independent. When this critical stock price (denoted as S^*) is given,[6] the holding value at t, that is, the PBO price in the holding region ($S_t > S^*$), is contributed by three components: the expected payoff value given default, the expected early exercise values, and the expected holding values at the next exercisable time point. Specifically, the holding value $\mathrm{HV}(S_t)$ follows:

$$\begin{aligned} \mathrm{HV}(S_t) &= e^{-r\tau}\left(1-e^{-\lambda_2\tau}\right)X \\ &+e^{-(r+\lambda_2)\tau}\int_{-\infty}^{\ln S^*}(X-S_{t+\tau})^+\phi(\ln S_{t+\tau}|\ln S_t)d\ln S_{t+\tau} \\ &+e^{-(r+\lambda_2)\tau}\int_{\ln S^*}^{\ln S_{\max}}\mathrm{HV}(S_{t+\tau})\phi(\ln S_{t+\tau}|\ln S_t)d\ln S_{t+\tau}. \end{aligned} \tag{4}$$

The first component, the expected value given default (denoted as $\mathrm{EVGD}(S_t,\tau)$), represents the present value of the expected payoff given the default occurring in the following interval of $(t,t+\tau]$. Since $S_{t+\tau}=0$ in the event of default, we assume that holders of the perpetual Bermudan put receive the strike price $(X-S_{t+\tau})^+ = X$ at $t+\tau$. Although the value of EVGD is independent of S_t in the current setting, we still introduce S_t as a parameter of EVGD to maintain the generality of our method for pricing options with more complicated payoff functions in the event of default. The second component, the expected early exercise values (denoted as $\mathrm{EEEV}(S_t,\tau)$), of the above equation has a simple Merton-type closed-form solution:

$$\mathrm{EEEV}(S_t,\tau) = e^{-\lambda_2\tau}\Sigma_{m=0}^{\infty}\frac{e^{-\lambda_1'\tau}\left(\lambda_1'\tau\right)^m}{m!}f\left(S_t,X,v_m^2,r_m,q,\tau,S^*\right), \tag{5}$$

where $\lambda_1' \equiv \lambda_1 e^{\gamma}$, $f\left(S_t,X,v_m^2,r_m,q,\tau,S^*\right) = Xe^{-r_m\tau}N(-d_{m,2}) - S_t e^{-q\tau}N(-d_{m,1})$, $d_{m,1} = \frac{\ln\left(S_t/S^*\right)+\left(r_m-q+v_m^2/2\right)\tau}{v_m\sqrt{\tau}}$, and $d_{m,2} = d_{m,1} - v_m\sqrt{\tau}$.

To evaluate the expected holding values at the next exercisable time point (the last component of Equation (4)), we suggest using the Gauss-Legendre Quadrature (GQ) method due to its higher order of convergence rate. Moreover, we truncate the upside of the holding region with a maximum stock price, which is defined as

$$S_{\max} = S^* e^{H\sigma^*}, \tag{6}$$

where $\sigma^* \equiv \sqrt{\sigma^2 + \lambda\left(\mu_J^2 + \sigma_J^2\right)}$ is the expected annual volatility of the logarithmic stock price and H is a multiplicative factor. As explained later, maintaining a constant distance between $\ln S_{\max}$ and $\ln S^*$ can improve the efficiency of using the GQ method in the proposed approach. It is straightforward to tell that a larger value of H leads to a more ideal situation where $S_{\max}$ can further approach infinity. However, extremely large values of $S_{\max}$ could cause serious round-off errors because of the limited precision for numerical computations of a computer. In our experiments, $H = 40$ and $H = 60$ are examined, that is, the log-difference between

[6]Based on the proposed method, the critical stock price is the solution of Equation (14), which will be introduced later.

$S_{\max}$ and S^* is 40 or 60 times the expected annual volatility of the logarithmic stock price. Note that the above equation is not the only way to decide $S_{\max}$, but this setting is believed to be conservative and able to capture the effective holding region for pricing PBOs.[7]

We set the number of abscissas in the holding region as:

$$n = \left[\frac{\ln \frac{S_{\max}}{S^*}}{\tau^{0.25}/F}\right], \tag{7}$$

where $[d]$ denotes the integer closest to d, and F is a multiplying factor introduced to scale up the number of abscissas. The exponent of τ is suggested to be 0.25 due to the results of our experiments reported in a later section.

Two new variables y and x are introduced to represent the log prices at the current and next exercisable time points, respectively, and k_m is defined as an elasticity parameter as follows:

$$y= \ln S_t,\ \ x= \ln S_{t+\tau},\ \ k_m \equiv \frac{2(r_m - q)}{v_m^2} - 1.$$

Following Andricopoulos, Widdicks, Duck, and Newton (2003), the interim functions $B_m(y,x,\tau)$ for density and $A_m(y,\tau)$ for normalization and discounting, conditional on m events of the P_{1t} process occurring, allow the time-t and time-$(t+\tau)$ holding value functions $\mathrm{HV}(S_t) = \mathrm{HV}(e^y)$ and $\mathrm{HV}(S_{t+\tau}) = \mathrm{HV}(e^x)$ to be linked via the following integration:

$$\begin{aligned}\mathrm{HV}(e^y) &= \mathrm{EVGD}(e^y,\tau) + \mathrm{EEEV}(e^y,\tau)\\ &+\int_{\ln S^*}^{\ln S_{\max}} \mathrm{HV}(e^x)\Sigma_{m=0}^{\infty}\frac{e^{-\lambda_1\tau}(\lambda_1\tau)^m}{m!}A_m(y,\tau)B_m(y,x,\tau)dx,\end{aligned} \tag{8}$$

where

$$A_m(y,\tau) = \frac{1}{\sqrt{2\pi v_m^2\tau}}e^{-\frac{1}{2}k_m y-\frac{1}{8}k_m^2 v_m^2\tau-(r+\lambda_2)\tau}, \tag{9}$$

$$B_m(y,x,\tau) = e^{-\frac{(x-y)^2}{2v_m^2\tau}+\frac{1}{2}k_m x}. \tag{10}$$

At any exercisable time point, denote the stock price and the holding value of the perpetual Bermudan put at the i-th abscissa as $S(i)$ and $\mathrm{HV}(i)$, respectively, for $i = 1,2,\ldots,n$. When applying the GQ algorithm to calculate the integration in Equation (8) numerically, the weights w_j^{GQ} and the abscissas a_j^{GQ} in the GQ method are determined by solving the following equation:

$$\Sigma_{j=1}^{n}(a_j^{\mathrm{GQ}})^l w_j^{\mathrm{GQ}} = \int_{-1}^{1} z^l dz\ \ \forall l\in\{0, 1,\ldots,2n-1\}.$$

Using the periodic property and the chosen weights and abscissas, the holding value function of Equation (8) follows a quadrature expression:

$$\begin{aligned}\mathrm{HV}(i) &= \mathrm{EVGD}(e^{x_i},\tau) + \mathrm{EEEV}(e^{x_i},\tau)\\ &+\Sigma_{j=1}^{n}\ \mathrm{HV}(j)\Sigma_{m=0}^{\infty}\ \frac{e^{-\lambda_1\tau}(\lambda_1\tau)^m}{m!}A_m(x_{\mathrm{i}},\tau)B_m(x_i,x_j,\tau)w_j,\end{aligned} \tag{11}$$

[7]For example, when $S^* = 74.2620$ and $\sigma^* = 0.3$ (the last option contract in Appendix A), $H = 60$ can generate a value for $S_{\max}$ to be 4.88E+09, which is sufficiently high to represent the possibly maximal stock price for a long period of time.

where $w_j = \frac{b-a}{2} w_j^{\mathrm{GQ}}$, $x_j = \frac{b-a}{2} a_j^{\mathrm{GQ}} + \frac{(b+a)}{2}$, $a = S(1) = \ln S^*$, and $b = S(n) = \ln S_{\max}$. Note that the abscissas x_i at time t (for determining $S(i) = e^{x_i}$ and $\mathrm{HV}(i)$), for $i = 1, \ldots, n$, are the same as the abscissas x_j at time $t + \tau$ (for determining $S(j) = e^{x_j}$ and $\mathrm{HV}(j)$), for $j = 1, \ldots, n$, respectively. Moreover, according to the periodic property, $\mathrm{HV}(i) = \mathrm{HV}(j)$ if $i = j$.

Therefore, we rewrite Equation (11) as the following matrix-vector form:

$$I \times \mathrm{HV} = I \times (\mathrm{EVGD} + \mathrm{EEEV}) + M_\Sigma \times \mathrm{HV}, \tag{12}$$

where I is the $n \times n$ identity matrix and each of HV, EVGD, and EEEV is an $n \times 1$ vector across $S(i) = e^{x_i}$, for $i = 1, \ldots, n$. In addition, by defining the $n \times n$ matrix $M_m(i,j) \equiv A_m(x_i, \tau) B_m(x_i, x_j, \tau) w_j$ for $1 \le i, j \le n$, the $n \times n$ matrix M_Σ can be derived via $M_\Sigma = \Sigma_{m=0}^{\infty} \frac{e^{-\lambda_1 \tau}(\lambda_1 \tau)^m}{m!} M_m$. Finally, from Equation (12), the vector of holding values can be solved as follows:

$$\mathrm{HV} = (I - M_\Sigma)^{-1} \times (\mathrm{EVGD} + \mathrm{EEEV}). \tag{13}$$

The implementation of Equation (13) relies on the condition that the critical stock price must be known. Thus, we conjecture a reasonable initial value of the critical stock price, denoted as S_0^*,[8] and obtain the initial vector of holding values HV_0 via Equation (13). The critical stock price in the next iteration is then the solution of S in the following equation:

$$\begin{aligned} X - S = {} & \mathrm{EVGD}(S, \tau) + \mathrm{EEEV}(S, \tau) \\ & + \Sigma_{j=1}^{n} \mathrm{HV}(j) \Sigma_{m=0}^{\infty} \frac{e^{-\lambda_1 \tau}(\lambda_1 \tau)^m}{m!} A_m(\ln S, \tau) B_m(\ln S, x_j, \tau) w_j, \end{aligned} \tag{14}$$

where the right-hand side of Equation (14) is derived by substituting HV_0 into it and replacing e^{x_i} with S in Equation (11). We next employ Newton's method together with the numerical differentiation to find the solution of S in Equation (14). For solving Equation (14), the convergence criterion of Newton's method is 1.0E-10, and on average five to seven iterations are sufficient to obtain a solution of the critical stock price.

Based on the new critical stock price S_1^*, we repeat the evaluation of Equations (13) and (14) alternately until the critical stock price converges. The iterative procedure for finding the next critical stock price S_{k+1}^* continues until the difference between S_k^* and S_{k+1}^* is smaller than 1.0E-10. It is worth noting that the proposed method is efficient in solving the vector of holding values HV in Equation (13) during the iterative procedure. Since $S_{\max,\kappa} = S_k^* e^{H\sigma^*}$ and thus the distance between the $a = \ln S_k^*$ and $b = \ln S_{\max,k}$ is fixed to be $H\sigma^*$, the relative differences between any two abscissas x_i and x_j remain unchanged when S_k^* varies. Consequently, $M_m(i,j) \equiv A_m(x_i, \tau) B_m(x_i, x_j, \tau) w_j$ and thus M_Σ do not change during the iterative procedure. In other words, the computation of $(I - M_\Sigma)^{-1}$, which is time consuming when n is large, needs to be conducted only in the first iteration. For the following iterations, after the vectors of *EVGD* and *EEEV* are obtained, we can solve the vector of HV by simply performing one matrix multiplication via Equation (13).

Once equipped with the convergent results of holding values HV and the critical stock price S^*, it is straightforward to compute the option value of perpetual Bermudan puts. If a time point that passes the last exercisable time point t by $T \epsilon [0, \tau)$ is considered and the prevailing stock price is S_{t+T}, then the option value can be derived as follows.

$$\begin{aligned} V(S_{t+T}, t+T) = {} & \mathrm{EVGD}(S_{t+T}, \tau - T) + \mathrm{EEEV}(S_{t+T}, \tau - T) + \\ & \Sigma_{j=1}^{n} \mathrm{HV}(j) \Sigma_{m=0}^{\infty} \frac{e^{-\lambda_1(\tau - T)}[\lambda_1(\tau - T)]^m}{m!} A_m(\ln S_{t+T}, \tau - T) B_m(\ln S_{t+T}, x_j, \tau - T) w_j. \end{aligned} \tag{15}$$

When $T = 0$, Equation (15) yields the holding value of the perpetual Bermudan put at the exercisable time point t if S_t is higher than the critical stock price S^*; otherwise, the perpetual Bermudan put should be exercised immediately and thus the option value equals $X - S_t$.

[8]The optimal exercise boundary of the perpetual American put option under the PDF is employed to be the initial guess S_0^* when we implement the computer program, that is, $S_0^* = \theta X / (\theta - 1)$, where $\theta = \sigma^{-2}\left[-(r - q - 0.5\sigma^2) - \sqrt{(r - q - 0.5\sigma^2)^2 + 2\sigma^2 r}\right]$. For details, please refer to Merton (1973).

There is one implementation issue that should be addressed, that is, how to determine the upper limit of m, denoted as m^*, when computing the component of EEEV(S_t,τ) as well as the transition probabilities in Equations (11) to (15). We first examine the cumulative Poisson-jump probability, $\Sigma_{m=0}^{m^*} \frac{e^{-\lambda_1\tau}(\lambda_1\tau)^m}{m!}$, by increasing m^* sequentially such that $1-\Sigma_{m=0}^{m^*} \frac{e^{-\lambda_1\tau}(\lambda_1\tau)^m}{m!}$ is smaller than 1.0E-14.[9] Next, this m^* is employed to generate EEEV(S_t,τ) and the transition probabilities in Equations (11) to (15).

This paper also examines the updating method proposed in Muthuraman (2008) to find the next iteration of S_{k+1}^* under the assumption of the PD process. The basic idea in Muthuraman (2008) is to transform the free boundary problem of pricing American puts into a series of moving boundary problems. Based on a lower-bound initial guess of the critical stock price S_k^* and the corresponding grid of holding values derived by solving the partial differentiation equation, Muthuraman (2008) proposes a rule to derive upward improvements for the next iteration of the critical stock price. He proves that if the current S_k^* is below the optimal critical stock price, then there must exist some values of S above S_k^* such that $\mathrm{HV}_S(S)\leq 1$, where $\mathrm{HV}_S(.)$ is the partial derivative of the holding value with respect to the underlying asset price. Moreover, the current S_k^* can be adjusted upward to be the maximum among those S with $\mathrm{HV}_S(S)\leq -1$, and thus the option value corresponding to the new S_{k+1}^* can be enhanced. For implementation, the value of S_{k+1}^* is determined by finding a value of S upward along the dimension of the stock price until $\mathrm{HV}(S)+S$ is minimized. In addition to employing Newton's method to solve the critical stock price, this paper also adopts the moving boundary approach (MBA) in Muthuraman (2008) to find the critical stock price. We repeat the MBA until the values of S_k^* and S_{k+1}^* converge within 1.0E-10.

3 | NUMERICAL RESULTS

This section is dedicated to demonstrate the advantages of the proposed method for pricing PBOs under the PD ($\lambda_1=\lambda_2=0$), the LJD ($\lambda_2=0$), and the LJDR processes. For each examined process, we first present the speed and accuracy analyses for option values and critical stock prices of PBOs, respectively. Next, several issues associated with τ, the time interval between two neighboring exercisable time points, and n, the number of abscissas in the holding region, is analyzed. We not only identify a proper relation between τ and n but also investigate the convergent results when τ approaching zero, which can be used to approximate the option values of perpetual American options. Before showing the numerical results, we would like to emphasis that the ultimate goal of this paper is to price PBOs under the LJDR process. The reason to test the proposed method under the PD process is because all of the other methods that we can compare are based on the PD process, including the finite difference method for pricing Bermudan option (introduced later), the MBA methods proposed by Muthuraman (2008), and the analytic-form formula for perpetual American options. Therefore, the accuracy and efficiency of our method can be clearly discerned when comparing with those PD-process-based methods.

3.1 | Option values under the PD processes

Table 1 compares pricing errors and computational times of our method (using either Newton's method or the MBA in Muthuraman (2008) to determine the next S_{k+1}^*) and the finite difference method for evaluating PBOs. We employ the explicit finite difference method with the logarithmic transformation (EFDM_LT) technique, one of the most efficient option pricing models as suggested by Geski and Shastri (1985), to approximate PBOs provided that the time to maturity is limited to be 500 years.[10] We choose the upper bound of S to be 1.0E + 09 in the EFDM_LT due to the trade-off between accuracy and efficiency. According to our test, when the upper bound of S is larger than 1.0E + 09 for the method of EFDM_LT, the marginal benefit of a higher upper bound on the accuracy of 500-year Bermudan puts prices is negligible. The differences in option price estimates with the upper bounds of 1.0E + 09 and 1.0E + 10 are smaller than 1.0E-13 in our experiments. Moreover, to improve the convergence rate, we follow the suggestion in Hull (2014) to set the grid distance, $\Delta 1\mathrm{n}\, S$, to be $\sigma\sqrt{1.5\Delta t}$ so as to ensure the values of $\Delta 1\mathrm{n}\, S$ and Δt are well collocated. With this setting, we calculate values of 500-year Bermudan options when Δt equals 0.0001, 0.00005, and 0.00001. The parameters of option contracts examined in Table 1 are adapted from Table 2 of Ju (1998) and Table 4 of Muroi and Yamada (2008). In addition, we consider the time interval between two exercisable time points, τ, to be 0.004 (daily), 0.02 (weekly), 0.083 (monthly), 0.25 (quarterly), 0.5 (semiannually), and 1 (annually). We compute the prices of

[9]In other words, the probability of having more than m^* jumps is less than 1.0E-14.

[10]For 500-year and 1000-year Bermudan puts, the difference between their option values never exceeds 1.0E-13 based on the EFDM_LT. Therefore, we believe 500 years are long enough to evaluate option values of perpetual Bermudan puts when the explicit finite difference method is applied.

TABLE 1 Pricing errors and computational times of different methods for pricing perpetual Bermudan puts under the PD process ($\lambda_1 = \lambda_2 = 0$)

	Our method ($H = 60$, $F = 300$)	**Our method with MBA ($H = 60$, $F = 300$)**	**Our method ($H = 40$, $F = 300$)**	**Our method with MBA ($H = 40$, $F = 300$)**	**EFDM_LT ($\Delta t = 0.0001$)**	**EFDM_LT ($\Delta t = 0.00005$)**	**EFDM_LT ($\Delta t = 0.00001$)**
RMSRE	0.0000032%	0.0000032%	0.0000039%	0.0000039%	0.0013064%	0.0006366%	0.0000686%
RMSRE ($r \le q$)	0.0000036%	0.0000036%	0.0000044%	0.0000044%	0.0002771%	0.0001543%	0.0000330%
RMSRE ($r > q$)	0.0000029%	0.0000029%	0.0000036%	0.0000036%	0.0016714%	0.0008121%	0.0000843%
Time (sec.)	108,444	8,193,793	54,943	5,675,396	53,519	147,380	1,657,850

Table 1 reports the summary statistics of pricing errors and computational times for pricing perpetual Bermudan puts based on our method and the explicit finite difference method with logarithmic transformation (EFDM_LT). The complete table of examined option contracts and generated option values are reported in Appendix A. In addition to our method using Newton's method to determine S^*, we also utilize the moving boundary approach (MBA) in Muthuraman (2008) to determine S^* to generate option prices in our method. To ensure monotonic convergence, when the number of abscissas in the holding region, n, determined according to Equation (7), is less than 5000, then $n = 5000$ is used instead.

perpetual Bermudan puts at the exercisable time point, that is, $T = 0$ in Equation (15). Table 1 shows only summary statistics of pricing errors and computational times of different pricing methods. Detailed pricing results are presented in Appendix A.

Since the GQ method, a numerical integration approach, is used in the proposed method, it can be inferred that a larger value of n (number of abscissas) yields more accurate pricing results. Therefore, this paper employs option values and critical stock prices corresponding to $n \to \infty$ as the benchmarks. To achieve this, we propose a regression-based approach to extrapolate option values and critical stock prices (S^*) of perpetual Bermudan puts for n approaching infinity. We first compute option prices and values of S^* based on our method with $H = 60$ and $n = 6000, 7000, \ldots,$ and 11000 and next perform a quadratic regression of the option values (or S^*) over $1/n$ and $1/n^2$, that is:

$$\text{Option value} = \alpha_0 + \alpha_0(1/n) + \alpha_2(1/n^2) + \varepsilon_V,$$

TABLE 2 Convergence to perpetual American puts under the PD process ($\lambda_1 = \lambda_2 = 0$)

Option parameters		Perpetual American put values	Quadratic regression of benchmark perpetual Bermudan put values over τ				Quadratic regression of perpetual Bermudan put values of our method with $H = 60$ and $F = 300$ over τ				Quadratic regression of perpetual Bermudan put values of our method with $H = 40$ and $F = 300$ over τ				Benchmark option values	Our method with $H = 60$ and $F = 300$	Our method with $H = 40$ and $F = 300$
(r, q, σ)	(S_0, X)		Intercept	Coeff. of τ	Coeff. of τ^2	R^2	Intercept	Coeff. of τ	Coeff. of τ^2	R^2	Intercept	Coeff. of τ	Coeff. of τ^2	R^2	$\tau = 0.004$	$\tau = 0.004$	$\tau = 0.004$
(0.08, 0.12, 0.2)	(80, 100)	31.2500000	31.2505473	−0.3407847	−0.0302201	0.9999872	31.2505460	−0.3407836	−0.0302204	0.9999872	31.2505454	−0.3407800	−0.0302232	0.9999872	31.2487343	31.2487329	31.2487323
(0.08, 0.12, 0.2)	(90, 100)	27.7777778	27.7781621	−0.3003690	−0.0318858	0.9999939	27.7781609	−0.3003680	−0.0318860	0.9999939	27.7781603	−0.3003648	−0.0318885	0.9999939	27.7766527	27.7766515	27.7766509
(0.08, 0.12, 0.2)	(100, 100)	25.0000000	25.0003423	−0.2702412	−0.0288837	0.9999941	25.0003413	−0.2702404	−0.0288839	0.9999941	25.0003408	−0.2702375	−0.0288862	0.9999941	24.9989874	24.9989864	24.9989858
(0.08, 0.12, 0.2)	(110, 100)	22.7272727	22.7275905	−0.2458406	−0.0259170	0.9999937	22.7275895	−0.2458398	−0.0259172	0.9999937	22.7275891	−0.2458372	−0.0259192	0.9999937	22.7263522	22.7263512	22.7263507
(0.08, 0.12, 0.2)	(120, 100)	20.8333333	20.8336264	−0.2253997	−0.0236599	0.9999936	20.8336255	−0.2253990	−0.0236601	0.9999936	20.8336251	−0.2253966	−0.0236620	0.9999936	20.8324895	20.8324886	20.8324882
(0.08, 0.08, 0.2)	(80, 100)	25.5373755	25.5369225	−0.5124515	−0.0454295	0.9999826	25.5369214	−0.5124507	−0.0454296	0.9999826	25.5369209	−0.5124477	−0.0454320	0.9999826	25.5353209	25.5353198	25.5353193
(0.08, 0.08, 0.2)	(90, 100)	21.2470626	21.2474839	−0.4453152	−0.0054569	0.9999938	21.2474830	−0.4453146	−0.0054571	0.9999938	21.2474826	−0.4453121	−0.0054590	0.9999938	21.2453531	21.2453522	21.2453518
(0.08, 0.08, 0.2)	(100, 100)	18.0237917	18.0240615	−0.3753579	−0.0101731	0.9999970	18.0240608	−0.3753573	−0.0101732	0.9999970	18.0240604	−0.3753552	−0.0101749	0.9999970	18.0223416	18.0223408	18.0223404
(0.08, 0.08, 0.2)	(110, 100)	15.5313532	15.5315330	−0.3221439	−0.0113103	0.9999986	15.5315324	−0.3221434	−0.0113104	0.9999986	15.5315321	−0.3221416	−0.0113118	0.9999986	15.5301036	15.5301029	15.5301026
(0.08, 0.08, 0.2)	(120, 100)	13.5581509	13.5583060	−0.2811686	−0.0099716	0.9999987	13.5583054	−0.2811681	−0.0099717	0.9999987	13.5583051	−0.2811666	−0.0099730	0.9999987	13.5570600	13.5570594	13.5570592
(0.08, 0.04, 0.2)	(80, 100)	21.3771652	21.3800946	−1.0004345	−0.3637574	0.9998925	21.3800937	−1.0004333	−0.3637579	0.9998925	21.3800933	−1.0004310	−0.3637597	0.9998925	21.3732645	21.3732636	21.3732631
(0.08, 0.04, 0.2)	(90, 100)	15.8095761	15.8077513	−0.6750271	−0.0708458	0.9999377	15.8077506	−0.6750266	−0.0708459	0.9999377	15.8077503	−0.6750247	−0.0708474	0.9999377	15.8066913	15.8066907	15.8066903
(0.08, 0.04, 0.2)	(100, 100)	12.0700758	12.0700041	−0.5513638	0.0206145	0.9999996	12.0700035	−0.5513634	0.0206144	0.9999996	12.0700033	−0.5513620	0.0206133	0.9999996	12.0678734	12.0678729	12.0678726
(0.08, 0.04, 0.2)	(110, 100)	9.4554118	9.4555093	−0.4349406	0.0191289	0.9999988	9.4555089	−0.4349403	0.0191288	0.9999988	9.4555087	−0.4349392	0.0191280	0.9999988	9.4536865	9.4536861	9.4536859
(0.08, 0.04, 0.2)	(120, 100)	7.5662912	7.5662915	−0.3460605	0.0112374	0.9999998	7.5662911	−0.3460603	0.0112373	0.9999998	7.5662910	−0.3460594	0.0112366	0.9999998	7.5649106	7.5649103	7.5649101
(0.08, 0, 0.2)	(80, 100)	20.0000000	20.0000000	0.0000000	0.0000000	1.0000000	20.0000000	0.0000000	0.0000000	1.0000000	20.0000000	0.0000000	0.0000000	1.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(90, 100)	12.4859015	12.4874248	−1.2142401	−0.2772048	0.9999283	12.4874242	−1.2142393	−0.2772051	0.9999283	12.4874240	−1.2142380	−0.2772061	0.9999283	12.4809293	12.4809287	12.4809285
(0.08, 0, 0.2)	(100, 100)	8.1920000	8.1896715	−0.7475114	−0.0220826	0.9999187	8.1896712	−0.7475111	−0.0220827	0.9999187	8.1896710	−0.7475102	−0.0220834	0.9999187	8.1887377	8.1887373	8.1887372
(0.08, 0, 0.2)	(110, 100)	5.5952462	5.5947691	−0.5414849	0.0497553	0.9999924	5.5947688	−0.5414847	0.0497553	0.9999924	5.5947687	−0.5414841	0.0497548	0.9999924	5.5930180	5.5930178	5.5930177
(0.08, 0, 0.2)	(120, 100)	3.9506173	3.9504853	−0.3866060	0.0405425	0.9999994	3.9504851	−0.3866059	0.0405425	0.9999994	3.9504850	−0.3866054	0.0405421	0.9999994	3.9490440	3.9490439	3.9490438
(0.05, 0, 0.3)	(100, 80)	14.4935536	14.4935040	−0.3817184	0.0071493	0.9999998	14.4935037	−0.3817183	0.0071492	0.9999998	14.4935036	−0.3817181	0.0071492	0.9999998	14.4920242	14.4920239	14.4920237
(0.05, 0, 0.3)	(100, 90)	18.5850416	18.5850277	−0.4907052	0.0115461	0.9999998	18.5850273	−0.4907051	0.0115461	0.9999998	18.5850272	−0.4907048	0.0115461	0.9999998	18.5830805	18.5830801	18.5830799
(0.05, 0, 0.3)	(100, 100)	23.2146791	23.2147612	−0.6154807	0.0196527	0.9999993	23.2147607	−0.6154806	0.0196526	0.9999993	23.2147605	−0.6154803	0.0196526	0.9999993	23.2122294	23.2122290	23.2122288
(0.05, 0, 0.3)	(100, 110)	28.3888136	28.3888878	−0.7525949	0.0260054	0.9999997	28.3888872	−0.7525947	0.0260053	0.9999997	28.3888869	−0.7525943	0.0260053	0.9999997	28.3858179	28.3858173	28.3858171
(0.05, 0, 0.3)	(100, 120)	34.1132497	34.1128001	−0.8922458	0.0126466	0.9999961	34.1127994	−0.8922456	0.0126466	0.9999961	34.1127991	−0.8922451	0.0126465	0.9999961	34.1096500	34.1096493	34.1096490
RMSRE vs. Perpetual American Put (all)			0.0074509%				0.0074512%				0.0074513%				0.0189428%	0.0189458%	0.0189473%
RMSRE vs. Perpetual American Put ($r \le q$)			0.0015086%				0.0015054%				0.0015039%				0.0063694%	0.0063734%	0.0063754%
RMSRE vs. Perpetual American Put ($r > q$)			0.0095398%				0.0095406%				0.0095409%				0.0238956%	0.0238989%	0.0239005%

This table shows the convergent behavior of perpetual Bermudan put prices to perpetual American put prices. We conduct the quadratic regression of perpetual Bermudan put values over different time intervals between two consecutive exercisable time points, τ, based on the prices of the perpetual Bermudan puts in Table 1 (Appendix A). We next interpret the intercept of the quadratic regression as the approximation of the value of the perpetual American put. In addition to the option values generated by our method (with $(H,F) = (40,300)$ and $(60,300)$, the benchmark option values in Table 1 (Appendix A) are also employed as inputs for the quadratic regression. The last three columns compare the option values in the case of $\tau = 0.004$ (corresponding to daily exercisable) based on our method and the benchmark results reported in Table 1 (Appendix A).

$$\text{Critical stock price} = \beta_0 + \beta_0(1/n) + \beta_2(1/n^2) + \varepsilon_S,$$

where ε_v and ε_s are standard white noises. The benchmark option value (or benchmark S^*) can be obtained as the intercept α_0 (or β_0), since it represents the option value (or S^*) for $n \to \infty$. For each quadratic regression, the R-squared value is required to be higher than 0.999999 to ensure that the convergence behavior is sufficiently satisfactory for obtaining benchmarks precisely. Appendix B presents the details regarding the benchmark prices of perpetual Bermudan puts.

We argue that this regression-based extrapolation approach is superior to the Richardson extrapolation method, which is commonly used in the field of financial engineering. This is because it is almost impossible to measure how close are between the results of Richardson extrapolation and the true benchmarks. As for our extrapolation approach, in contrast, R-squared values can be employed to gauge the extrapolation performance. Since the R-squared values in our numerical results are always higher than 0.999999, we are confident of the accuracy of the benchmarks of option values and the critical stock prices presented in this paper.

We first observe from Table 1 that the proposed method can generate more accurate option prices, but consumes less time than the EFDM_LT. For example, our method with $H = 40$ and $F = 300$ can generate option prices with a root mean squared relative error (RMSRE) of 0.0000039%, which is about 1/18 of the RMSRE of the EFDM_LT with $\Delta t = 0.00001$, but the computation time of our method with $H = 40$ and $F = 300$ is only 1/30 that of EFDM_LT with $\Delta t = 0.00001$. Second, Table 1 shows the proposed method can generate nearly identical pricing errors regardless of using Newton's method or the MBA in Muthuraman (2008) to determine S^*. In fact, the differences of option values of these two approaches are always smaller than 1.0E-07 as shown in Appendix A. In other words, our method can incorporate the main part of the competing method of Muthuraman (2008) under the PD process. Since Muthuraman (2008) already proves his method can generate convergent option prices, our experiment results conclude that it is appropriate to employ Newton's method to solve the critical stock price S^*. Incorporating this method in our method saves a lot of computational time without losing any accuracy. Third, since the EFDM_LT generates option values convergent to our benchmark prices as Δt approaches zero, the accuracy of our benchmark prices is thus verified.

In Figure 1 we compare the accuracy and speed of the proposed method and the EFDM_LT based on the 150 contracts examined in Table 1 (Appendix A). For our method, we further examine different combinations of $H = 40$ and 60 and $F = 200$, 300, and 400. Generally speaking, with the increase of H and F in the proposed method and the decrease of Δt in EFDM_LT, the obtained option values converge to the benchmark at the cost of requiring more computational time. In addition, it can be found that the proposed method dominates the competing methods in both efficiency and accuracy. For instance, the computational time of our method with $(H,F) = (40, 300)$ is almost the same as that of the EFDM_LT with Δt being 0.0001 (54,943 sec. vs. 53,519 sec., as shown in Table 1), but our method's RMSRE with $(H,F) = (40, 300)$ is only 1/300 of that of the EFDM_LT with Δt being 0.0001. Moreover, based on the proposed method, the accuracy of using Newton's method or the MBA in Muthuraman (2008) to determine PBO prices is almost identical. Since the computational time based on Newton's method is only 1/100 of that based on the MBA, our method is dramatically faster than Muthuraman's (2008) method with the same accuracy.

In addition to comparing with the MBA in Muthuraman (2008) and the EFDM_LT, we conduct another analysis to verify the accuracy of the proposed method. It is well known that when the time interval between two consecutive exercisable time points, τ, approaches zero, a perpetual Bermudan put can be regarded as a perpetual American put, for which the closed-form pricing formula has been derived by Merton (1973). The correctness of the proposed method can thus be confirmed if our method can generate accurate option prices for a perpetual American put as τ approaches zero. To demonstrate this point, we perform a regression for the values of perpetual Bermudan puts over τ.

For each set of option parameters (S_0, X, r, q, σ) in Table 1, we specifically regress the perpetual Bermudan put values generated by our method over τ and τ^2, that is:

$$\text{Option value} = \gamma_0 + \gamma_1\tau + \gamma_2\tau^2 + \varepsilon_\tau,$$

where ε_τ is a standard white noise. As such, γ_0, the regression intercept (representing the perpetual Bermudan put value as τ approaches zero), should reflect the corresponding perpetual American put value. Table 2 reports the results of this quadratic regression analysis for all option contracts in Table 1 (Appendix A) and compares the regression intercepts with the analytical option prices of perpetual American puts derived according to Merton (1973). We employ not only the option prices generated from the proposed method (with $(H,F) = (40,300)$ and $(H,F) = (60,300)$), but also the benchmark option values in Table 1 to perform this quadratic regression analysis. Note first that the extremely high R^2 in Table 2 implies that the perpetual Bermudan put prices generated by our method can converge almost perfectly to the corresponding perpetual American put prices when τ

approaches zero, and so the intercept can be an accurate approximation for the price of a perpetual American put. As a matter of fact, all RMSREs between the regression intercepts and the corresponding perpetual American put values are fairly small and less than 0.0075% in Table 2.

Last, we discuss the issue of determining n, which is the number of abscissas in the holding region. Recall that we assume that the value of n is proportional to $\tau^{-0.25}$ in Equation (7), where τ is the time interval between two neighboring exercisable time points. The reason for this assumption is to ensure that the option price errors corresponding to different τ are of similar magnitude given all other option parameters being fixed. Table 3 employs the 13–18th (illustrative cases for $r < q$) and 73rd–78th (illustrative cases for $r > q$) option contracts in Table 1 (Appendix A) as examples to illustrate how to determine the relationship between n and τ.

To generate Table 3, we first evaluate the option contracts (with different $\tau = 0.004, 0.02, 0.083, 0.25, 0.5$, and 1) with n being 5000, 6000, ..., 25000 given $H = 60$. Second, we take the relative option price error of $\tau = 1$ and $n = 5000$ as a reference point[11] and next derive the values of n such that in the cases of $\tau = 0.004, 0.02, 0.083, 0.25$, and 0.5, their relative option price errors are the same as this reference point. For instance, in Panel (a) of Table 3, since the relative option price error of $\tau = 1$ and $n = 5000$ is −167E-10, to obtain the value of n for the case of $\tau = 0.5$, we apply the linear interpolation on the relative option price errors corresponding to $n = 6000$ and 7000 (that is, −179E-10 and −132E-10) to derive that when $n = 6252$, the relative option price error of $\tau = 0.5$ equals −167E-10. Finally, we perform the least-squares regression for the logarithmic values of the obtained interpolated values of n over $1\text{n}(\tau)$, for $\tau = 0.004, 0.02, 0.083, 0.25, 0.5$, and 1. In Panel (a) of Table 3, the regression result is $1\text{n}(n) = -0.2725 \times 1\text{n}(\tau) + 8.5480$ with $R^2 = 0.9986$. The extremely high R^2 implies an almost linear relation between $1\text{n}(n)$ and $1\text{n}(\tau)$. Thus, the regression coefficient in front of $1\text{n}(\tau)$ can be used to determine the exponent of τ in Equation (7). In Panels (a) and (b) of Table 3, the slope coefficients are −0.2725 and −0.2717, respectively. Among our experiments for all option contracts in Table 1 (Appendix A), the slope coefficients are all around −0.25. As a result, to determine the value of n in Equation (7), we assume that n is proportional to $\tau^{-0.25}$.

TABLE 3 Error analyses and linear regression of $1\text{n}(n)$ over $1\text{n}(\tau)$ under the PD process

(a) Contracts 13–18 in Table 1 (Appendix A) ($r = 0.08$, $q = 0.12$, $\sigma = 0.2$, $S_0 = 100$, $X = 100$)

τ	Benchmark option values	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	24.99898742	24.99897870	24.99898136	24.99898297	24.99898401	24.99898473	24.99898524	24.99898562	24.99898591	24.99898613	24.99898631	24.99898645	24.99898657	24.99898667	24.99898675	24.99898682	24.99898687	24.99898692	24.99898697	24.99898701	24.99898704	24.99898707
0.02	24.99485947	24.99485566	24.99485682	24.99485753	24.99485798	24.99485829	24.99485852	24.99485868	24.99485881	24.99485891	24.99485898	24.99485905	24.99485910	24.99485914	24.99485917	24.99485920	24.99485923	24.99485925	24.99485927	24.99485929	24.99485930	24.99485932
0.083	24.97797416	24.97797238	24.97797292	24.97797325	24.97797347	24.97797361	24.97797372	24.97797379	24.97797385	24.97797390	24.97797393	24.97797396	24.97797399	24.97797401	24.97797402	24.97797404	24.97797405	24.97797406	24.97797407	24.97797408	24.97797409	24.97797409
0.25	24.93133218	24.93133122	24.93133151	24.93133169	24.93133181	24.93133188	24.93133194	24.93133198	24.93133202	24.93133204	24.93133206	24.93133208	24.93133209	24.93133210	24.93133211	24.93133212	24.93133212	24.93133213	24.93133213	24.93133214	24.93133214	24.93133214
0.5	24.85766440	24.85766375	24.85766395	24.85766407	24.85766415	24.85766420	24.85766424	24.85766426	24.85766428	24.85766430	24.85766431	24.85766432	24.85766433	24.85766434	24.85766435	24.85766435	24.85766436	24.85766436	24.85766436	24.85766437	24.85766437	24.85766437
1	24.70127749	24.70127708	24.70127721	24.70127728	24.70127733	24.70127736	24.70127739	24.70127741	24.70127742	24.70127743	24.70127744	24.70127745	24.70127745	24.70127746	24.70127746	24.70127746	24.70127747	24.70127747	24.70127747	24.70127747	24.70127747	24.70127748

Relative error vs. Benchmark (×1.0E-10)

τ	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	−3487	−2420	−1778	−1361	−1075	−870	−719	−604	−514	−443	−386	−339	−300	−268	−240	−217	−196	−179	−163	−150	−138
0.02	−1524	−1058	−777	−595	−470	−381	−315	−264	−225	−194	−169	−149	−132	−118	−105	−95	−86	−79	−72	−66	−61
0.083	−715	−496	−365	−279	−221	−179	−148	−124	−106	−91	−79	−70	−62	−55	−50	−45	−41	−37	−34	−31	−29
0.25	−388	−269	−198	−152	−120	−97	−80	−67	−57	−49	−43	−38	−34	−30	−27	−24	−22	−20	−18	−17	−16
0.5	−258	−179	−132	−101	−80	−64	−53	−45	−38	−33	−29	−25	−22	−20	−18	−16	−15	−13	−12	−11	−10
1	−167	−116	−85	−65	−52	−42	−35	−29	−25	−21	−19	−16	−14	−13	−12	−10	−9	−9	−8	−7	−7

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of −167E-10 (the error of $\tau = 1$ and $n = 5000$)	5000	6252	7663	10374	15102	22755
Linear regression of ln(n) over ln(τ)	$\ln(n) = -0.2725 \times \ln(\tau) + 8.5480$, $R^2 = 0.9986$					

(b) Contracts 73–78 in Table 1 (Appendix A) ($r = 0.08$, $q = 0.04$, $\sigma = 0.2$, $S_0 = 100$, $X = 100$)

τ	Benchmark option values	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	12.06787336	12.06786914	12.06787043	12.06787121	12.06787172	12.06787206	12.06787231	12.06787249	12.06787263	12.06787274	12.06787283	12.06787290	12.06787295	12.06787300	12.06787304	12.06787307	12.06787310	12.06787313	12.06787315	12.06787317	12.06787318	12.06787320
0.02	12.05906680	12.05906495	12.05906552	12.05906586	12.05906608	12.05906623	12.05906634	12.05906642	12.05906648	12.05906653	12.05906657	12.05906660	12.05906662	12.05906664	12.05906666	12.05906668	12.05906669	12.05906670	12.05906671	12.05906672	12.05906672	12.05906673
0.083	12.02427749	12.02427662	12.02427688	12.02427704	12.02427715	12.02427722	12.02427727	12.02427731	12.02427734	12.02427736	12.02427738	12.02427739	12.02427740	12.02427741	12.02427742	12.02427743	12.02427743	12.02427744	12.02427744	12.02427745	12.02427745	12.02427745
0.25	11.93326517	11.93326470	11.93326484	11.93326493	11.93326498	11.93326502	11.93326505	11.93326507	11.93326509	11.93326510	11.93326511	11.93326512	11.93326512	11.93326513	11.93326513	11.93326514	11.93326514	11.93326514	11.93326515	11.93326515	11.93326515	11.93326515
0.5	11.79964003	11.79963971	11.79963981	11.79963987	11.79963990	11.79963993	11.79963995	11.79963996	11.79963997	11.79963998	11.79963999	11.79963999	11.79964000	11.79964000	11.79964000	11.79964001	11.79964001	11.79964001	11.79964001	11.79964001	11.79964001	11.79964001
1	11.53922601	11.53922582	11.53922588	11.53922592	11.53922594	11.53922595	11.53922597	11.53922597	11.53922598	11.53922599	11.53922599	11.53922599	11.53922600	11.53922600	11.53922600	11.53922600	11.53922600	11.53922600	11.53922600	11.53922601	11.53922601	11.53922601

Relative error vs. Benchmark (×1.0E-10)

τ	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	−3497	−2428	−1784	−1365	−1079	−874	−722	−607	−517	−445	−388	−341	−302	−269	−242	−218	−198	−180	−165	−151	−139
0.02	−1534	−1065	−783	−599	−473	−384	−317	−266	−227	−196	−170	−150	−133	−118	−106	−96	−87	−79	−72	−67	−61
0.083	−725	−503	−370	−283	−224	−181	−150	−126	−107	−92	−81	−71	−63	−56	−50	−45	−41	−37	−34	−31	−29
0.25	−397	−275	−202	−155	−122	−99	−82	−69	−59	−51	−44	−39	−34	−31	−27	−25	−22	−20	−19	−17	−16
0.5	−266	−185	−136	−104	−82	−67	−55	−46	−39	−34	−30	−26	−23	−21	−18	−17	−15	−14	−13	−12	−11
1	−167	−116	−85	−65	−51	−42	−34	−29	−25	−21	−19	−16	−14	−13	−12	−10	−9	−9	−8	−7	−7

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of −167E-10 (the error of $\tau = 1$ and $n = 5000$)	5000	6369	7749	10456	15175	22860
Linear regression of ln(n) over ln(τ)	$\ln(n) = -0.2717 \times \ln(\tau) + 8.5577$, $R^2 = 0.9980$					

Panels (a) and (b) illustrate the error analyses of different values of the number of abscissas, n, for contracts 13–18 (representative cases for $r < q$) and contracts 73–78 (representative cases for $r > q$) in Table 1 (Appendix A), respectively. The value of H is fixed as 60. For each set of contracts, we identify the interpolated value of n for each τ such that the errors for all τ are identical as the error given $\tau = 1$ and $n = 5000$. Next, we regress the logarithm of those interpolated values of n over the logarithm of the corresponding τ. The slope coefficients are −0.2725 and −0.2717 for each set of option contracts, respectively.

[11]The choice of this reference point is simply for convenience. With this setting, it is not necessary to derive the interpolated n for the case of $\tau = 1$. In fact, one can select any arbitrary reference point to obtain similar results.

Note that this assumption of n being proportional to $\tau^{-0.25}$ is critically important for all of the error analyses in this paper. If a different assumption is adopted, for example, assuming n is proportional to $\tau^{-0.5}$,[12] then one will allocate far more than enough nodes in the holding region when τ is short. Therefore, the pricing error for an option contract with a shorter τ will be significantly smaller than that with a longer τ. As a consequence, under the assumption that n is proportional to $\tau^{-0.5}$, it is meaningless to compare the performance of different models by comparing the RMSRE results since all RMSRE results in, for example, Table 1 and Figure 1 will be solely dominated by the comparatively large pricing errors for $\tau = 1$.

3.2 | Option values and critical stock prices when jumps are presenting

For the LJD and the LJDR process, our method possesses an incomparable advantage in evaluating PBOs. To the best of our knowledge, most studies in the literature on pricing PBOs consider only the PD processes, and there is no feasible method able to evaluate PBOs even under the LJD process. The following two subsections demonstrate the pricing results of the proposed method under the LJD process and the LJDR process.

3.2.1 | Option values and critical stock prices under the LJD processes

Table 4 presents the values of PBOs under the LJD process based by our method. The examined option contracts are adapted from Table I of Amin (1993) and Table 2 of Ju (1998). In addition to the values of perpetual Bermudan puts under the LJD

TABLE 4 Option values of perpetual Bermudan puts under the LJD process ($\lambda_2 = 0$)

Option parameters (with jump)				Benchmark option values	Our method ($H = 60$, $F = 150$)	Our method ($H = 40$, $F = 150$)	Option parameters (no jump, but with a comparably total σ)				Benchmark option values	Our method ($H = 60$, $F = 150$)	Our method ($H = 40$, $F = 150$)	(1) − (4)	(2) − (5)	(3) − (6)
(r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J)$	τ	(1)	(2)	(3)	(r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J)$	τ	(4)	(5)	(6)			
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.004	22.0872798	22.0872793	22.0872790	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	0.004	22.1779064	22.1779059	22.1779057	−0.0906267	−0.0906266	−0.0906266
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.02	22.0808649	22.0808645	22.0808642	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	0.02	22.1718430	22.1718425	22.1718423	−0.0909780	−0.0909780	−0.0909780
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.083	22.0552413	22.0552408	22.0552406	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	0.083	22.1472583	22.1472579	22.1472577	−0.0920171	−0.0920171	−0.0920171
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.25	21.9865290	21.9865286	21.9865284	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	0.25	22.0802892	22.0802888	22.0802886	−0.0937602	−0.0937602	−0.0937602
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.5	21.8809013	21.8809009	21.8809009	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	0.5	21.9760654	21.9760650	21.9760650	−0.0951641	−0.0951641	−0.0951641
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	1	21.6645250	21.6645246	21.6645248	(0.08, 0.12,0.550568)	(40, 45)	(0, 0, 0)	1	21.7634709	21.7634705	21.7634706	−0.0989459	−0.0989459	−0.0989459
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.004	18.8796137	18.8796133	18.8796131	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	0.004	18.9600175	18.9600171	18.9600169	−0.0804038	−0.0804038	−0.0804038
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.02	18.8741305	18.8741301	18.8741299	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	0.02	18.9548339	18.9548335	18.9548332	−0.0807034	−0.0807034	−0.0807034
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.083	18.8522280	18.8522277	18.8522275	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	0.083	18.9338163	18.9338159	18.9338157	−0.0815883	−0.0815883	−0.0815883
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.25	18.7934931	18.7934927	18.7934926	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	0.25	18.8765624	18.8765620	18.8765619	−0.0830693	−0.0830693	−0.0830693
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.5	18.7031720	18.7031717	18.7031717	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	0.5	18.7875120	18.7875116	18.7875117	−0.0843400	−0.0843400	−0.0843400
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	1	18.5178420	18.5178417	18.5178418	(0.08, 0.12,0.550568)	(40, 40)	(0, 0, 0)	1	18.6041832	18.6041829	18.6041830	−0.0863412	−0.0863412	−0.0863412
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.004	15.8027256	15.8027252	15.8027251	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	0.004	15.8728145	15.8728142	15.8728140	−0.0700890	−0.0700889	−0.0700889
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.02	15.7981360	15.7981356	15.7981355	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	0.02	15.8684749	15.8684746	15.8684744	−0.0703389	−0.0703389	−0.0703389
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.083	15.7798030	15.7798027	15.7798026	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	0.083	15.8508796	15.8508793	15.8508791	−0.0710765	−0.0710765	−0.0710765
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.25	15.7306397	15.7306394	15.7306393	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	0.25	15.8029476	15.8029472	15.8029471	−0.0723078	−0.0723078	−0.0723078
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.5	15.6550219	15.6550216	15.6550216	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	0.5	15.7284422	15.7284419	15.7284419	−0.0734203	−0.0734203	−0.0734203
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	1	15.4995232	15.4995230	15.4995230	(0.08, 0.12,0.550568)	(40, 35)	(0, 0, 0)	1	15.5739664	15.5739661	15.5739662	−0.0744432	−0.0744432	−0.0744432
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.004	17.7856328	17.7856325	17.7856323	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.004	17.8832227	17.8832223	17.8832221	−0.0975899	−0.0975898	−0.0975898
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.02	17.7760959	17.7760955	17.7760953	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.02	17.8744426	17.8744422	17.8744420	−0.0983467	−0.0983467	−0.0983467
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.083	17.7386304	17.7386301	17.7386299	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.083	17.8395757	17.8395753	17.8395751	−0.1009453	−0.1009452	−0.1009452
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.25	17.6414106	17.6414103	17.6414101	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.25	17.7475414	17.7475410	17.7475409	−0.1061308	−0.1061308	−0.1061307
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.5	17.4980477	17.4980474	17.4980474	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.5	17.6105100	17.6105096	17.6105097	−0.1124623	−0.1124623	−0.1124623
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	1	17.2137504	17.2137501	17.2137502	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	1	17.3412537	17.3412534	17.3412535	−0.1275033	−0.1275033	−0.1275033
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.004	14.8547372	14.8547369	14.8547368	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.004	14.9380246	14.9380242	14.9380240	−0.0832874	−0.0832873	−0.0832873
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.02	14.8467720	14.8467717	14.8467715	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.02	14.9306904	14.9306901	14.9306899	−0.0839185	−0.0839184	−0.0839184
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.083	14.8154813	14.8154810	14.8154809	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.083	14.9015658	14.9015655	14.9015653	−0.0860845	−0.0860844	−0.0860844
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.25	14.7342881	14.7342878	14.7342877	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.25	14.8247165	14.8247162	14.8247161	−0.0904284	−0.0904284	−0.0904284
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.5	14.6146457	14.6146454	14.6146454	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.5	14.7096261	14.7096258	14.7096258	−0.0949804	−0.0949804	−0.0949804
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	1	14.3799650	14.3799648	14.3799648	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	1	14.4858361	14.4858359	14.4858360	−0.1058711	−0.1058711	−0.1058711
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.004	12.1116704	12.1116702	12.1116701	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.004	12.1812271	12.1812268	12.1812267	−0.0695567	−0.0695566	−0.0695566
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.02	12.1051761	12.1051759	12.1051757	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.02	12.1752465	12.1752462	12.1752461	−0.0700704	−0.0700703	−0.0700703
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.083	12.0796638	12.0796636	12.0796635	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.083	12.1514968	12.1514965	12.1514964	−0.0718329	−0.0718329	−0.0718329
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.25	12.0134612	12.0134609	12.0134609	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.25	12.0888380	12.0888378	12.0888377	−0.0753768	−0.0753768	−0.0753768
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.5	11.9158866	11.9158863	11.9158864	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.5	11.9948034	11.9948032	11.9948032	−0.0789168	−0.0789168	−0.0789168
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	1	11.7253346	11.7253344	11.7253345	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	1	11.8112202	11.8112200	11.8112201	−0.0858856	−0.0858856	−0.0858856
RMSRE vs. Benchmark (all)					0.0000019%	0.0000026%						0.0000020%	0.0000027%			
RMSRE vs. Benchmark ($r \leq q$)					0.0000020%	0.0000026%						0.0000020%	0.0000027%			
RMSRE vs. Benchmark ($r > q$)					0.0000019%	0.0000025%						0.0000020%	0.0000027%			
Time (sec.)					339,218	174,643						39,475	20,578			

This table reports the values of perpetual Bermudan puts based on our method under the LJD process in Merton (1976). The option parameters are adapted from Table 2 of Ju (1998) and Table I of Amin (1993). We compute the prices of perpetual Bermudan puts at the exercisable time point, that is, $T = 0$ in Equation (15). In addition to the values of perpetual Bermudan puts under the LJD process reported in columns 1–7, we also report the prices of perpetual Bermudan puts under the PD process with a comparably expected total variance in the lognormal jump-diffusion case for comparison in columns 8–14. The values of comparably expected total variance are derived via $\sigma^2 + \lambda_1 (\mu_J^2 + \sigma_J^2)$ as suggested in Amin (1993).

[12]Note that Andricopoulos et al. (2003) suggest that n can be determined in proportion to $\tau^{-0.5}$ in their universal option pricing model based on the Simpson quadrature method.

process reported in columns 1–7, the values of perpetual Bermudan puts under the PD process with comparably expected total variances[13] are also listed for comparison in columns 8–14.

We first note that the proposed method with $(H,F) = (60,150)$ or $(40,150)$ can generate fairly accurate option prices. Here, the RMSREs versus the benchmark option values are 0.0000019% and 0.0000026% for $(H,F) = (60,150)$ and $(40,150)$, respectively.[14] The RMSREs exhibit similar magnitudes for the LJD process and the corresponding PD process if we control the expected total variance. By comparing the option prices under the LJD process and the corresponding PD process, the option prices with jumps are lower than those without jumps by 0.09 dollars on average, which represents about 0.5% of the option prices with jumps. This phenomenon is consistent with the results in Table I of Amin (1993) where the American put prices under the PD process with comparably expected total variances are more expensive than those under the corresponding LJD process when the time to maturity is long.

We further employ the same option contracts in Table 4 to conduct accuracy and speed analyses for different combinations of $H(=40,60)$ and $F(=100,150,200)$ in Figure 2. The results are consistent with our expectation that with an increase of H and F, more computational time is required and the option values will further converge to the benchmark. By comparing to Figure 1, it is worth noting that the convergence pattern of option prices generated by our method are alike under either the PD process or the LJD process. Nevertheless, in order to obtain comparable accuracy levels based on our method under the pure diffusion and the LJD processes, more time is unavoidably consumed when the underlying asset price is posited to follow the LJD process.

For the same option contracts in Table 4, Table 5 reports the critical stock prices of perpetual Bermudan puts generated by the proposed approach under the LJD process. Note first that our method with either $(H,F) = (60,150)$ or $(40,150)$ can generate accurate estimations for the critical stock prices of perpetual Bermudan puts. Here, the RMSREs versus the benchmark critical stock prices are 0.0001094% and 0.0001640% for our method with $(H,F) = (60,150)$ or $(40,150)$, respectively. Moreover, the critical stock prices under the LJD process are higher than those under the corresponding PD process. This phenomenon is reasonable, because, as shown in Table 4, the perpetual Bermudan puts are cheaper under the LJD process than under the PD process, the option holders is apt to exercise the perpetual Bermudan puts earlier, and thus the critical stock prices should be higher. The detailed explanation of this phenomenon can refer to Figure 5 of Amin (1993).

Similar to the PD process case in Section 3.1, we intend to identify the relation between the number of abscissas in the holding region, n, and the time interval between two consecutive exercisable time points, τ, under the LJD process

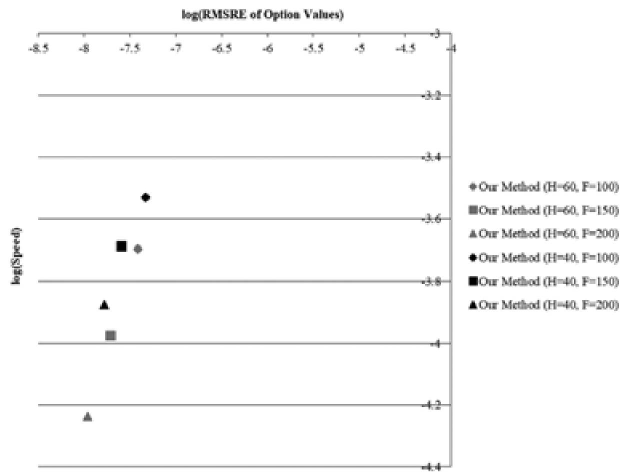

FIGURE 2 Speed-Accuracy Analysis for our Method with Different H and F under the LJD process: Figure 2 compares the accuracy and speed of our method with different H and F to price perpetual Bermudan options under the LJD process. We employ the speed (number of options priced per second) and the root mean squared relative errors (RMSREs) of our method (with different combinations of $H = 40,60$ and $F = 100,150,200$) for the option contracts in Table 4 to plot this figure

[13]We follow the suggestion in Amin (1993) to derive the values of comparably expected total variance via $\sigma^2 + \lambda_1\left(\mu_J^2 + \sigma_J^2\right)$.

[14]The reason why we suggest $F = 150$ (rather than $F = 300$) for the LJD process is that according to our preliminary tests, $F = 150$ is sufficient to generate option values under the LJD process as accurately as those under the PD process in Table 1.

TABLE 5 Critical stock prices of perpetual Bermudan puts under the LJD process ($\lambda_2 = 0$)

Option parameters (with jump) (r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J)$	τ	Benchmark critical stock prices (1)	Our method ($H = 60$, $F = 150$) (2)	Our method ($H = 40$, $F = 150$) (3)	Option parameters (no jump, but with a comparably total σ) (r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J)$	τ	Benchmark critical stock prices (4)	Our method ($H = 60$, $F = 150$) (5)	Our method ($H = 40$, $F = 150$) (6)	(1) − (4)	(2) − (5)	(3) − (6)
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.004	12.1649318	12.1649702	12.1649894	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	0.004	11.4195061	11.4195229	11.4195314	0.7454257	0.7454743	0.7454473
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.02	12.3321015	12.3321158	12.3321230	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	0.02	11.7111991	11.7112063	11.7112100	0.6209024	0.6209183	0.6209095
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.083	12.7293819	12.7293870	12.7293896	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	0.083	12.2840821	12.2840854	12.2840870	0.4452998	0.4453039	0.4453016
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.25	13.4557415	13.4557436	13.4557443	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	0.25	13.1607664	13.1607681	13.1607686	0.2949751	0.2949759	0.2949756
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.5	14.3016392	14.3016404	14.3016403	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	0.5	14.0834111	14.0834122	14.0834121	0.2182281	0.2182282	0.2182282
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	1	15.6595497	15.6595504	15.6595501	(0.08, 0.12, 0.550568)	(40, 45)	(0, 0, 0)	1	15.4963320	15.4963326	15.4963324	0.1632178	0.1632178	0.1632178
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.004	10.8132727	10.8133068	10.8133239	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	0.004	10.1506721	10.1506871	10.1506945	0.6626006	0.6626438	0.6626198
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.02	10.9618680	10.9618807	10.9618871	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	0.02	10.4099548	10.4099612	10.4099644	0.5519133	0.5519274	0.5519196
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.083	11.3150061	11.3150107	11.3150130	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	0.083	10.9191841	10.9191870	10.9191885	0.3958220	0.3958257	0.3958237
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.25	11.9606592	11.9606610	11.9606616	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	0.25	11.6984590	11.6984605	11.6984610	0.2622001	0.2622008	0.2622005
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.5	12.7125682	12.7125692	12.7125692	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	0.5	12.5185877	12.5185886	12.5185886	0.1939805	0.1939806	0.1939806
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	1	13.9195997	13.9196003	13.9196001	(0.08, 0.12, 0.550568)	(40, 40)	(0, 0, 0)	1	13.7745173	13.7745178	13.7745177	0.1450825	0.1450825	0.1450825
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.004	9.4616136	9.4616435	9.4616584	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	0.004	8.8818381	8.8818512	8.8818577	0.5797755	0.5798133	0.5797923
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.02	9.5916345	9.5916456	9.5916512	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	0.02	9.1087104	9.1087160	9.1087189	0.4829241	0.4829365	0.4829296
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.083	9.9006304	9.9006344	9.9006363	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	0.083	9.5542861	9.5542886	9.5542899	0.3463443	0.3463475	0.3463457
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.25	10.4655768	10.4655784	10.4655789	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	0.25	10.2361516	10.2361530	10.2361534	0.2294251	0.2294257	0.2294254
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.5	11.1234972	11.1234981	11.1234980	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	0.5	10.9537642	10.9537651	10.9537650	0.1697329	0.1697331	0.1697330
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	1	12.1796498	12.1796503	12.1796501	(0.08, 0.12, 0.550568)	(40, 35)	(0, 0, 0)	1	12.0527026	12.0527031	12.0527030	0.1269471	0.1269472	0.1269472
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.004	16.9384168	16.9384432	16.9384564	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.004	15.8638895	15.8639024	15.8639088	1.0745273	1.0745577	1.0745408
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.02	17.1630473	17.1630573	17.1630624	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.02	16.2614316	16.2614372	16.2614400	0.9016157	0.9016257	0.9016202
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.083	17.6833279	17.6833316	17.6833335	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.083	17.0256767	17.0256793	17.0256806	0.6576512	0.6576538	0.6576523
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.25	18.6047605	18.6047622	18.6047627	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.25	18.1557514	18.1557528	18.1557532	0.4490091	0.4490096	0.4490094
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	0.5	19.6409493	19.6409503	19.6409502	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	0.5	19.2980854	19.2980863	19.2980862	0.3428639	0.3428640	0.3428640
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607)	1	21.2313168	21.2313174	21.2313172	(0.08, 0, 0.550568)	(40, 45)	(0, 0, 0)	1	20.9642529	20.9642535	20.9642533	0.2670639	0.2670639	0.2670639
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.004	15.0563705	15.0563940	15.0564057	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.004	14.1012351	14.1012466	14.1012523	0.9551354	0.9551624	0.9551474
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.02	15.2560420	15.2560510	15.2560555	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.02	14.4546058	14.4546108	14.4546133	0.8014362	0.8014451	0.8014401
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.083	15.7185137	15.7185170	15.7185187	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.083	15.1339349	15.1339372	15.1339383	0.5845788	0.5845811	0.5845798
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.25	16.5375649	16.5375664	16.5375668	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.25	16.1384457	16.1384469	16.1384473	0.3991192	0.3991197	0.3991195
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	0.5	17.4586216	17.4586225	17.4586224	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	0.5	17.1538537	17.1538545	17.1538544	0.3047679	0.3047680	0.3047680
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607)	1	18.8722816	18.8722821	18.8722819	(0.08, 0, 0.550568)	(40, 40)	(0, 0, 0)	1	18.6348915	18.6348920	18.6348918	0.2373901	0.2373901	0.2373901
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.004	13.1743242	13.1743447	13.1743550	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.004	12.3385807	12.3385907	12.3385958	0.8357435	0.8357671	0.8357540
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.02	13.3490368	13.3490446	13.3490485	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.02	12.6477801	12.6477845	12.6477867	0.7012566	0.7012645	0.7012601
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.083	13.7536994	13.7537024	13.7537039	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.083	13.2421930	13.2421950	13.2421960	0.5115065	0.5115085	0.5115074
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.25	14.4703693	14.4703706	14.4703710	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.25	14.1211400	14.1211411	14.1211414	0.3492293	0.3492297	0.3492295
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	0.5	15.2762939	15.2762947	15.2762946	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	0.5	15.0096220	15.0096227	15.0096226	0.2666720	0.2666720	0.2666720
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607)	1	16.5132464	16.5132468	16.5132467	(0.08, 0, 0.550568)	(40, 35)	(0, 0, 0)	1	16.3055300	16.3055305	16.3055303	0.2077164	0.2077164	0.2077164
RMSRE vs. Benchmark (all)					0.0001094%	0.0001640%						0.0000538%	0.0000805%			
RMSRE vs. Benchmark ($r \le q$)					0.0001385%	0.0002077%						0.0000665%	0.0000996%			
RMSRE vs. Benchmark ($r > q$)					0.0000687%	0.0001030%						0.0000369%	0.0000552%			
Time (sec.)					339,218	174,643						39,475	20,578			

Table 5 reports the critical stock prices of perpetual Bermudan puts generated by our method under Merton's (1976) LJD process and the corresponding PD process with a comparably expected total variance. All option parameters are the same as those in Table 4. The critical stock prices under the LJD process generated from our method are shown in columns 5 and 7. The pure diffusion counterparts with the expected total variance of $\sigma^2 + \lambda_1(\mu_J^2 + \sigma_J^2)$ are reported in columns 12 and 14 for comparison.

TABLE 6 Error analyses and linear regression of 1n(n) over 1n(τ) under the LJD process

(a) Contracts 7–12 in Table 4 ($r = 0.08$, $q = 0.12$, $\sigma = 0.223607$, $S_0 = 40$, $X = 40$, $\lambda_1 = 5$, $\mu_J = -0.025$, $\sigma_J = 0.223607$)

τ	Benchmark option values	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	18.87961371	18.87960726	18.87960923	18.87961043	18.87961120	18.87961173	18.87961211	18.87961239	18.87961260	18.87961277	18.87961290	18.87961300	18.87961309	18.87961316	18.87961322	18.87961327	18.87961332	18.87961335	18.87961339	18.87961341	18.87961344	18.87961346
0.02	18.87413050	18.87412767	18.87412854	18.87412906	18.87412939	18.87412963	18.87412979	18.87412992	18.87413001	18.87413008	18.87413014	18.87413019	18.87413022	18.87413026	18.87413028	18.87413030	18.87413032	18.87413034	18.87413035	18.87413037	18.87413038	18.87413039
0.083	18.85222804	18.85222671	18.85222712	18.85222736	18.85222752	18.85222763	18.85222771	18.85222777	18.85222781	18.85222785	18.85222787	18.85222789	18.85222791	18.85222793	18.85222794	18.85222795	18.85222796	18.85222797	18.85222797	18.85222798	18.85222798	18.85222799
0.25	18.79349309	18.79349237	18.79349259	18.79349272	18.79349281	18.79349286	18.79349291	18.79349294	18.79349296	18.79349298	18.79349299	18.79349301	18.79349302	18.79349302	18.79349303	18.79349304	18.79349304	18.79349305	18.79349305	18.79349305	18.79349305	18.79349306
0.5	18.70317199	18.70317152	18.70317166	18.70317175	18.70317181	18.70317184	18.70317187	18.70317189	18.70317191	18.70317192	18.70317193	18.70317194	18.70317194	18.70317195	18.70317195	18.70317196	18.70317196	18.70317196	18.70317197	18.70317197	18.70317197	18.70317197
1	18.51784203	18.51784173	18.51784182	18.51784188	18.51784192	18.51784194	18.51784196	18.51784197	18.51784198	18.51784199	18.51784200	18.51784200	18.51784201	18.51784201	18.51784201	18.51784201	18.51784202	18.51784202	18.51784202	18.51784202	18.51784202	18.51784202

Relative error vs. Benchmark (x1.0E-10)

τ	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	−3420	−2372	−1741	−1332	−1052	−851	−703	−590	−502	−432	−376	−330	−292	−260	−233	−210	−190	−173	−158	−145	−133
0.02	−1498	−1040	−764	−585	−462	−374	−309	−260	−221	−191	−166	−146	−129	−115	−104	−93	−85	−77	−71	−65	−60
0.083	−705	−489	−360	−275	−218	−176	−146	−122	−104	−90	−78	−69	−61	−54	−49	−44	−40	−36	−33	−31	−28
0.25	−382	−265	−195	−149	−118	−96	−79	−66	−57	−49	−42	−37	−33	−29	−26	−24	−22	−20	−18	−17	−15
0.5	−253	−176	−129	−99	−78	−63	−52	−44	−37	−32	−28	−25	−22	−20	−18	−16	−14	−13	−12	−11	−10
1	−163	−113	−83	−64	−50	−41	−34	−28	−24	−21	−18	−16	−14	−13	−11	−10	−9	−8	−8	−7	−7

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of –163E-10 (the error of $\tau = 1$ and $n = 5000$)	5000	6270	7691	10420	15145	22643
Linear Regression of ln(n) over ln(τ)	$\ln(n) = -0.2717 \times \ln(\tau) + 8.5515$, $R^2 = 0.9982$					

(b) Contracts 25-30 in Table 4 ($r = 0.08$, $q = 0$, $\sigma = 0.223607$, $S_0 = 40$, $X = 40$, $\lambda_1 = 5$, $\mu_J = -0.025$, $\sigma_J = 0.223607$)

τ	Benchmark Option Values	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	14.85473720	14.85473263	14.85473403	14.85473487	14.85473542	14.85473579	14.85473606	14.85473626	14.85473641	14.85473652	14.85473662	14.85473669	14.85473675	14.85473680	14.85473685	14.85473688	14.85473691	14.85473694	14.85473696	14.85473698	14.85473700	14.85473702
0.02	14.84677196	14.84676991	14.84677054	14.84677092	14.84677116	14.84677133	14.84677145	14.84677154	14.84677160	14.84677166	14.84677170	14.84677173	14.84677176	14.84677178	14.84677180	14.84677182	14.84677183	14.84677184	14.84677185	14.84677186	14.84677187	14.84677188
0.083	14.81548132	14.81548033	14.81548063	14.81548081	14.81548093	14.81548101	14.81548107	14.81548112	14.81548115	14.81548117	14.81548119	14.81548121	14.81548122	14.81548124	14.81548124	14.81548125	14.81548126	14.81548126	14.81548127	14.81548127	14.81548128	14.81548128
0.25	14.73428808	14.73428753	14.73428770	14.73428780	14.73428787	14.73428791	14.73428795	14.73428797	14.73428799	14.73428800	14.73428801	14.73428802	14.73428803	14.73428804	14.73428804	14.73428805	14.73428805	14.73428805	14.73428806	14.73428806	14.73428806	14.73428806
0.5	14.61464568	14.61464531	14.61464542	14.61464549	14.61464554	14.61464557	14.61464559	14.61464560	14.61464562	14.61464563	14.61464563	14.61464564	14.61464565	14.61464565	14.61464565	14.61464566	14.61464566	14.61464566	14.61464566	14.61464566	14.61464567	14.61464567
1	14.37996500	14.37996476	14.37996483	14.37996488	14.37996491	14.37996493	14.37996494	14.37996495	14.37996496	14.37996497	14.37996497	14.37996497	14.37996498	14.37996498	14.37996498	14.37996498	14.37996499	14.37996499	14.37996499	14.37996499	14.37996499	14.37996499

Relative error vs. Benchmark (x1.0E-10)

τ	$n = 5000$	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	$n = 12000$	$n = 13000$	$n = 14000$	$n = 15000$	$n = 16000$	$n = 17000$	$n = 18000$	$n = 19000$	$n = 20000$	$n = 21000$	$n = 22000$	$n = 23000$	$n = 24000$	$n = 25000$
0.004	−3074	−2133	−1566	−1199	−947	−766	−633	−532	−453	−390	−339	−298	−264	−235	−211	−190	−172	−157	−143	−131	−121
0.02	−1376	−955	−702	−537	−425	−344	−284	−239	−203	−175	−153	−134	−119	−106	−95	−86	−78	−71	−65	−60	−55
0.083	−671	−466	−342	−262	−207	−168	−139	−116	−99	−86	−75	−65	−58	−52	−46	−42	−38	−35	−32	−29	−27
0.25	−378	−262	−193	−148	−117	−94	−78	−66	−56	−48	−42	−37	−33	−29	−26	−24	−21	−20	−18	−16	−15
0.5	−257	−178	−131	−100	−79	−64	−53	−45	−38	−33	−29	−25	−22	−20	−18	−16	−15	−13	−12	−11	−10
1	−167	−116	−85	−65	−52	−42	−35	−29	−25	−21	−19	−16	−14	−13	−12	−10	−9	−9	−8	−7	−7

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of –167E-10 (the error of $\tau = 1$ and $n = 5000$)	5000	6236	7566	10017	14363	21328
Linear Regression of ln(n) over ln(τ)	$\ln(n) = -0.2599 \times \ln(\tau) + 8.5500$, $R^2 = 0.9985$					

Panels (a) and (b) of Table 6 present the error analyses of different values of the number of abscissas, n, for contracts 7–12 (representative cases for $r < q$) and contracts 25–30 (representative cases for $r > q$) in Table 4, respectively. The value of H is fixed as 60. For each set of contracts, we identify the interpolated value of n for each τ such that the errors for all τ are identical as the error given $\tau = 1$ and $n = 5000$. Next, we regress the logarithm of those interpolated values of n over the logarithm of the corresponding τ. The slope coefficients are −0.2717 and −0.2599 for each set of contracts. The results are similar to those under the PD process in Table 3.

TABLE 7 Option values of perpetual Bermudan puts under the LJDR process

Option parameters (with jump and default)						
(r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J, \lambda_2)$	τ	Benchmark option values	Our method ($H = 60$, $F = 150$)	Our method ($H = 40$, $F = 150$)
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.004	25.2819819	25.2819812	25.2819808
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.02	25.2547308	25.2547302	25.2547299
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.083	25.1483683	25.1483678	25.1483675
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.25	24.8754464	24.8754459	24.8754458
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.5	24.4789000	24.4788996	24.4788996
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	1	23.7197658	23.7197654	23.7197655
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.004	22.0119532	22.0119526	22.0119523
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.02	21.9886646	21.9886641	21.9886639
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.083	21.8977337	21.8977333	21.8977331
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.25	21.6642499	21.6642495	21.6642494
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.5	21.3247010	21.3247007	21.3247007
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	1	20.6732840	20.6732837	20.6732838
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.004	18.8348451	18.8348446	18.8348444
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.02	18.8153306	18.8153302	18.8153300
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.083	18.7391029	18.7391025	18.7391023
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.25	18.5432203	18.5432200	18.5432200
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.5	18.2580729	18.2580727	18.2580727
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	1	17.7099960	17.7099958	17.7099959
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.004	22.8893622	22.8893618	22.8893616
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.02	22.8651958	22.8651954	22.8651952
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.083	22.7700304	22.7700301	22.7700299
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.25	22.5236245	22.5236242	22.5236241
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.5	22.1641805	22.1641802	22.1641803
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	1	21.4764946	21.4764943	21.4764944
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.004	19.8718984	19.8718981	19.8718980
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.02	19.8515287	19.8515284	19.8515283

(Continues)

TABLE 7 (Continued)

Option parameters (with jump and default) (r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J, \lambda_2)$	τ	Benchmark option values	Our method ($H = 60$, $F = 150$)	Our method ($H = 40$, $F = 150$)
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.083	19.7712969	19.7712967	19.7712965
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.25	19.5634080	19.5634078	19.5634077
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.5	19.2597307	19.2597305	19.2597305
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	1	18.6773194	18.6773192	18.6773193
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.004	16.9652399	16.9652396	16.9652395
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.02	16.9484072	16.9484070	16.9484069
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.083	16.8820903	16.8820901	16.8820899
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.25	16.7101176	16.7101174	16.7101173
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.5	16.4585324	16.4585322	16.4585322
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	1	15.9746582	15.9746580	15.9746581
RMSRE vs. Benchmark (all)					0.0000016%	0.0000023%
RMSRE vs. Benchmark ($r \le q$)					0.0000019%	0.0000027%
RMSRE vs. Benchmark ($r > q$)					0.0000013%	0.0000018%
Time (sec.)					240,643	130,791

This table reports the values of perpetual Bermudan puts based on our method under the LJDR process. The option parameters (except λ_2) are adapted from Table 2 of Ju (1998) and Table I of Amin (1993). We compute the prices of perpetual Bermudan puts at the exercisable time point, that is, $T = 0$ in Equation (15). The method described in Appendix B is employed to derive the benchmark option values.

by controlling pricing errors. We take the 7–12th contracts (representative cases for $r < q$) and the 25–30th contracts (representative cases for $r > q$) in Table 4 as examples to conduct the same regression analysis as that in Table 3. Table 6 shows that the regression results are $1\text{n}(n) = -0.2717 \times 1\text{n}(\tau) + 8.5515$ with $R^2 = 0.9982$ for $r < q$ and $1\text{n}(n) = -0.2599 \times 1\text{n}(\tau) + 8.5500$ with $R^2 = 0.9985$ for $r > q$, respectively. Since the regression coefficients in front of $1\text{n}(\tau)$ are around -0.25 for all parameter sets, this suggests that n should be in proportion to $\tau^{-0.25}$ when pricing PBOs under the LJD process.

3.2.2 | Option values and critical stock prices under the LJDR processes

For the comparison purpose, this subsection reexamines the option contracts with the lognormal jumps in Table 4 by further considering a nonzero value of λ_2. The default intensity λ_2 is assumed to be 0.05, which means the average time to encounter the default event is around 20 years. The numerical results under the LJDR process are shown in Table 7. The RMSREs of our method with either $(H,F) = (60,150)$ or $(40,150)$ are extremely small, which are 0.0000016% and 0.0000023%, respectively. By comparing Tables 4 and 7, it can be found first that PBOs become more valuable with the presence of default. Under $\lambda_2 = 0.05$, the option values increase by 24% on average. The significant rise in option values implies the importance to consider the possibility of default. These results can be expected since we assume that in the default event, holders of perpetual Bermudan puts can receive $(X - S_{t+\tau})^+ = X$ due to $S_{t+\tau} = 0$. Moreover, the influence of default risk is relatively minor for option contracts with $r < q$ (option values increasing by 15%) and more significantly for option contracts with $r > q$ (option values increasing by 33%). The reason behind this phenomenon is because when $r < q$ ($r > q$), the drift term of the stock price process is inclined to be negative (positive) and the possibility to meet a very low stock price in a long run is relatively high (low). Consequently, the marginal benefit of introducing the event of default (to increase the payoff due to the occurrence of zero stock price) is less pronounced for the cases of $r < q$. Last, one can observe that even with the presence of default, the pricing errors of our method shown in Table 7 are of similar magnitude with those in Table 4. In the meanwhile, it costs less computational time under the LJDR process. One possible explanation for this improvement in convergence rate under the LJDR process is due to assuming EVGD(S_t,τ), part of the Bermudan option value, as a constant.

In addition, Figure 3 conducts the accuracy and speed analysis for the option contracts in Table 7. The results with default in Figure 3 are very similar to the results in Figure 2. By taking all of the results in Figures 1–3 into consideration, we conclude that the accuracy performance of our method is consistent for the PD, the LJD, the LJDR processes, although more computational time is needed when the jumps are considered.

Table 8 presents the critical stock prices of the option contracts in Table 7 under the LJDR process. It can be found that our method with either (H,F) = (60,150) or (40,150) can generate accurate critical stock prices of perpetual Bermudan puts. We find that the RMSREs in Table 8 are very similar to those in Table 5. Moreover, the critical stock prices under the LJDR process in Table 8 are lower than those under the LJD process in Table 5. This phenomenon is because, as suggested from the first component in Equation (4), the holding values of perpetual Bermudan puts would be larger due to the default event and thus less likely to be early exercised under the LJDR model.

For the LJDR process, we are also interested in the relation between the number of abscissas in the holding region, n, and the time interval between two consecutive exercisable time points, τ. Following the same methodology to generate Tables 3 and 6, we take the 7–12th contracts (representative cases for $r < q$) and the 25–30th contracts (representative cases for $r > q$) in Table 7 as examples to produce Table 9. Table 9 shows that the regression results for these two sets of parameters are $\ln(n) = -0.3115 \times \ln(\tau) + 8.5438$ with $R^2 = 0.9988$ for $r < q$ and $\ln(n) = -0.2722 \times \ln(\tau) + 8.5398$ with $R^2 = 0.9993$ for $r > q$, respectively. By observing all the results in Tables 3, 6, and 9, the regression coefficients in front of $\ln(\tau)$ are always near -0.25 regardless of considering the PD, the LJD, or the LJDR processes. These results attest the correctness of our method to assume that n is proportional to $\tau^{-0.25}$ in Equation (7).

3.2.3 | Perpetual American put values under the LJD and the LJDR processes

We lastly perform quadratic regression analyses of the option values over τ (just like what we do in Table 2) under the LJD and the LJDR processes in Table 10. The intercept coefficients can be interpreted as option values of PBOs when τ approaches zero,

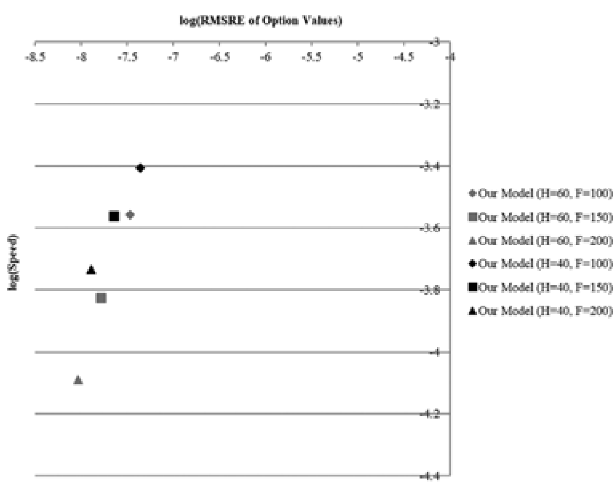

FIGURE 3 Speed-Accuracy Analysis for our Method with Different H and F under the LJDR process: Figure 3 compares the accuracy and speed of our method with different combinations of $H = 40{,}60$ and $F = 100{,}150{,}200$ to price perpetual Bermudan options under the LJDR process. The speed is measured by the number of options priced per second, and the accuracy is measured by the root mean squared relative errors (RMSREs). This figure is generated based on the pricing results for the option contracts with nonzero λ_1 and λ_2 in Table 7

TABLE 8 Critical stock prices of perpetual Bermudan puts under the LJDR process

Option parameters (with Jump and Default)						
(r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J, \lambda_2)$	τ	Benchmark Critical Stock Prices	Our Method ($H = 60$, $F = 150$)	Our Method ($H = 40$, $F = 150$)
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.004	11.1052644	11.1052966	11.1053128
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.02	11.4758796	11.4758916	11.4758975
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.083	12.2404035	12.2404077	12.2404099
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.25	13.4616476	13.4616493	13.4616498
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.5	14.7586445	14.7586454	14.7586453
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	1	16.6969805	16.6969810	16.6969808
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.004	9.8713461	9.8713748	9.8713891
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.02	10.2007819	10.2007925	10.2007978
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.083	10.8803587	10.8803624	10.8803643
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.25	11.9659090	11.9659105	11.9659110
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.5	13.1187951	13.1187959	13.1187959
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	1	14.8417604	14.8417609	14.8417607
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.004	8.6374278	8.6374529	8.6374655
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.02	8.9256841	8.9256934	8.9256981
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.083	9.5203138	9.5203171	9.5203188
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.25	10.4701704	10.4701717	10.4701721
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.5	11.4789457	11.4789464	11.4789464
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	1	12.9865404	12.9865408	12.9865406
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.004	14.0163652	14.0163881	14.0163996
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.02	14.3624886	14.3624973	14.3625017
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.083	15.1030955	15.1030988	15.1031004
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.25	16.3188729	16.3188743	16.3188747
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	0.5	17.6127668	17.6127676	17.6127676
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	1	19.5126806	19.5126810	19.5126809
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.004	12.4589913	12.4590117	12.4590219
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.02	12.7666565	12.7666643	12.7666682

(Continues)

TABLE 8 (Continued)

Option parameters (with Jump and Default)						
(r, q, σ)	(S_0, X)	$(\lambda_1, \mu_J, \sigma_J, \lambda_2)$	τ	Benchmark Critical Stock Prices	Our Method ($H = 60$, $F = 150$)	Our Method ($H = 40$, $F = 150$)
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.083	13.4249738	13.4249767	13.4249781
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.25	14.5056648	14.5056660	14.5056664
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	0.5	15.6557927	15.6557935	15.6557934
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	1	17.3446049	17.3446053	17.3446052
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.004	10.9016174	10.9016352	10.9016442
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.02	11.1708245	11.1708313	11.1708347
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.083	11.7468521	11.7468546	11.7468559
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.25	12.6924567	12.6924578	12.6924581
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	0.5	13.6988186	13.6988193	13.6988192
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	1	15.1765293	15.1765297	15.1765296
RMSRE vs. Benchmark (all)					0.0001032%	0.0001548%
RMSRE vs. Benchmark ($r \le q$)					0.0001270%	0.0001904%
RMSRE vs. Benchmark ($r > q$)					0.0000720%	0.0001080%
Time (s)					240,643	130,791

Table 8 reports the critical stock prices of perpetual Bermudan puts based on our method under the LJDR process. The option parameters (except λ_2) are adapted from Table 2 of Ju (1998) and Table I of Amin (1993). The benchmark critical stock prices are computed using the detailed procedure described in Appendix B.

which should theoretically be the perpetual American option prices. Note that this novel way to apply our method is highly important, because to the best of our knowledge, there is no literature available to compute the option value of a perpetual American option under Merton's (1976) LJD process and the LJDR process. In Table 10, it is first observed that the R^2 values are extremely high, which demonstrates the exact convergence of perpetual Bermudan put prices to perpetual American put prices under the LJD and the LJDR processes. Moreover, the differences between the intercept coefficients based on the benchmark option values and those of our method (with $(H,F) = (40,150)$ or $(60,150)$) are minor and within 1.0E-6. Thus, the approximated perpetual American put prices based on our method can be expected to converge accurately. Recall that under the PD process in Table 2, our regression-based approximations for perpetual American put prices under the PD process are highly accurate, with RMSREs smaller than 0.0075%. These results together strengthen our confidence on the accuracy of our regression-based approximations for perpetual American put prices under the LJD and the LJDR processes.

4 | CONCLUSION

This paper proposes a simple yet efficient and accurate method for pricing PBOs under the LJDR processes. Under the degenerated case of the PD process, the accuracy and efficiency of our method is verified by comparing with the finite difference method and the MBA method proposed by Muthuraman (2008). Under the LJD and the LJDR processes, the proposed method is the first feasible one that is able to value PBOs. Our method retains excellent performance in accuracy under the LJD and the

TABLE 9 Error Analyses and Linear Regression of 1n(n) over 1n(τ) under the LJDR process

(a) Contracts 7–12 in Table 7 (r = 0.08, q = 0.12, σ = 0.223607, S_0 = 40, X = 40, λ_1 = 5, μ_J = -0.025, σ_J = 0.223607, λ_2 = 0.05)

τ	Benchmark Option Values	n = 5000	n = 6000	n = 7000	n = 8000	n = 9000	n = 10000	n = 11000	n = 12000	n = 13000	n = 14000	n = 15000	n = 16000	n = 17000	n = 18000	n = 19000	n = 20000	n = 21000		n = 26000	n = 27000	n = 28000
0.004	22.01195316	22.01194430	22.01194701	22.01194864	22.01194970	22.01195043	22.01195095	22.01195133	22.01195163	22.01195185	22.01195203	22.01195218	22.01195230	22.01195240	22.01195248	22.01195255	22.01195261	22.01195266		22.01195275	22.01195278	22.01195281
0.02	21.98866463	21.98866108	21.98866216	21.98866282	21.98866324	21.98866353	21.98866374	21.98866389	21.98866401	21.98866410	21.98866417	21.98866423	21.98866428	21.98866432	21.98866435	21.98866438	21.98866441	21.98866443		21.98866446	21.98866447	21.98866449
0.083	21.89773372	21.89773226	21.89773271	21.89773298	21.89773315	21.89773327	21.89773336	21.89773342	21.89773347	21.89773351	21.89773354	21.89773356	21.89773358	21.89773360	21.89773361	21.89773362	21.89773363	21.89773364		21.89773365	21.89773366	21.89773366
0.25	21.66424990	21.66424920	21.66424941	21.66424954	21.66424962	21.66424968	21.66424972	21.66424975	21.66424978	21.66424980	21.66424981	21.66424982	21.66424983	21.66424984	21.66424985	21.66424985	21.66424986	21.66424986	…	21.66424987	21.66424987	21.66424987
0.5	21.32470100	21.32470056	21.32470070	21.32470078	21.32470083	21.32470087	21.32470089	21.32470091	21.32470093	21.32470094	21.32470095	21.32470095	21.32470096	21.32470096	21.32470097	21.32470097	21.32470097	21.32470098		21.32470098	21.32470098	21.32470098
1	20.67328400	20.67328374	20.67328382	20.67328387	20.67328390	20.67328392	20.67328394	20.67328395	20.67328396	20.67328396	20.67328397	20.67328397	20.67328398	20.67328398	20.67328398	20.67328398	20.67328399	20.67328399		20.67328399	20.67328399	20.67328399

Relative error vs. Benchmark (x1.0E-10)

τ	n = 5000	n = 6000	n = 7000	n = 8000	n = 9000	n = 10000	n = 11000	n = 12000	n = 13000	n = 14000	n = 15000	n = 16000	n = 17000	n = 18000	n = 19000	n = 20000	n = 21000		n = 26000	n = 27000	n = 28000
0.004	−4027	−2795	−2053	−1571	−1241	−1004	−830	−697	−593	−511	−445	−391	−346	−309	−277	−250	−226		−147	−136	−127
0.02	−1613	−1120	−823	−630	−498	−403	−333	−280	−239	−206	−179	−157	−139	−124	−112	−101	−91		−60	−55	−52
0.083	−667	−463	−340	−261	−206	−167	−138	−116	−99	−85	−74	−65	−58	−51	−46	−42	−38		−25	−23	−21
0.25	−325	−226	−166	−127	−100	−81	−67	−56	−48	−42	−36	−32	−28	−25	−23	−20	−18	…	−12	−11	−10
0.5	−206	−143	−105	−80	−63	−51	−42	−36	−30	−26	−23	−20	−18	−16	−14	−13	−12		−8	−7	−7
1	−128	−89	−65	−50	−40	−32	−27	−22	−19	−16	−14	−13	−11	−10	−9	−8	−7		−5	−4	−4

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of −128E-10 (the error of τ = 1 and n = 5000)	5000	6381	7968	11432	17738	27881
Linear Regression of ln(n) over ln(τ)	ln(n) = −0.3115×ln(τ) + 8.5438, R^2 = 0.9988					

(b) Contracts 25–30 in Table 7 (r = 0.08, q = 0, σ = 0.223607, S_0 = 40, X = 40, λ_1 = 5, μ_J = −0.025, σ_J = 0.223607, λ_2 = 0.05)

τ	Benchmark Option Values	n = 5000	n = 6000	n = 7000	n = 8000	n = 9000	n = 10000	n = 11000	n = 12000	n = 13000	n = 14000	n = 15000	n = 16000	n = 17000	n = 18000	n = 19000	n = 20000	n = 21000	n = 22000	n = 23000	n = 24000	n = 25000
0.004	19.87189841	19.87189380	19.87189521	19.87189606	19.87189661	19.87189699	19.87189726	19.87189746	19.87189761	19.87189773	19.87189782	19.87189790	19.87189796	19.87189801	19.87189805	19.87189809	19.87189812	19.87189815	19.87189817	19.87189819	19.87189821	19.87189822
0.02	19.85152873	19.85152677	19.85152736	19.85152773	19.85152796	19.85152812	19.85152824	19.85152832	19.85152839	19.85152844	19.85152848	19.85152851	19.85152854	19.85152856	19.85152858	19.85152859	19.85152860	19.85152862	19.85152863	19.85152863	19.85152864	19.85152865
0.083	19.77129691	19.77129602	19.77129629	19.77129646	19.77129657	19.77129664	19.77129669	19.77129673	19.77129676	19.77129678	19.77129680	19.77129681	19.77129683	19.77129684	19.77129684	19.77129685	19.77129686	19.77129686	19.77129687	19.77129687	19.77129687	19.77129688
0.25	19.56340800	19.56340752	19.56340767	19.56340776	19.56340781	19.56340785	19.56340788	19.56340790	19.56340792	19.56340793	19.56340794	19.56340795	19.56340795	19.56340796	19.56340796	19.56340797	19.56340797	19.56340797	19.56340797	19.56340798	19.56340798	19.56340798
0.5	19.25973071	19.25973040	19.25973050	19.25973055	19.25973059	19.25973062	19.25973064	19.25973065	19.25973066	19.25973067	19.25973067	19.25973068	19.25973068	19.25973069	19.25973069	19.25973069	19.25973070	19.25973070	19.25973070	19.25973070	19.25973070	19.25973070
1	18.67731942	18.67731922	18.67731928	18.67731931	18.67731934	18.67731935	18.67731937	18.67731937	18.67731938	18.67731939	18.67731939	18.67731939	18.67731940	18.67731940	18.67731940	18.67731940	18.67731940	18.67731940	18.67731941	18.67731941	18.67731941	18.67731941

Relative error vs. Benchmark (x1.0E-10)

τ	n = 5000	n = 6000	n = 7000	n = 8000	n = 9000	n = 10000	n = 11000	n = 12000	n = 13000	n = 14000	n = 15000	n = 16000	n = 17000	n = 18000	n = 19000	n = 20000	n = 21000	n = 22000	n = 23000	n = 24000	n = 25000
0.004	−2317	−1608	−1181	−904	−714	−578	−477	−401	−342	−294	−256	−225	−199	−178	−159	−144	−130	−119	−109	−100	−93
0.02	−988	−686	−504	−386	−305	−247	−204	−171	−146	−126	−110	−96	−85	−76	−68	−62	−56	−51	−47	−43	−40
0.083	−451	−313	−230	−176	−139	−113	−93	−78	−67	−58	−50	−44	−39	−35	−31	−28	−26	−23	−21	−20	−18
0.25	−243	−168	−124	−95	−75	−61	−50	−42	−36	−31	−27	−24	−21	−19	−17	−15	−14	−13	−11	−11	−10
0.5	−163	−113	−83	−64	−50	−41	−34	−28	−24	−21	−18	−16	−14	−13	−11	−10	−9	−8	−8	−7	−7
1	−106	−74	−54	−42	−33	−27	−22	−18	−16	−14	−12	−10	−9	−8	−7	−7	−6	−5	−5	−5	−4

τ	1	0.5	0.25	0.083	0.02	0.004
Interpolated n to generate an error of −106E-10 (the error of τ = 1 and n = 5000)	5000	6224	7603	10329	15258	23266
Linear regression of ln(n) over ln(τ)	ln(n) = −0.2722×ln(τ) + 8.5398, R^2 = 0.9993					

Panels (a) and (b) of Table 9 present the error analyses of different values of the number of abscissas, n, for contracts 7–12 (representative cases for $r < q$) and contracts 25–30 (representative cases for $r > q$) in Table 7, respectively. The value of H is fixed as 60. For each set of contracts, we identify the interpolated value of n for each τ such that the errors for all τ are identical as the error given $\tau = 1$ and $n = 5000$. Next, we regress the logarithm of those interpolated values of n over the logarithm of the corresponding τ. The slope coefficients are −0.3115 and −0.2722 for each set of contracts. The results are similar to those under the LJD process in Table 6.

LJDR processes. Moreover, we propose a novel way to price perpetual American options under these two processes. A quadratic regression analysis of the option value over the time interval between two consecutive exercisable time points is performed, and the intercept coefficients can be employed to accurately estimate perpetual American option prices due to the extremely high R-squared values (higher than 0.9999). To the best of our knowledge, the proposed method herein is also the first one to obtain the option values of perpetual American options under the LJD and the LJDR process.

TABLE 10 Approximations of perpetual American put prices under the LJD and the LJDR processes

Option parameters			Quadratic regression of benchmark perpetual Bermudan put values over τ				Quadratic regression of perpetual Bermudan put values of our method with H = 60 and F = 150 over τ				Quadratic regression of perpetual Bermudan put values of our method with H = 40 and F = 150 over τ				Benchmark option values	Our method with H = 60 and F = 150	Our method with H = 40 and F = 150
(r, q, σ)	(S_0, X)	(λ_1, μ_J, σ_J, λ_2)	Intercept	Coeff. of τ	Coeff. of τ^2	R^2	Intercept	Coeff. of τ	Coeff. of τ^2	R^2	Intercept	Coeff. of τ	Coeff. of τ^2	R^2	τ = 0.004	τ = 0.004	τ = 0.004
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0)	22.0890557	−0.4069960	−0.0175795	0.9999985	22.0890552	−0.4069958	−0.0175796	0.9999985	22.0890550	−0.4069951	−0.0175799	0.9999985	22.0872798	22.0872793	22.0872790
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0)	18.8811233	−0.3476433	−0.0156740	0.9999986	18.8811229	−0.3476432	−0.0156740	0.9999986	18.8811226	−0.3476426	−0.0156743	0.9999986	18.8796137	18.8796133	18.8796131
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0)	15.8039787	−0.2907048	−0.0137780	0.9999989	15.8039784	−0.2907047	−0.0137781	0.9999989	15.8039782	−0.2907042	−0.0137783	0.9999989	15.8027256	15.8027252	15.8027251
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0)	17.7876431	−0.5863226	0.0125094	0.9999971	17.7876428	−0.5863225	0.0125094	0.9999971	17.7876426	−0.5863220	0.0125092	0.9999971	17.7856328	17.7856325	17.7856323
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0)	14.8564967	−0.4918505	0.0153622	0.9999986	14.8564964	−0.4918504	0.0153623	0.9999986	14.8564963	−0.4918500	0.0153621	0.9999986	14.8547372	14.8547369	14.8547368
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0)	12.1131347	−0.4017718	0.0139982	0.9999991	12.1131344	−0.4017718	0.0139982	0.9999991	12.1131343	−0.4017714	0.0139981	0.9999991	12.1116704	12.1116702	12.1116701
(0.08, 0.12, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	25.2877073	−1.6715016	0.1037484	0.9999975	25.2877067	−1.6715008	0.1037479	0.9999975	25.2877064	−1.6714998	0.1037473	0.9999975	25.2819819	25.2819812	25.2819808
(0.08, 0.12, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	22.0168481	−1.4286700	0.0852684	0.9999975	22.0168476	−1.4286693	0.0852680	0.9999975	22.0168473	−1.4286684	0.0852675	0.9999975	22.0119532	22.0119526	22.0119523
(0.08, 0.12, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	18.8389584	−1.1976109	0.0687836	0.9999975	18.8389580	−1.1976103	0.0687833	0.9999975	18.8389577	−1.1976096	0.0687829	0.9999975	18.8348451	18.8348446	18.8348444
(0.08, 0, 0.223607)	(40, 45)	(5, −0.025, 0.223607, 0.05)	22.8950496	−1.5065080	0.0880250	0.9999995	22.8950492	−1.5065077	0.0880249	0.9999995	22.8950490	−1.5065072	0.0880246	0.9999995	22.8893622	22.8893618	22.8893616
(0.08, 0, 0.223607)	(40, 40)	(5, −0.025, 0.223607, 0.05)	19.8766952	−1.2697911	0.0704748	0.9999995	19.8766949	−1.2697909	0.0704747	0.9999995	19.8766947	−1.2697904	0.0704745	0.9999995	19.8718984	19.8718981	19.8718980
(0.08, 0, 0.223607)	(40, 35)	(5, −0.025, 0.223607, 0.05)	16.9692038	−1.0492198	0.0547234	0.9999995	16.9692035	−1.0492197	0.0547233	0.9999995	16.9692034	−1.0492193	0.0547231	0.9999995	16.9652399	16.9652396	16.9652395

Table 10 presents the results of the approximate perpetual American put prices generated by our method under the LJD and the LJDR processes. Given the perpetual Bermudan put prices of our method in Tables 4 and 7, we conduct quadratic regressions of our perpetual Bermudan put values over different intervals between two consecutive exercisable time points, τ. The intercepts of those quadratic regressions can be employed to approximate perpetual American put prices, because perpetual Bermudan put prices converge theoretically to perpetual American put prices if the inter-exercise interval, τ, approaches zero. In addition to using the option values generated by our method (with $(H,F) = (40,150)$ and $(60,150)$), we also employ the benchmark option values as inputs for the quadratic regression. The last three columns compare the option values in the case of $\tau = 0.004$ (corresponding to daily exercisable) based on our method and the benchmark results reported in Tables 4 and 7.

ACKNOWLEDGMENT

The authors acknowledge the financial support from the Ministry of Science and Technology of Taiwan.

ORCID

Jr-Yan Wang http://orcid.org/0000-0001-7952-121X

REFERENCES

Alobaidi, G., Mansi, S., & Mallier, R. (2014). Numerical solution of an integral equation for perpetual Bermudan options. *International Journal of Computer Mathematics*, *91*, 1005–1011.

Amin, K. (1993). Jump diffusion option valuation in discrete time. *Journal of Finance*, *48*, 1833–1863.

Andricopoulos, A., Widdicks, M., Duck, P., & Newton, D. (2003). Universal option valuation using quadrature methods. *Journal of Financial Economics*, *67*, 447–471.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, *81*, 637–654.

Boyarchenko, S., & Levendorskii, S. (2002). Pricing of perpetual Bermudan options. *Quantitative Finance*, *22*, 432–442.

Boyle, G., & Guthrie, G. (2003). Investment, uncertainty, and liquidity. *Journal of Finance*, *58*, 2143–2166.

Chung, S. L., & Shackleton, B. (2007). Generalised Geske-Johnson interpolation of option prices. *Journal of Business Finance and Accounting*, *34*, 976–1001.

Geski, R., & Shastri, K. (1985). Valuation by Approximation: A comparison of alternative approaches. *Journal of Financial and Quantitative Analysis*, *20*, 45–72.

Guthrie, G. (2007). Missed opportunities: Optimal investment timing when information is costly. *Journal of Financial and Quantitative Analysis*, *42*, 467–488.

Hull, J. (2014). *Options, Futures, and other derivatives* (9th ed.). Cambridge, U.K.: Pearson.

Ju, N. (1998). Pricing an American option by approximating its early exercise boundary as a multipiece exponential function. *Review of Financial Studies*, *11*, 627–646.

Lambrecht, B., & Myers, S. (2007). A theory of takeovers and disinvestment. *Journal of Finance*, *62*, 809–845.

Leland, H. (1994). Corporate debt value, bond covenants, and optimal capital structure. *Journal of Finance*, *49*, 1213–1252.

Lin, J., & Liang, J. (2007). Pricing of perpetual American and Bermudan options by binomial tree method. *Frontiers of Mathematics in China*, *2*, 243–256.

Ma, J., & Luo, P. (2012). Integral equation methods for pricing perpetual Bermudan options. *Journal of Applied Finance and Banking*, *2*, 51–64.

Mcdonald, R., & Siegel, D. (1986). The value of waiting to invest. *Quarterly Journal of Economics*, *101*, 707–728.

Merton, R. (1973). Theory of rational option pricing. *The Bell Journal of Economics and Management Science*, *4*, 141–183.

Merton, R. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, *3*, 125–144.

Muroi, Y., & Yamada, T. (2008). An explicit finite difference approach to the pricing problems of perpetual Bermudan options. *Asia-Pacific Financial Markets*, *15*, 229–253.

Muthuraman, K. (2008). A moving boundary approach to American option pricing. *Journal of Economic Dynamics and Control*, *32*, 3520–3537.

Quigg, L. (1993). Empirical testing of real option- pricing models. *Journal of Finance*, *48*, 621–640.

Sundaresan, S., & Wang, N. (2007). Investment under uncertainty with strategic debt service. *American Economic Review*, *97*, 256–261.

How to cite this article: Chung S-L, Wang J-Y. A simple iteration algorithm to price perpetual Bermudan options under the lognormal jump-diffusion-ruin process. *J Futures Markets*. 2018;38:898–924.https://doi.org/10.1002/fut.21911

APPENDIX A

Option values of perpetual Bermudan puts under the the PD process ($\lambda_1 = \lambda_2 = 0$)

(r, q, σ)	(S_0, X)	τ	Benchmark Option Values	Our Method ($H = 60$, $F = 300$)	Our Method with MBA ($H = 60$, $F = 300$)	Our Method ($H = 40$, $F = 300$)	Our Method with MBA ($H = 40$, $F = 300$)	EFDM_LT ($\Delta t = 0.0001$)	EFDM_LT ($\Delta t = 0.00005$)	EFDM_LT ($\Delta t = 0.00001$)
(0.08, 0.12, 0.2)	(80, 100)	0.004	31.2487343	31.2487329	31.2487329	31.2487323	31.2487323	31.2487558	31.2487442	31.2487371
(0.08, 0.12, 0.2)	(80, 100)	0.02	31.2435743	31.2435730	31.2435730	31.2435724	31.2435724	31.2435971	31.2435924	31.2435779
(0.08, 0.12, 0.2)	(80, 100)	0.083	31.2224677	31.2224665	31.2224665	31.2224662	31.2224662	31.2225385	31.2225012	31.2224719
(0.08, 0.12, 0.2)	(80, 100)	0.25	31.1641671	31.1641659	31.1641659	31.1641663	31.1641663	31.1642037	31.1641929	31.1641742
(0.08, 0.12, 0.2)	(80, 100)	0.5	31.0719640	31.0719632	31.0719632	31.0719635	31.0719635	31.0720203	31.0719851	31.0719715
(0.08, 0.12, 0.2)	(80, 100)	1	30.8796546	30.8796541	30.8796541	30.8796543	30.8796543	30.8796874	30.8796772	30.8796611
(0.08, 0.12, 0.2)	(90, 100)	0.004	27.7766527	27.7766515	27.7766515	27.7766509	27.7766509	27.7766732	27.7766625	27.7766560
(0.08, 0.12, 0.2)	(90, 100)	0.02	27.7720661	27.7720649	27.7720649	27.7720643	27.7720643	27.7720877	27.7720832	27.7720700
(0.08, 0.12, 0.2)	(90, 100)	0.083	27.7533046	27.7533035	27.7533035	27.7533033	27.7533033	27.7533665	27.7533360	27.7533094
(0.08, 0.12, 0.2)	(90, 100)	0.25	27.7014801	27.7014790	27.7014790	27.7014794	27.7014794	27.7015140	27.7015040	27.7014872
(0.08, 0.12, 0.2)	(90, 100)	0.5	27.6196255	27.6196248	27.6196248	27.6196250	27.6196250	27.6196772	27.6196453	27.6196330
(0.08, 0.12, 0.2)	(90, 100)	1	27.4459753	27.4459749	27.4459749	27.4459750	27.4459750	27.4460051	27.4459959	27.4459818
(0.08, 0.12, 0.2)	(100, 100)	0.004	24.9989874	24.9989864	24.9989864	24.9989858	24.9989858	24.9990095	24.9990069	24.9989907
(0.08, 0.12, 0.2)	(100, 100)	0.02	24.9948595	24.9948584	24.9948584	24.9948579	24.9948579	24.9948826	24.9948855	24.9948633
(0.08, 0.12, 0.2)	(100, 100)	0.083	24.9779742	24.9779732	24.9779732	24.9779730	24.9779730	24.9780314	24.9779946	24.9779788
(0.08, 0.12, 0.2)	(100, 100)	0.25	24.9313322	24.9313313	24.9313313	24.9313315	24.9313315	24.9313664	24.9313643	24.9313389
(0.08, 0.12, 0.2)	(100, 100)	0.5	24.8576644	24.8576638	24.8576638	24.8576640	24.8576640	24.8577145	24.8576928	24.8576714
(0.08, 0.12, 0.2)	(100, 100)	1	24.7012775	24.7012771	24.7012771	24.7012772	24.7012772	24.7013081	24.7013067	24.7012837
(0.08, 0.12, 0.2)	(110, 100)	0.004	22.7263522	22.7263512	22.7263512	22.7263507	22.7263507	22.7263719	22.7263695	22.7263556
(0.08, 0.12, 0.2)	(110, 100)	0.02	22.7225995	22.7225986	22.7225986	22.7225981	22.7225981	22.7226201	22.7226228	22.7226035
(0.08, 0.12, 0.2)	(110, 100)	0.083	22.7072492	22.7072483	22.7072483	22.7072482	22.7072482	22.7073062	22.7072704	22.7072540
(0.08, 0.12, 0.2)	(110, 100)	0.25	22.6648474	22.6648466	22.6648466	22.6648468	22.6648468	22.6648781	22.6648762	22.6648540
(0.08, 0.12, 0.2)	(110, 100)	0.5	22.5978736	22.5978731	22.5978731	22.5978732	22.5978732	22.5979187	22.5978990	22.5978805
(0.08, 0.12, 0.2)	(110, 100)	1	22.4558896	22.4558892	22.4558892	22.4558893	22.4558893	22.4559171	22.4559158	22.4558956
(0.08, 0.12, 0.2)	(120, 100)	0.004	20.8324895	20.8324886	20.8324886	20.8324882	20.8324882	20.8325356	20.8325005	20.8324931
(0.08, 0.12, 0.2)	(120, 100)	0.02	20.8290496	20.8290487	20.8290487	20.8290483	20.8290483	20.8290965	20.8290659	20.8290536
(0.08, 0.12, 0.2)	(120, 100)	0.083	20.8149785	20.8149776	20.8149776	20.8149775	20.8149775	20.8150122	20.8150071	20.8149843
(0.08, 0.12, 0.2)	(120, 100)	0.25	20.7761101	20.7761094	20.7761094	20.7761096	20.7761096	20.7761662	20.7761316	20.7761166

(Continues)

(Continued)

(r, q, σ)	(S_0, X)	τ	Benchmark Option Values	Our Method ($H = 60$, $F = 300$)	Our Method with MBA ($H = 60$, $F = 300$)	Our Method ($H = 40$, $F = 300$)	Our Method with MBA ($H = 40$, $F = 300$)	EFDM_LT ($\Delta t = 0.0001$)	EFDM_LT ($\Delta t = 0.00005$)	EFDM_LT ($\Delta t = 0.00001$)
(0.08, 0.12, 0.2)	(120, 100)	0.5	20.7147177	20.7147171	20.7147171	20.7147173	20.7147173	20.7147869	20.7147360	20.7147244
(0.08, 0.12, 0.2)	(120, 100)	1	20.5846192	20.5846189	20.5846189	20.5846190	20.5846190	20.5846722	20.5846384	20.5846252
(0.08, 0.08, 0.2)	(80, 100)	0.004	25.5353209	25.5353198	25.5353198	25.5353193	25.5353193	25.5353245	25.5353519	25.5353286
(0.08, 0.08, 0.2)	(80, 100)	0.02	25.5270323	25.5270312	25.5270312	25.5270307	25.5270307	25.5270932	25.5270687	25.5270429
(0.08, 0.08, 0.2)	(80, 100)	0.083	25.4937549	25.4937538	25.4937538	25.4937536	25.4937536	25.4938884	25.4938076	25.4937628
(0.08, 0.08, 0.2)	(80, 100)	0.25	25.4044966	25.4044957	25.4044957	25.4044960	25.4044960	25.4045615	25.4045336	25.4045062
(0.08, 0.08, 0.2)	(80, 100)	0.5	25.2705057	25.2705050	25.2705050	25.2705052	25.2705052	25.2705864	25.2705415	25.2705106
(0.08, 0.08, 0.2)	(80, 100)	1	24.9788441	24.9788437	24.9788437	24.9788438	24.9788438	24.9788630	24.9788838	24.9788488
(0.08, 0.08, 0.2)	(90, 100)	0.004	21.2453531	21.2453522	21.2453522	21.2453518	21.2453518	21.2453589	21.2453808	21.2453609
(0.08, 0.08, 0.2)	(90, 100)	0.02	21.2384571	21.2384562	21.2384562	21.2384557	21.2384557	21.2385105	21.2384892	21.2384672
(0.08, 0.08, 0.2)	(90, 100)	0.083	21.2107699	21.2107690	21.2107690	21.2107688	21.2107688	21.2108799	21.2108164	21.2107780
(0.08, 0.08, 0.2)	(90, 100)	0.25	21.1364510	21.1364502	21.1364502	21.1364504	21.1364504	21.1365079	21.1364836	21.1364602
(0.08, 0.08, 0.2)	(90, 100)	0.5	21.0229125	21.0229119	21.0229119	21.0229121	21.0229121	21.0229810	21.0229433	21.0229177
(0.08, 0.08, 0.2)	(90, 100)	1	20.7968074	20.7968070	20.7968070	20.7968072	20.7968072	20.7968224	20.7968429	20.7968124
(0.08, 0.08, 0.2)	(100, 100)	0.004	18.0223416	18.0223408	18.0223408	18.0223404	18.0223404	18.0223522	18.0223806	18.0223486
(0.08, 0.08, 0.2)	(100, 100)	0.02	18.0164917	18.0164909	18.0164909	18.0164905	18.0164905	18.0165427	18.0165345	18.0165007
(0.08, 0.08, 0.2)	(100, 100)	0.083	17.9930047	17.9930040	17.9930040	17.9930038	17.9930038	17.9931009	17.9930332	17.9930122
(0.08, 0.08, 0.2)	(100, 100)	0.25	17.9299662	17.9299655	17.9299655	17.9299657	17.9299657	17.9300201	17.9300093	17.9299745
(0.08, 0.08, 0.2)	(100, 100)	0.5	17.8335113	17.8335108	17.8335108	17.8335110	17.8335110	17.8335751	17.8335529	17.8335162
(0.08, 0.08, 0.2)	(100, 100)	1	17.6385877	17.6385873	17.6385873	17.6385874	17.6385874	17.6386040	17.6386325	17.6385923
(0.08, 0.08, 0.2)	(110, 100)	0.004	15.5301036	15.5301029	15.5301029	15.5301026	15.5301026	15.5301125	15.5301368	15.5301103
(0.08, 0.08, 0.2)	(110, 100)	0.02	15.5250626	15.5250620	15.5250620	15.5250617	15.5250617	15.5251064	15.5250992	15.5250711
(0.08, 0.08, 0.2)	(110, 100)	0.083	15.5048236	15.5048230	15.5048230	15.5048229	15.5048229	15.5049135	15.5048519	15.5048309
(0.08, 0.08, 0.2)	(110, 100)	0.25	15.4505016	15.4505011	15.4505011	15.4505012	15.4505012	15.4505479	15.4505385	15.4505095
(0.08, 0.08, 0.2)	(110, 100)	0.5	15.3674466	15.3674462	15.3674462	15.3674463	15.3674463	15.3675015	15.3674821	15.3674515
(0.08, 0.08, 0.2)	(110, 100)	1	15.1981117	15.1981115	15.1981115	15.1981115	15.1981115	15.1981253	15.1981498	15.1981163
(0.08, 0.08, 0.2)	(120, 100)	0.004	13.5570600	13.5570594	13.5570594	13.5570592	13.5570592	13.5571046	13.5570828	13.5570665
(0.08, 0.08, 0.2)	(120, 100)	0.02	13.5526595	13.5526589	13.5526589	13.5526587	13.5526587	13.5527345	13.5526851	13.5526675
(0.08, 0.08, 0.2)	(120, 100)	0.083	13.5349918	13.5349912	13.5349912	13.5349911	13.5349911	13.5350465	13.5350286	13.5350001
(0.08, 0.08, 0.2)	(120, 100)	0.25	13.4875713	13.4875708	13.4875708	13.4875709	13.4875709	13.4876483	13.4875972	13.4875788
(0.08, 0.08, 0.2)	(120, 100)	0.5	13.4150689	13.4150685	13.4150685	13.4150686	13.4150686	13.4151531	13.4150937	13.4150738
(0.08, 0.08, 0.2)	(120, 100)	1	13.2671939	13.2671936	13.2671936	13.2671937	13.2671937	13.2672419	13.2672210	13.2671985

(Continues)

(Continued)

(r, q, σ)	(S_0, X)	τ	Benchmark Option Values	Our Method ($H = 60$, $F = 300$)	Our Method with MBA ($H = 60$, $F = 300$)	Our Method ($H = 40$, $F = 300$)	Our Method with MBA ($H = 40$, $F = 300$)	EFDM_LT ($\Delta t = 0.0001$)	EFDM_LT ($\Delta t = 0.00005$)	EFDM_LT ($\Delta t = 0.00001$)
(0.08, 0.04, 0.2)	(80, 100)	0.004	21.3732645	21.3732636	21.3732636	21.3732631	21.3732631	21.3733899	21.3733065	21.3732784
(0.08, 0.04, 0.2)	(80, 100)	0.02	21.3576661	21.3576652	21.3576652	21.3576647	21.3576647	21.3577197	21.3576768	21.3576800
(0.08, 0.04, 0.2)	(80, 100)	0.083	21.2966286	21.2966277	21.2966277	21.2966275	21.2966275	21.2968692	21.2967183	21.2966289
(0.08, 0.04, 0.2)	(80, 100)	0.25	21.1161603	21.1161596	21.1161596	21.1161598	21.1161598	21.1162399	21.1162094	21.1161758
(0.08, 0.04, 0.2)	(80, 100)	0.5	20.7818475	20.7818471	20.7818471	20.7818472	20.7818472	20.7818425	20.7818549	20.7818609
(0.08, 0.04, 0.2)	(80, 100)	1	20.0171052	20.0171050	20.0171050	20.0171051	20.0171051	20.0199211	20.0185259	20.0171166
(0.08, 0.04, 0.2)	(90, 100)	0.004	15.8066913	15.8066907	15.8066907	15.8066903	15.8066903	15.8067839	15.8067231	15.8067033
(0.08, 0.04, 0.2)	(90, 100)	0.02	15.7951564	15.7951557	15.7951557	15.7951554	15.7951554	15.7951959	15.7951650	15.7951684
(0.08, 0.04, 0.2)	(90, 100)	0.083	15.7495864	15.7495858	15.7495858	15.7495856	15.7495856	15.7497571	15.7496543	15.7495886
(0.08, 0.04, 0.2)	(90, 100)	0.25	15.6312630	15.6312624	15.6312624	15.6312626	15.6312626	15.6313256	15.6313029	15.6312774
(0.08, 0.04, 0.2)	(90, 100)	0.5	15.4554068	15.4554064	15.4554064	15.4554065	15.4554065	15.4554014	15.4554143	15.4554212
(0.08, 0.04, 0.2)	(90, 100)	1	15.0613756	15.0613754	15.0613754	15.0613755	15.0613755	15.0614389	15.0614119	15.0613889
(0.08, 0.04, 0.2)	(100, 100)	0.004	12.0678734	12.0678729	12.0678729	12.0678726	12.0678726	12.0679495	12.0679198	12.0678830
(0.08, 0.04, 0.2)	(100, 100)	0.02	12.0590668	12.0590663	12.0590663	12.0590660	12.0590660	12.0591024	12.0590955	12.0590765
(0.08, 0.04, 0.2)	(100, 100)	0.083	12.0242775	12.0242770	12.0242770	12.0242769	12.0242769	12.0244087	12.0243108	12.0242797
(0.08, 0.04, 0.2)	(100, 100)	0.25	11.9332652	11.9332647	11.9332647	11.9332649	11.9332649	11.9333176	11.9333172	11.9332765
(0.08, 0.04, 0.2)	(100, 100)	0.5	11.7996400	11.7996397	11.7996397	11.7996398	11.7996398	11.7996372	11.7996659	11.7996517
(0.08, 0.04, 0.2)	(100, 100)	1	11.5392260	11.5392258	11.5392258	11.5392259	11.5392259	11.5392834	11.5392767	11.5392378
(0.08, 0.04, 0.2)	(110, 100)	0.004	9.4536865	9.4536861	9.4536861	9.4536859	9.4536859	9.4537434	9.4537211	9.4536946
(0.08, 0.04, 0.2)	(110, 100)	0.02	9.4467876	9.4467872	9.4467872	9.4467870	9.4467870	9.4468129	9.4468084	9.4467958
(0.08, 0.04, 0.2)	(110, 100)	0.083	9.4195345	9.4195341	9.4195341	9.4195341	9.4195341	9.4196448	9.4195647	9.4195371
(0.08, 0.04, 0.2)	(110, 100)	0.25	9.3482795	9.3482791	9.3482791	9.3482792	9.3482792	9.3483180	9.3483186	9.3482890
(0.08, 0.04, 0.2)	(110, 100)	0.5	9.2425947	9.2425945	9.2425945	9.2425945	9.2425945	9.2425894	9.2426130	9.2426044
(0.08, 0.04, 0.2)	(110, 100)	1	9.0397350	9.0397349	9.0397349	9.0397349	9.0397349	9.0397766	9.0397734	9.0397449
(0.08, 0.04, 0.2)	(120, 100)	0.004	7.5649106	7.5649103	7.5649103	7.5649101	7.5649101	7.5650013	7.5649295	7.5649178
(0.08, 0.04, 0.2)	(120, 100)	0.02	7.5593901	7.5593898	7.5593898	7.5593896	7.5593896	7.5594553	7.5593979	7.5593973
(0.08, 0.04, 0.2)	(120, 100)	0.083	7.5375820	7.5375817	7.5375817	7.5375816	7.5375816	7.5376382	7.5376206	7.5375863
(0.08, 0.04, 0.2)	(120, 100)	0.25	7.4805619	7.4805616	7.4805616	7.4805617	7.4805617	7.4806373	7.4805845	7.4805702
(0.08, 0.04, 0.2)	(120, 100)	0.5	7.3960263	7.3960261	7.3960261	7.3960262	7.3960262	7.3960663	7.3960323	7.3960346
(0.08, 0.04, 0.2)	(120, 100)	1	7.2314746	7.2314745	7.2314745	7.2314745	7.2314745	7.2315504	7.2314964	7.2314831
(0.08, 0, 0.2)	(80, 100)	0.004	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(80, 100)	0.02	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000

(Continues)

(Continued)

(r, q, σ)	(S_0, X)	τ	Benchmark Option Values	Our Method ($H = 60$, $F = 300$)	Our Method with MBA ($H = 60$, $F = 300$)	Our Method ($H = 40$, $F = 300$)	Our Method with MBA ($H = 40$, $F = 300$)	EFDM_LT ($\Delta t = 0.0001$)	EFDM_LT ($\Delta t = 0.00005$)	EFDM_LT ($\Delta t = 0.00001$)
(0.08, 0, 0.2)	(80, 100)	0.083	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(80, 100)	0.25	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(80, 100)	0.5	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(80, 100)	1	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000	20.0000000
(0.08, 0, 0.2)	(90, 100)	0.004	12.4809293	12.4809287	12.4809287	12.4809285	12.4809285	12.4809183	12.4809556	12.4809525
(0.08, 0, 0.2)	(90, 100)	0.02	12.4611920	12.4611914	12.4611914	12.4611912	12.4611912	12.4613242	12.4612332	12.4612016
(0.08, 0, 0.2)	(90, 100)	0.083	12.3849424	12.3849419	12.3849419	12.3849418	12.3849418	12.3851926	12.3850463	12.3849586
(0.08, 0, 0.2)	(90, 100)	0.25	12.1750273	12.1750269	12.1750269	12.1750270	12.1750270	12.1751099	12.1750619	12.1750346
(0.08, 0, 0.2)	(90, 100)	0.5	11.8047441	11.8047438	11.8047438	11.8047439	11.8047439	11.8048292	11.8047432	11.8047492
(0.08, 0, 0.2)	(90, 100)	1	10.9970136	10.9970135	10.9970135	10.9970136	10.9970136	10.9970633	10.9970143	10.9970228
(0.08, 0, 0.2)	(100, 100)	0.004	8.1887377	8.1887373	8.1887373	8.1887372	8.1887372	8.1887253	8.1887814	8.1887523
(0.08, 0, 0.2)	(100, 100)	0.02	8.1757880	8.1757876	8.1757876	8.1757874	8.1757874	8.1758696	8.1758414	8.1757937
(0.08, 0, 0.2)	(100, 100)	0.083	8.1253475	8.1253471	8.1253471	8.1253471	8.1253471	8.1254986	8.1253810	8.1253575
(0.08, 0, 0.2)	(100, 100)	0.25	7.9976377	7.9976374	7.9976374	7.9976375	7.9976375	7.9976899	7.9976873	7.9976422
(0.08, 0, 0.2)	(100, 100)	0.5	7.8137579	7.8137577	7.8137577	7.8137578	7.8137578	7.8138281	7.8137837	7.8137621
(0.08, 0, 0.2)	(100, 100)	1	7.4194870	7.4194869	7.4194869	7.4194870	7.4194870	7.4195371	7.4195131	7.4194957
(0.08, 0, 0.2)	(110, 100)	0.004	5.5930180	5.5930178	5.5930178	5.5930177	5.5930177	5.5929980	5.5930416	5.5930280
(0.08, 0, 0.2)	(110, 100)	0.02	5.5841732	5.5841730	5.5841730	5.5841728	5.5841728	5.5842173	5.5842034	5.5841771
(0.08, 0, 0.2)	(110, 100)	0.083	5.5497238	5.5497235	5.5497235	5.5497235	5.5497235	5.5498285	5.5497479	5.5497309
(0.08, 0, 0.2)	(110, 100)	0.25	5.4617869	5.4617867	5.4617867	5.4617868	5.4617868	5.4618103	5.4618142	5.4617899
(0.08, 0, 0.2)	(110, 100)	0.5	5.3371179	5.3371178	5.3371178	5.3371178	5.3371178	5.3371544	5.3371291	5.3371207
(0.08, 0, 0.2)	(110, 100)	1	5.1029243	5.1029243	5.1029243	5.1029243	5.1029243	5.1029543	5.1029398	5.1029312
(0.08, 0, 0.2)	(120, 100)	0.004	3.9490440	3.9490439	3.9490439	3.9490438	3.9490438	3.9490762	3.9490479	3.9490513
(0.08, 0, 0.2)	(120, 100)	0.02	3.9427990	3.9427988	3.9427988	3.9427987	3.9427987	3.9428764	3.9428075	3.9428019
(0.08, 0, 0.2)	(120, 100)	0.083	3.9184754	3.9184753	3.9184753	3.9184752	3.9184752	3.9185077	3.9185064	3.9184826
(0.08, 0, 0.2)	(120, 100)	0.25	3.8564276	3.8564274	3.8564274	3.8564275	3.8564275	3.8564894	3.8564343	3.8564299
(0.08, 0, 0.2)	(120, 100)	0.5	3.7673301	3.7673300	3.7673300	3.7673300	3.7673300	3.7673995	3.7673254	3.7673322
(0.08, 0, 0.2)	(120, 100)	1	3.6044163	3.6044162	3.6044162	3.6044162	3.6044162	3.6044795	3.6044155	3.6044214
(0.05, 0, 0.3)	(100, 80)	0.004	14.4920242	14.4920239	14.4920239	14.4920237	14.4920237	14.4920776	14.4920631	14.4920345
(0.05, 0, 0.3)	(100, 80)	0.02	14.4859108	14.4859106	14.4859106	14.4859104	14.4859104	14.4860078	14.4859354	14.4859173
(0.05, 0, 0.3)	(100, 80)	0.083	14.4617611	14.4617608	14.4617608	14.4617607	14.4617607	14.4618088	14.4618004	14.4617713
(0.05, 0, 0.3)	(100, 80)	0.25	14.3985248	14.3985246	14.3985246	14.3985245	14.3985245	14.3985930	14.3985490	14.3985352

(Continues)

(Continued)

(r, q, σ)	(S_0, X)	τ	Benchmark Option Values	Our Method ($H = 60$, $F = 300$)	Our Method with MBA ($H = 60$, $F = 300$)	Our Method ($H = 40$, $F = 300$)	Our Method with MBA ($H = 40$, $F = 300$)	EFDM_LT ($\Delta t = 0.0001$)	EFDM_LT ($\Delta t = 0.00005$)	EFDM_LT ($\Delta t = 0.00001$)
(0.05, 0, 0.3)	(100, 80)	0.5	14.3044590	14.3044588	14.3044588	14.3044587	14.3044587	14.3045300	14.3045052	14.3044654
(0.05, 0, 0.3)	(100, 80)	1	14.1189287	14.1189284	14.1189284	14.1189285	14.1189285	14.1189763	14.1189584	14.1189387
(0.05, 0, 0.3)	(100, 90)	0.004	18.5830805	18.5830801	18.5830801	18.5830799	18.5830799	18.5831350	18.5831301	18.5830895
(0.05, 0, 0.3)	(100, 90)	0.02	18.5752414	18.5752410	18.5752410	18.5752409	18.5752409	18.5753598	18.5753036	18.5752545
(0.05, 0, 0.3)	(100, 90)	0.083	18.5442742	18.5442739	18.5442739	18.5442737	18.5442737	18.5443425	18.5443095	18.5442835
(0.05, 0, 0.3)	(100, 90)	0.25	18.4631874	18.4631870	18.4631870	18.4631869	18.4631869	18.4632602	18.4632492	18.4631961
(0.05, 0, 0.3)	(100, 90)	0.5	18.3425056	18.3425053	18.3425053	18.3425053	18.3425053	18.3425828	18.3425407	18.3425173
(0.05, 0, 0.3)	(100, 90)	1	18.1058762	18.1058759	18.1058759	18.1058760	18.1058760	18.1059507	18.1059330	18.1058866
(0.05, 0, 0.3)	(100, 100)	0.004	23.2122294	23.2122290	23.2122290	23.2122288	23.2122288	23.2123737	23.2123016	23.2122456
(0.05, 0, 0.3)	(100, 100)	0.02	23.2024376	23.2024371	23.2024371	23.2024369	23.2024369	23.2025846	23.2024892	23.2024490
(0.05, 0, 0.3)	(100, 100)	0.083	23.1637563	23.1637559	23.1637559	23.1637557	23.1637557	23.1638043	23.1638253	23.1637723
(0.05, 0, 0.3)	(100, 100)	0.25	23.0624719	23.0624715	23.0624715	23.0624713	23.0624713	23.0626237	23.0625236	23.0624879
(0.05, 0, 0.3)	(100, 100)	0.5	22.9116893	22.9116889	22.9116889	22.9116888	22.9116888	22.9118415	22.9117678	22.9116974
(0.05, 0, 0.3)	(100, 100)	1	22.6189727	22.6189723	22.6189723	22.6189724	22.6189724	22.6191033	22.6190345	22.6189887
(0.05, 0, 0.3)	(100, 110)	0.004	28.3858179	28.3858173	28.3858173	28.3858171	28.3858171	28.3860007	28.3859144	28.3858373
(0.05, 0, 0.3)	(100, 110)	0.02	28.3738436	28.3738431	28.3738431	28.3738428	28.3738428	28.3740115	28.3739403	28.3738584
(0.05, 0, 0.3)	(100, 110)	0.083	28.3265409	28.3265404	28.3265404	28.3265402	28.3265402	28.3266197	28.3266241	28.3265602
(0.05, 0, 0.3)	(100, 110)	0.25	28.2026526	28.2026521	28.2026521	28.2026519	28.2026519	28.2028403	28.2027490	28.2026718
(0.05, 0, 0.3)	(100, 110)	0.5	28.0188951	28.0188946	28.0188946	28.0188946	28.0188946	28.0190828	28.0189824	28.0189037
(0.05, 0, 0.3)	(100, 110)	1	27.6623298	27.6623294	27.6623294	27.6623295	27.6623295	27.6624996	27.6624268	27.6623493
(0.05, 0, 0.3)	(100, 120)	0.004	34.1096500	34.1096493	34.1096493	34.1096490	34.1096490	34.1098548	34.1097245	34.1096734
(0.05, 0, 0.3)	(100, 120)	0.02	34.0952611	34.0952605	34.0952605	34.0952602	34.0952602	34.0953715	34.0953300	34.0952740
(0.05, 0, 0.3)	(100, 120)	0.083	34.0384202	34.0384196	34.0384196	34.0384193	34.0384193	34.0385761	34.0384957	34.0384432
(0.05, 0, 0.3)	(100, 120)	0.25	33.8895081	33.8895076	33.8895076	33.8895073	33.8895073	33.8896914	33.8895761	33.8895315
(0.05, 0, 0.3)	(100, 120)	0.5	33.6706992	33.6706986	33.6706986	33.6706986	33.6706986	33.6708775	33.6707959	33.6707148
(0.05, 0, 0.3)	(100, 120)	1	33.2330523	33.2330518	33.2330518	33.2330519	33.2330519	33.2332634	33.2331039	33.2330744

Appendix B

Benchmark option values and S^* of perpetual Bermudan puts

This paper proposes a regression-based extrapolation approach to determine the benchmarks of the option value and the critical stock price of a perpetual Bermudan put. We perform quadratic regressions for option values or critical stock prices over $1/n$ and adopt intercept results as benchmarks. The reasons for considering the values n being 6000, 7000, …, 11000 are as follows. First, we find that the option prices of perpetual Bermudan puts based on our method can converge monotonically when the number of abscissas in the holding region, n, is larger than 5000. Second, we distinguish that the marginal improvement when $n = 6000$ is significant in comparison to the results of $n = 5000$. Therefore, we choose the minimum value of n to be 6000 for the conservative reason as well as for pursuit of accuracy. Third, we also notice that if we need the R-squared value of the quadratic regression to be higher than 0.999999, then at least six observations are needed. Consequently, we consider the six values of n to be 6000, 7000, 8000, 9000, 10000, and 11000 for computing the benchmark option value and S^* of the perpetual Bermudan put. Each first five option contracts in Tables 1 (Appendix A), 4, 7 (for the PD process, the LJD process, and the LJDR process, respectively) are employed to illustrate how to determine the benchmarks of the option value and critical stock price of a perpetual Bermudan put. The results are shown in Tables B.1 to B.6.

Table B.1 Benchmark option values of perpetual Bermudan puts under the PD process: $S_0 = 80$, $X = 100$, $r = 0.08$, $q = 0.12$, and $\sigma = 0.2$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of ($1/n$)	Coeff. of ($1/n^2$)	R^2
0.004	31.2487267	31.2487287	31.2487300	31.2487309	31.2487316	31.2487320	31.2487343	0.0000890	−272.8201597	1.0000000
0.02	31.2435710	31.2435719	31.2435725	31.2435729	31.2435731	31.2435734	31.2435743	0.0000039	−119.0251434	1.0000000
0.083	31.2224662	31.2224666	31.2224668	31.2224670	31.2224671	31.2224672	31.2224677	−0.0000010	−55.7941488	1.0000000
0.25	31.1641663	31.1641665	31.1641666	31.1641667	31.1641668	31.1641669	31.1641671	−0.0000009	−30.2113117	1.0000000
0.5	31.0719634	31.0719636	31.0719637	31.0719637	31.0719638	31.0719638	31.0719640	−0.0000007	−20.0415874	1.0000000
1	30.8796543	30.8796544	30.8796544	30.8796545	30.8796545	30.8796545	30.8796546	−0.0000005	−12.9539992	1.0000000

Table B.2 Benchmarks of critical stock prices of perpetual Bermudan puts under the PD process: $S_0 = 80$, $X = 100$, $r = 0.08$, $q = 0.12$, and $\sigma = 0.2$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of ($1/n$)	Coeff. of ($1/n^2$)	R^2
0.004	50.3723124	50.3721832	50.3720993	50.3720418	50.3720006	50.3719702	50.3718253	0.0001374	17537.2953	1.0000000
0.02	50.8408041	50.8407796	50.8407637	50.8407528	50.8407449	50.8407392	50.8407116	0.0000916	3329.9713	1.0000000
0.083	51.7518023	51.7517970	51.7517935	51.7517912	51.7517895	51.7517882	51.7517822	0.0000292	723.8476	1.0000000
0.25	53.1226448	53.1226433	53.1226423	53.1226416	53.1226411	53.1226407	53.1226390	0.0000093	208.6436	1.0000000
0.5	54.5378687	54.5378681	54.5378676	54.5378673	54.5378671	54.5378670	54.5378662	0.0000041	90.1044	1.0000000
1	56.6576933	56.6576931	56.6576929	56.6576928	56.6576927	56.6576926	56.6576923	0.0000018	36.3426	1.0000000

Table B.3 Benchmark option values of perpetual Bermudan puts under the LJD process: $S_0 = 40$, $X = 45$, $r = 0.08$, $q = 0.12$, $\sigma = 0.223607$, $\lambda_1 = 5$, $\mu_J = -0.025$, and $\sigma_J = 0.223607$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of ($1/n$)	Coeff. of ($1/n^2$)	R^2
0.004	22.0872745	22.0872759	22.0872768	22.0872774	22.0872779	22.0872782	22.0872798	0.0001649	−189.6177587	1.0000000
0.02	22.0808626	22.0808633	22.0808637	22.0808639	22.0808641	22.0808643	22.0808649	0.0000076	−82.7494335	1.0000000
0.083	22.0552402	22.0552405	22.0552407	22.0552408	22.0552409	22.0552409	22.0552413	−0.0000003	−38.8614260	1.0000000
0.25	21.9865284	21.9865286	21.9865287	21.9865287	21.9865288	21.9865288	21.9865290	−0.0000006	−20.9966655	1.0000000
0.5	21.8809009	21.8809010	21.8809011	21.8809011	21.8809012	21.8809012	21.8809013	−0.0000005	−13.8585921	1.0000000
1	21.6645248	21.6645248	21.6645249	21.6645249	21.6645249	21.6645249	21.6645250	−0.0000004	−8.8228748	1.0000000

Table B.4 Benchmarks of critical stock prices of perpetual Bermudan puts under the LJD process: $S_0 = 40$, $X = 45$, $r = 0.08$, $q = 0.12$, $\sigma = 0.223607$, $\lambda_1 = 5$, $\mu_J = -0.025$, and $\sigma_J = 0.223607$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of (1/n)	Coeff. of (1/n^2)	R^2
0.004	12.1653463	12.1652363	12.1651649	12.1651160	12.1650810	12.1650551	12.1649318	−0.000416493	14925.14644	1.0000000
0.02	12.3321705	12.3321522	12.3321403	12.3321322	12.3321264	12.3321221	12.3321015	4.24797E−05	2484.118657	1.0000000
0.083	12.7293940	12.7293908	12.7293887	12.7293873	12.7293863	12.7293855	12.7293819	1.63498E−05	435.6919592	1.0000000
0.25	13.4557444	13.4557437	13.4557432	13.4557428	13.4557426	13.4557424	13.4557415	4.55009E−06	103.1432167	1.0000000
0.5	14.3016403	14.3016400	14.3016398	14.3016397	14.3016396	14.3016395	14.3016392	1.85408E−06	41.25276750	1.0000000
1	15.6595502	15.6595500	15.6595500	15.6595499	15.6595499	15.6595498	15.6595497	7.43466E−07	16.38533495	1.0000000

Table B.5 Benchmark option values of perpetual Bermudan puts under the LJDR process: $S_0 = 40$, $X = 45$, $r = 0.08$, $q = 0.12$, $\sigma = 0.223607$, $\lambda_1 = 5$, $\mu_J = -0.025$, $\sigma_J = 0.223607$, and $\lambda_2 = 0.05$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of (1/n)	Coeff. of (1/n^2)	R^2
0.004	25.2819745	25.2819764	25.2819777	25.2819786	25.2819792	25.2819797	25.2819819	0.0001146	−267.1763813	1.0000000
0.02	25.2547278	25.2547286	25.2547291	25.2547295	25.2547297	25.2547299	25.2547308	0.0000051	−106.7254484	1.0000000
0.083	25.1483671	25.1483674	25.1483676	25.1483677	25.1483678	25.1483679	25.1483683	−0.0000007	−43.9345320	1.0000000
0.25	24.8754458	24.8754459	24.8754460	24.8754461	24.8754461	24.8754462	24.8754464	−0.0000006	−21.1985448	1.0000000
0.5	24.4788996	24.4788997	24.4788998	24.4788998	24.4788999	24.4788999	24.4789000	−0.0000005	−13.1819635	1.0000000
1	23.7197655	23.7197656	23.7197656	23.7197657	23.7197657	23.7197657	23.7197658	−0.0000003	−7.9360837	1.0000000

Table B.6 Benchmarks of critical stock prices of perpetual Bermudan puts under the LJDR process: $S_0 = 40$, $X = 45$, $r = 0.08$, $q = 0.12$, $\sigma = 0.223607$, $\lambda_1 = 5$, $\mu_J = -0.025$, $\sigma_J = 0.223607$, and $\lambda_2 = 0.05$

τ	$n = 6000$	$n = 7000$	$n = 8000$	$n = 9000$	$n = 10000$	$n = 11000$	Intercept (Benchmark)	Coeff. of (1/n)	Coeff. of (1/n^2)	R^2
0.004	11.1056125	11.1055202	11.1054602	11.1054191	11.1053897	11.1053679	11.1052644	−0.000454781	12536.38575	1.000000
0.02	11.4759373	11.4759220	11.4759121	11.4759052	11.4759004	11.4758968	11.4758796	2.80691E-05	2078.849451	1.000000
0.083	12.2404135	12.2404109	12.2404091	12.2404080	12.2404071	12.2404065	12.2404035	1.33107E-05	361.6845075	1.000000
0.25	13.4616499	13.4616493	13.4616489	13.4616486	13.4616484	13.4616483	13.4616476	3.26972E-06	84.29330193	1.000000
0.5	14.7586454	14.7586451	14.7586450	14.7586449	14.7586448	14.7586447	14.7586445	1.53350E-06	33.0858506	1.000000
1	16.6969808	16.6969807	16.6969807	16.6969806	16.6969806	16.6969806	16.6969805	5.87589E-07	12.7810589	1.000000