An Ingenious, Piecewise Linear Interpolation Algorithm for Pricing Arithmetic Average Options

Tian-Shyr Dai[1], Jr-Yan Wang[2], and Hui-Shan Wei[3]

[1] Department of Information and Finance Management, National Chiao Tung University, Hsinchu, Taiwan
cameldai@mail.nctu.edu.tw
[2] Graduate School of Finance, National Taiwan University of Science and Technology, Taipei, Taiwan
jywang@mail.ntust.edu.tw
[3] Department of Finance, National Central University, Taoyuan County, Taiwan

Abstract. Pricing arithmetic average options continues to intrigue researchers in the field of financial engineering. Since there is no analytical solution for this problem until present, developing an efficient numerical algorithm becomes a promising alternative. One of the most famous numerical algorithms for pricing arithmetic average options is introduced by Hull and White [10]. In this paper, motivated by the common idea of reducing the nonlinearity error in the adaptive mesh model [7] and the adaptive quadrature numerical integration method [6], the logarithmically equally-spaced placement rule in the Hull and White's model is replaced by an adaptive placement method, in which the number of representative average prices is proportional to the degree of curvature of the option value as a function of the arithmetic average price. Numerical experiments verify the superior performance of our method in terms of reducing the interpolation error. In fact, it is straightforward to apply this method to any pricing algorithm with the techniques of augmented state variables and the piece-wise linear interpolation approximation.

Keywords: Arithmetic average options, logarithmically equally-spaced placement, adaptive placement.

1 Introduction

Asian options are path dependent securities whose payoff depends on the average of the underlying prices during the option life. They were originally issued in 1987 by Banker's Trust Tokyo on crude oil contracts, and hence the name "Asian" option. Asian options are commonly traded in a thinly traded market to prevent price manipulation. Besides, Asian options are less expensive than comparable vanilla options, because the volatility of the average value of an underling asset is lower than the volatility of the value of the underling asset. In practice, end-users of commodities, energies, or foreign currencies tend to be exposed to average

M.-Y. Kao and X.-Y. Li (Eds.): AAIM 2007, LNCS 4508, pp. 262–272, 2007.
© Springer-Verlag Berlin Heidelberg 2007

prices over time, so Asian options are also attractive for them. This is because Asian options are often used as they more closely replicate the requirements of end-users exposed to price movements on the underlying asset.

To this date, more and more financial instruments include the average feature from Asian options, for example, structure notes issued by international banks, the contracts of convertible bonds in Taiwan, etc. If the underlying price process follows the geometric Brownian motion, the analytical pricing formula for geometric average options is feasible since the product of lognormally distributed prices remains to follow the lognormal distribution. Based upon this observation, Kemna and Vorst proposed an analytical solution for European geometric average options [11]. Unfortunately, it is still analytically intractable to price arithmetic average options due to the lack of proper mathematical representation for the sum of lognormal random variables. Thus many researches were devoted to deal with the distribution of the sum of lognormal random variables and derive approximated pricing formulae for arithmetic average options. Several works along this direction include the fast Fourier transformation in [2], the Edgeworth series expansion in [16], the reciprocal Gamma distribution in [13], the Laplace transform inversion in [9], etc.

The tree-based model is a possible alternative to price arithmetic average options. However, the naive pricing method based on the tree model which is able to derive the exact value of the arithmetic average options by recording all possible arithmetic average prices is simply intractable due to the exponential growth of the number of possible arithmetic average prices with respect to the number of time steps. In this paper, the exact option value stands for the option value derived from a tree-based model without any interpolation error.

To overcome the problem of the exponential growth of the number of possible arithmetic average prices, Dai and Lyuu develop a trinomial-tree pricing model for arithmetic average options that guarantees the convergence to the exact option value [5], in which the notion of integrality of stock prices is employed to reduce the time complexity of recording all possible arithmetic average prices to be sub-exponential. However, it is still intractable to price arithmetic average options via this model when the number of time steps is large.

On the other hand, in Hull and White's model [10], instead of keeping track of all possible arithmetic average prices, representative average prices are logarithmically equally-spaced placed between the maximum and minimum arithmetic average prices for each node, and the piece-wise linear interpolation is adopted to derive the corresponding option values for nonexistent average prices during the backward induction. Therefore, the interpolation error occurs and whether the interpolation error vanishes is uncertain for all but a single scenario in which the number of representative average prices for each node and the number of time steps for the tree model are well collocated, see [8].

Along with the line of [10], Neave and Turnbull [14] suggest using the conditional frequency distribution to adjust the number of representative average prices for each node. Cho and Lee [3] replace the uniform allocation of the number of representative average prices in the Hull and White's model with the

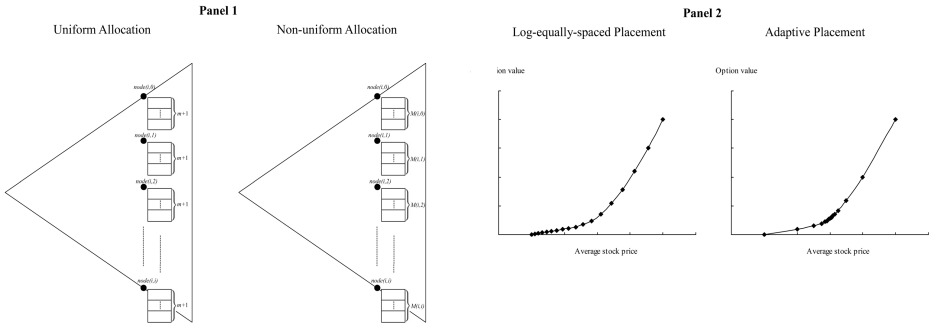

Fig. 1. The illustration of our adaptive placement method. Hull and White [10] adopted the combination of the uniform allocation and logarithmically equally-spaced placement rules in their pricing algorithm, i.e. m=100. Other modifications of the Hull and White's model focus on devising more efficient non-uniform allocation rules, i.e. $M(i,j)$ is different for each $node(i,j)$. However, the logarithmically equally-spaced placement rule is a common component in these models. In our adaptive placement method, the number of representative average prices is proportional to the degree of curvature of the option value function and an efficient non-uniform allocation of representative average prices is achieved automatically.

distribution of the number of possible geometric average prices. Klassen [12] proposes a revised version of the algorithm of [10], in which only a set of average prices at each node is considered, and the grid space for the logarithm of the arithmetic average prices is a pre-specified function of the time to maturity, the time steps, and the volatility of the stock price process. Although these methods of adjusting the allocation of the number of the representative average prices over the tree exhibit superior convergence rate to exact option values than the Hull and White's model, their major disadvantages are the absence of the economic meanings and the guarantee of the convergence of the interpolation error.

A different point of view is adopted in [1] and [4] to improve the convergence rate of the tree-based models for pricing arithmetic average options. Instead of recording the maximum and minimum arithmetic average prices, a more compact range is derived such that the interpolation error can be reduced effectively. Moreover, for European-style arithmetic average options, the optimal allocation of the representative average prices over the tree is derived in [4] to minimize the accumulated interpolation error of the option value.

Dedicated to devising the allocation of representative average prices over the tree to reduce the interpolation error, the suggested allocation rules of the above modifications are no longer uniformly distributed but are contingent on the probability reaching the underlying node, the time to maturity of the underlying node, the number of time steps in the tree model, and the volatility of the underlying process. The differences between uniform and non-uniform allocations are illustrated in Panel 1 of Fig. 1.

With the uniform allocation rule being replaced, the logarithmically equally-spaced placement rule proposed by Hull and White is still retained in the aforementioned works. Aiming at simultaneously guaranteeing the convergence of the interpolation error and improving the efficiency, we proposed a novel aspect to minimize the interpolation error by replacing the logarithmically equally-spaced placement rule with an adaptive placement method, in which more representative average values are needed in the area around which the option value function of the arithmetic average price is with higher degree of curvature, and fewer representative average values are placed where the option value function is with lower degree of curvature. The ideas of the adaptive placement method and the logarithmically equally-spaced rules are illustrated in Panel 2 of Fig. 1. To achieve this goal, the adaptive placement method is actually designed to govern the linear interpolation error between each pair of adjacent representative average prices under a limit criterion. Moreover, our method forms automatically an efficient non-uniform allocation of representative average prices over the tree.

2 Arithmetic Average Options

In this paper, the non-dividend-paying underlying stock price in the risk neutral world is assumed to follow the geometric Brownian motion: $dS_t/S_t = rdt + \sigma dZ$, where r is the risk free rate, σ is the volatility of the asset price, and Z is a Wiener process. Suppose that the stock price is sampled at the time points $0 = t_0 < t_1 < \cdots < t_n = T$ during the life of the arithmetic average options. If the corresponding stock prices are $S_{t_0}, S_{t_1}, \cdots, S_{t_n}$, the arithmetic average price from time 0 to t is $A(t) = (\sum_{i=0}^{l} S_{t_i})/(l+1)$, where $t_l \leq t < t_{l+1}$. In addition, the exercise value of the arithmetic average call considered in this paper at time t is $\max(A(t) - X, 0)$, where X is the strike price of the arithmetic average call. Furthermore, the stock price is assumed to be sampled periodically, which is often the case in the real world, and therefore $t_i = i\Delta t$ and $\Delta t = T/n$.

The Hull and White's Model

In the field of option pricing, the binomial-tree model divides the time horizon of an option into n discrete time steps and discretizes the stock prices at each time step. In Panel 1 of Fig. 2, it is shown that the stock price at time step 0 is S_0 (at $node(0,0)$), and the stock price can either move up to S_0u (at $node(1,0)$) or down to S_0d (at $node(1,1)$) at the first time step, where $u = \exp(\sigma\sqrt{\Delta t})$ is the magnitude of a upward movement for the stock price, and $d = \exp(-\sigma\sqrt{\Delta t})$ is the magnitude of a downward movement for the stock price. Similarly, each stock price can either move up or move down at subsequent time steps.

It is in theory possible to employ the binomial-tree model to calculate exact values of arithmetic average options by recording all possible average values reaching each node. Unfortunately, if the option life is divided into n periods, the number of all possible arithmetic average prices is 2^n, which implies that the computation complexity is intractable even for small values of n.

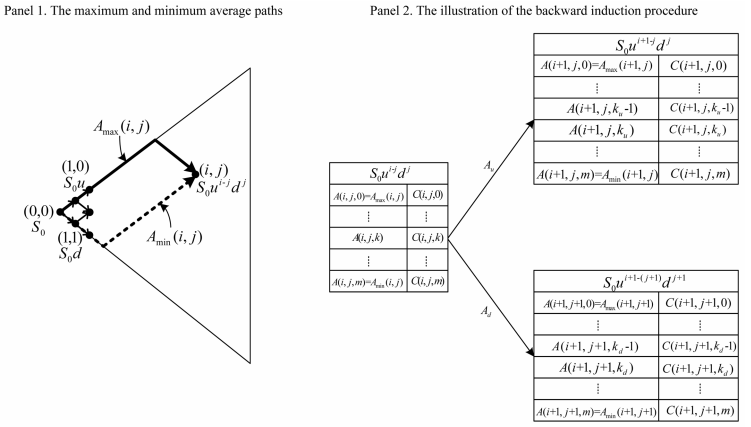

Fig. 2. The illustration of the Hull and White's model. In Panel 1, the $node(i,j)$ stands for the node at time point i with j cumulative down movements and the $S_0u^{i-j}d^j$ is the corresponding stock price. $A_{max}(i,j)$ ($A_{min}(i,j)$) is the maximum (minimum) average stock price among all possible paths from $node(0,0)$ to $node(i,j)$. In Panel 2, for each possible average price $A(i,j,k)$, it is necessary to find the corresponding A_u and A_d and then to derive the option values C_u and C_d by the piece-wise linear interpolation. The continuation value for $A(i,j,k)$ is $C(i,j,k) = (p \cdot C_u + (1-p) \cdot C_d)e^{-r\Delta t}$.

One of the most famous tree-based models to price arithmetic average options efficiently is proposed in [10]. In their algorithm, to avoid tracking all possible arithmetic average prices of each node, only the maximum and the minimum arithmetic average prices of all traversed paths for each node are calculated, which is illustrated in Panel 1 of Fig. 2.

For $node(i,j)$ with the stock price $S_0u^{i-j}d^j$ for $0 \le j \le i \le n$, the maximum arithmetic average price is contributed by a price path starting with $i-j$ consecutive up movements followed by j consecutive down movements, whose value is $A_{max}(i,j) = (S_0\frac{1-u^{i-j+1}}{1-u} + S_0u^{i-j}d\frac{1-d^j}{1-d})/(i+1)$. Likewise, the value of the corresponding minimum arithmetic average price can be calculated from a price path starting with j consecutive down movements followed by $i-j$ consecutive up movements: $A_{min}(i,j) = (S_0\frac{1-d^{j+1}}{1-d} + S_0d^ju\frac{1-u^{i-j}}{1-u})/(i+1)$. Once equipped with the knowledge about the maximum and minimum arithmetic average prices for each node, the logarithmic space between $A_{max}(i,j)$ and $A_{min}(i,j)$ is divided into m equal-length sub-intervals and $m+1$ representative average prices are obtained via $A(i,j,k) = \exp\left(\frac{m-k}{m}\ln(A_{max}(i,j)) + \frac{k}{m}\ln(A_{min}(i,j))\right)$.

After building the tree and the table of representative average prices for each node, we decide the payoff of each representative average price of the nodes at maturity first. Next, the option value is derived via the backward induction procedure. The backward induction procedure from $node(i+1,j)$ and $node(i+1,j+1)$ to $node(i,j)$ is illustrated in Panel 2 of Fig. 2.

For $A(i,j,k)$, the evolutions of the arithmetic average price at the next time point are $A_u = [(i+1)A(i,j,k)+S_0u^{i+1-j}d^j]/(i+2)$, and $A_d = [(i+1)A(i,j,k)+$

$S_0 u^{i+1-(j+1)} d^{(j+1)}]/(i+2)$. Suppose that A_u is inside the range $[A(i+1,j,k_u), A(i+1,j,k_u-1)]$. The option value C_u for the arithmetic average price A_u is approximated by the linear interpolation $C_u = w_u C(i+1,j,k_u) + (1-w_u)C(i+1,j,k_u-1)$, where $w_u = (A(i+1,j,k_u-1)-A_u)/(A(i+1,j,k_u-1)-A(i+1,j,k_u))$. Similarly, the option value of C_d for the arithmetic average price A_d is derived from $C_d = w_d C(i+1,j+1,k_d) + (1-w_d)C(i+1,j+1,k_d-1)$, where $w_d = (A(i+1,j+1,k_d-1) - A_d)/(A(i+1,j+1,k_d-1) - A(i+1,j+1,k_d))$, if A_d is inside the range $[A(i+1,j+1,k_d), A(i+1,j+1,k_d-1)]$. As a consequence, the continuation value for $A(i,j,k)$ is $C(i,j,k) = (p \cdot C_u + (1-p) \cdot C_d)e^{-r\Delta t}$.

Some Modifications for the Hull and White's Model
The interpolation error is inevitable in the Hull and White's model due to the limited number of representative average prices at each node and employing the piece-wise linear interpolation to find option values for nonexistent average prices. The brute-force method via increasing the number of representative average prices for each node is able to enhance the accuracy for the option values of course, but meanwhile it is accompanied with unacceptable computation time. In Section 4, in addition to the Hull and White's model, the performance of some modifications, including inserting the strike price into the average price table, applying the quadratic interpolation, and tightening the range for representative average prices,[1] will be compared to that of our adaptive placement method.

3 Our Models

The goal of our adaptive placement method is to intelligently reduce the interpolation error for pricing arithmetic average options in the tree-based model. Motivated by the common idea of dealing with the nonlinearity error in the Figlewski and Gao's adaptive mesh model [7] and the adaptive quadrature numerical integration method [6], our method differs from the Hull and White's method in the sense that more representative average prices are placed in the range where the option value function is with higher degree of curvature and fewer representative average prices are placed in the range where the option value function is with lower degree of curvature (see Fig. 1).

[1] For European fixed-strike-price arithmetic average calls, according to [1], for nodes at time point i, if some average price A is larger than the upper bound $(n+1)X/(i+1)$, because this path is sure to be in the money at maturity, the corresponding expected option value can be calculated directly via

$$e^{-r(n-i)\Delta t}[(i+1)A - (n+1)X + S_0 u^{i-j} d^j e^{r\Delta t} \frac{1-e^{r(n-i)\Delta t}}{1-e^{r\Delta t}}]/(n+1).$$

Therefore, the range $[A_{min}(i,j), A_{max}(i,j)]$ can be curtailed to $[\min(A_{\min}(i,j), (n+1)X/(i+1)), \min(A_{\max}(i,j), (n+1)X/(i+1))]$, and whenever the average price is above the upper bound, the corresponding expected option value can be derived via the above equation without any interpolation error.

The details of our adaptive placement method are elaborated as follows. For any $A \in [A_1, A_2]$, where A_1 and A_2 stand for any pair of adjacent representative average prices in the table of average prices for some node, the option value for A can be derived from the linear interpolation $C = C_1 \cdot \frac{A-A_2}{A_1-A_2} + C_2 \cdot \frac{A-A_1}{A_2-A_1}$, where C_1 and C_2 are corresponding option values for A_1 and A_2. By the mean-value theorem, the error term of the linear interpolation caused from the nonlinearity of the option value function in $[A_1, A_2]$ can be expressed as

$$\frac{C''(\xi)}{2!} \cdot (A - A_1) \cdot (A - A_2), \text{ for some number } \xi \text{ between } A_1 \text{ and } A_2. \quad (1)$$

Our adaptive placement method is designed to examine whether the linear interpolation error between each pair of adjacent representative average prices in Eq. (1) is below some pre-specified limit. Once the error of the linear interpolation inside the range of $[A_1, A_2]$ is not negligible, i.e. $C''(\xi)$ is too large or the distance between A_1 and A_2 is too far, we divide $[A_1, A_2]$ into finer subsets by inserting an extra representative average price inbetween and then repeat the same procedure of examining the error of the linear interpolation for each subset. Once the value of the error term between any pair of adjacent representative average prices is smaller than the predefined threshold (termed the *second order error criterion* in our method), this examining-and-dividing process is stopped.

In practice, another constant termed the *precision criterion* is also defined to represent the threshold of negligible refinement for both average prices and option values in our method. The above examining-and-dividing process is also terminated when the difference between adjacent representative average prices or their corresponding option values is smaller than this minimum criterion. The purpose of introducing the *precision criterion* is to prevent possibly infinite dividing caused from the non-differentiable point. Within each examination of the linear interpolation error, we approximate $C''(\xi)$ in Eq. (1) by the second order numerical differentiation. For any pair of adjacent representative average prices A_1 and A_2, the midpoint $A = (A_1 + A_2)/2$ is employed together to approximate the error term of the linear interpolation for this range.

The steps to price arithmetic average options in this paper are described as follows. During the phase of building the stock price tree, only the maximum and minimum average prices for each node are recorded as representative average prices. Meanwhile, we also determine whether the strike price is needed to be inserted into the range between the maximum and the minimum average prices. As a consequence, there will be two or three representative average prices for each node after building the stock price tree. Since the number of possible arithmetic average prices for the nodes at the initial three time steps is not larger than three, it is not necessary to perform the above procedure and instead we record all possible average prices for these nodes.

After building the stock price tree, the tables of representative average prices for all nodes are mainly constructed during the phase of backward induction We take an example to illustrate the examining-and-dividing process of our adaptive placement method step by step. Suppose $S_0 = X = 50$, $n = 40$, $T = 1$, $r = 10\%$, $\sigma = 80\%$, and both the *second order error criterion* and the

Initial table for *node(37,25)*

Average	Call
83.4062	28.1577
50.0000	0.0000
12.3309	0.0000

Step 1

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	**13.2048**	0.8740	(inserted)
50.0000	0.0000		
12.3309	0.0000		

Step 2

Average	Call	Interpolation error	
83.4062	28.1577		
75.0546	**20.6813**	3.2452E-15	(not inserted)
66.7031	13.2048		
50.0000	0.0000		
12.3309	0.0000		

Step 3

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
58.3515	**5.7284**	0.8740	(inserted)
50.0000	0.0000		
12.3309	0.0000		

Step 4

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
62.5273	**9.4666**	3.7088E-15	(not inserted)
58.3515	5.7284		
50.0000	0.0000		
12.3309	0.0000		

Step 5

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
58.3515	5.7284		
54.1758	**2.1788**	0.6853	(inserted)
50.0000	0.0000		
12.3309	0.0000		

Step 6

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
58.3515	5.7284		
56.2637	**3.8593**	9.4341E-02	(not inserted)
54.1758	2.1788		
50.0000	0.0000		
12.3309	0.0000		

Step 7

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
58.3515	5.7284		
54.1758	2.1788		
52.0879	**0.7221**	0.3673	(not inserted)
50.0000	0.0000		
12.3309	0.0000		

Step 8

Average	Call	Interpolation error	
83.4062	28.1577		
66.7031	13.2048		
58.3515	5.7284		
54.1758	2.1788		
50.0000	0.0000		
31.1655	**0.0000**	0.0000	(not inserted)
12.3309	0.0000		

Final table for *node(37,25)*

Average	Call
83.4062	28.1577
66.7031	13.2048
58.3515	5.7284
54.1758	2.1788
50.0000	0.0000
12.3309	0.0000

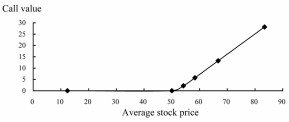

Fig. 3. The numerical example for the examining-and-dividing process. The values of parameters in this example are $S_0 = X = 50$, $n = 40$, $T = 1$, $r = 10\%$, $\sigma = 80\%$, and the *second order error criterion* and the *precision criterion* are both 0.5. These figures illustrate the examining-and-dividing process for $node(37, 25)$. Once the approximate linear interpolation error is larger than 0.5, the pair of the average price and the call value in boldface will be inserted into the table of representative average prices of $node(37, 25)$. The approximate linear interpolation error for any pair of adjacent representative prices in the final table is bounded by the *second order error criterion.*

precision criterion are 0.5. For $node(37, 25)$, the examining-and-dividing process is sketched in Fig. 3. Inside the frame of each step, there are three pairs of representative average prices and the corresponding call values, and we also report the linear interpolation error when these three pairs of representative average prices and option values are considered.

When the backward induction progresses to $node(37, 25)$, only 83.4062, 50, and 12.3309 are in the initial table of representative average prices, and their corresponding option values are 28.1577, 0, and 0 respectively. In step 1, the pairs of (83.4062, 28.1577), ((83.4062+50)/2=66.7031, 13.2048), and (50, 0) are considered to approximate the linear interpolation error for the range between 83.4062 and 50. Because the approximate linear interpolation error 0.8740 is larger than the *second order error criterion*, the pair of the average price 66.7031 and the corresponding call value 13.2048 should be inserted into the table of representative average prices. In step 2, the approximated linear interpolation error in the range between 83.4062 and 66.7031 is 3.2452E-15, which is smaller than the *second order error criterion* 0.5. Therefore, we do not insert the pair of the average price 75.0546(=(83.4062+66.7031)/2) and the corresponding option value 20.6813 since the linear interpolation works pretty well in the range between 83.4062 and 66.7031. Following the same reasoning, we can derive the final table of representative average prices and their corresponding option values through steps 3 to 8, in which the approximate linear interpolation error for any pair of adjacent representative average prices is below the *second order error criterion.*

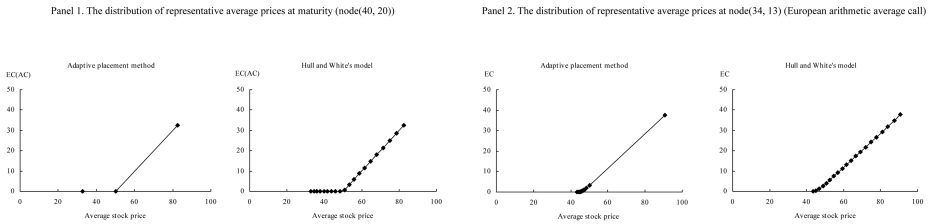

Fig. 4. Comparisons of the distributions of representative average prices of the adaptive placement method and the Hull and White's model. For the readability of this figure, the values of parameters are specified as: $S_0 = X = 50$, $n = 40$, $T = 1$, $r = 10\%$, $\sigma = 30\%$, the *second order error criterion* is 0.01, and the *precision criterion* is 0.001. In addition, the number of representative average prices in the Hull and White's model is 20.

In addition, the option value as the function of the arithmetic average price is plotted in Fig. 3. Since the performance of the piece-wise linear interpolation is poor around where the option value function is with high degree of curvature, our algorithm places more representative average prices in these areas to reduce the linear interpolation error. On the other hand, due to the satisfactory performance of the piece-wise linear interpolation for dealing with the option value function with low degree of curvature, our algorithm argues that less representative average prices placed in those areas will be sufficient.

4 Numerical Results

Comparisons with the Hull and White's Model

The differences between the logarithmically equally-spaced placed rule in the Hull and White model [10] and the adaptive placement rule in our model are shown in Fig. 4. In Panel 1 of Fig. 4, for some node at maturity, it is easily found that there is no linear interpolation error for both linear segments, and therefore it is not necessary to insert any representative average price. However, the Hull and White's model still employs $m + 1$ representative average prices for each node at maturity. For each linear segment, our method provides the interpolated results as accurate as those in the Hull and White's model, but near the kink, our method will outperform the Hull and White's model except the strike price happens to be one of representative average prices in their model.

In Panel 2 of Fig. 4, it is clear that the logarithmically equally-spaced placement in the Hull and White's model places too many representative average prices on the region with low degree of curvature, but only a few representative average prices are needed in our adaptive placement method to derive interpolated results with sufficient accuracy in this region. On the contrary, to deal with regions with high degree of curvature, the Hull and White's model generates unexpected large pricing errors due to large interpolation error.

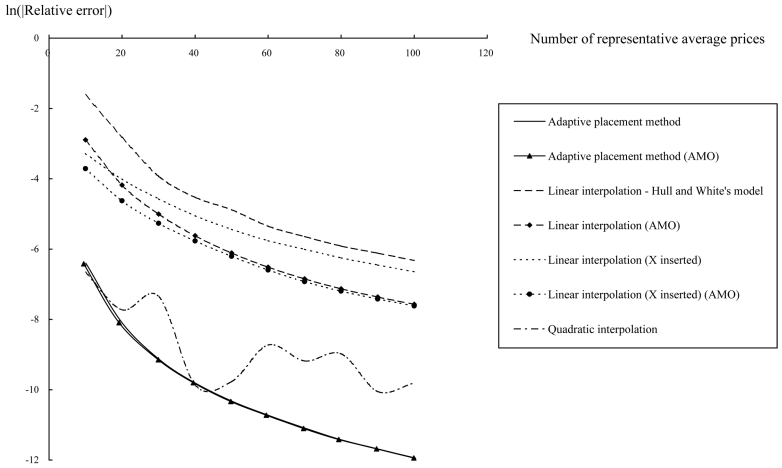

Fig. 5. The rates of convergence of seven different methods of pricing arithmetic average calls. Note that our adaptive placement method converges faster than other methods with respect to the number of representative average prices. Note also that except our adaptive placement method, the convergence rates of other methods are improved when the algorithm of AMO is applied.

Convergence Rates with the Number of Representative Average Prices This section compares the rate of convergence with respect to the number of representative average prices for different methods. The values of parameters in our numerical example are as follows: $S_0 = X = 50$, $T = 1$, $r = 10\%$, $\sigma = 80\%$, $n = 40$, and different numbers of representative average prices are examined. In order to obtain a better understanding of the rates of convergence, an analysis on the relative error and the number of representative average prices is performed. Plots of ln(|relative error|) in relation to the number of representative average prices for the arithmetic average calls are in Fig. 5.

Obviously, the Hull and White's model converges poorly, but the relative error decreases significantly when the strike price X is inserted as a representative average price. This is because for nodes near maturity, the kink is near where the average price is equal to the strike price, and the piece-wise linear interpolation is inclined to overestimate the option value around the kink.

Note that the improvement is minor when combining the AMO algorithms [1] with our adaptive placement method. The idea of the AMO algorithm is to derive the option values of the arithmetic average prices higher than some threshold without incurring any interpolation error and meanwhile concentrate representative average prices on a smaller range to further reduce the interpolation error. However, for the region above the threshold, the interpolation error is in fact very small, so our adaptive placement method already places fewer representative average prices in the region above the threshold, and automatically concentrates on dealing with the region below the threshold. Thus the concept of the AMO algorithm is already nested in our adaptive placement method.

5 Conclusion

This paper proposes the adaptive placement method to price arithmetic average options. In our method, the representative average prices are placed effectively to reduce the interpolation error. Numerical results show that our adaptive placement method is superior to other methods in reducing interpolation error. Thus this method can be employed to price arithmetic average options efficiently and accurately. In fact, this novel technique can be applied to any other algorithms with augmented state variables and the piece-wise linear interpolation approximation like GARCH option pricing algorithm in [15].

References

1. Aingworth, D., Motwani, R., Oldham, J. D.: Accurate Approximations for Asian Options. Proc. 11th Annual ACM-SIAM Symposium on discrete Algorithms (2000)
2. Carverhill, A. P., Clewlow, L. J.: Valuing Average Rate (Asian) Options. Risk 3 (1990) 25-29
3. Cho, H. Y., Lee, H. Y.: A Lattice Model for Pricing Geometric and Arithmetic Average Options. J. Financial Engrg. 6 (1997) 179-191
4. Dai, T.-S., Huang, G.-S., Lyuu, Y.-D.: An Efficient Convergent Lattice Algorithm for European Asian Options. Appl. Math. and Comput. 169 (2005) 1458-1471
5. Dai, T.-S., Lyuu, Y.-D.: An Exact Subexponential-Time Lattice Algorithm for Asian Options. Proc. ACM-SIAM Symposium on Discrete Algorithms (SODA04) New Oreleans (2004) 710-717
6. Faires, J. D., Burden, R.: Numerical Methods. 3rd edn. Brooks/Cole Press. (2003)
7. Figlewski, S., Gao, B.: The Adaptive Mesh Model: A New Approach to Efficient Option Pricing. J. Financial Econ. 53 (1999) 313-351
8. Forsyth, P. A., Vetzal, K. R., Zvan, R.: Convergence of Numerical Methods for Valuing Path-Dependent Options Using Interpolation. Rev. Derivatives Res. 5 (2002) 273-314
9. Geman, H., Yor, M.: Bessel Processes, Asian Options, and Perpetuities. Math. Finance 3 (1993) 349-375
10. Hull, J. C., White, A.: Efficient Procedures for Valuing European and American Path-Dependent Options. J. Derivatives 1 (1993) 21-31
11. Kemna, A., Vorst, T.: A Pricing Method for Option Based on Average Asset Values. J. Banking and Finance 14 (1990) 113-129
12. Klassen, T. R.: Simple, Fast and Flexible Pricing of Asian Options. J. Comput. Finance 4 (2001) 89-124
13. Milevsky, M. A., Posner, S. E.: Asian Option, the Sum of Lognormals and the Reciprocal Gamma Distribution. J. Financial and Quant. Anal. 33 (1998) 409-422
14. Neave, E. H., Turnbull, S. M.: Quick Solutions for Arithmetic Average Options on a Recombining Random Walk. Actuarial Approach For Financial Risks 2 (1994) 717-740
15. Ritchken, P., Trevor, R.: Pricing Options under Generalized GARCH and Stochastic Volatility Processes. J. Finance. 54 (1999) 377-402
16. Turnbull, S. M., Wakeman, L. M. A Quick Algorithm for pricing European Average Option. J. Financial and Quant. Anal. 26 (1991) 377-389