複雜金融衍生性商品之計價演算法研究(I)

計劃編號：NSC 90-2213-E-002-081

執行期限：90/08/01—91/07/31

主持人：呂育道　臺灣大學資訊工程系教授

一、中文摘要

衍生性金融商品為一種價值決定於某基本資產(如股票)的金融商品，其定價問題等於計算隨機變數之期望值。路徑相關型衍生性金商品(path-dependent derivative)的價值取決於基本資產價格之全部或部分歷史，它們的定價通常是複雜的計算問題。算術平均式選擇權又稱亞洲式選擇權，其價值取決於基本資產價格之算術平均。本計劃研究亞洲式選擇權之訂價問題。我們的結果如下: (1)一個有效率的近似演算法，(2)此演算法保證收斂。

英文摘要

Financial derivatives are financial instruments whose payoff depends on some underlying asset such as stock. Mathematically speaking, pricing such instruments amounts to calculating the expected value of a random variable. Path-dependent derivatives are derivatives whose payoff depends on the *history* of the underlying asset's price. Pricing is computationally non-trivial for such derivatives. Arithmetic average-rate option is also called Asian option. The payoff of an Asian option depends on the *arithmetic average* of the asset prices. This proposal investigates the pricing of Asian options. Our findings are: (1) an efficient approximation pricing algorithm and (2) guarantee that the algorithm converges to the correct value.

二、計劃緣由與目的

Derivatives are financial instruments whose payoff depends on some **underlying asset** such as stock. Mathematically speaking, pricing such products amounts to calculating the expected value of a random variable. Such financial products represent a quantum leap for the global economy as they provide tools for manage financial risks. The notional market size of derivatives is in the trillion US dollars range. Research in this area therefore has important practical implications and has been extremely active.

Path-dependent derivatives are derivatives whose payoff depends on the *history* of the underlying asset's price. The price is usually described as a **stochastic process**. For our purpose, we will take the asset prices to be a time series $S_0, S_1, \ldots, S_n$, where the price S_i is a lognormal random variable with a known mean and variance. Pricing is usually computationally non-trivial for path-dependent derivatives. Research in this area represents the frontier in **computational finance** and **financial economics**.

The purpose of our proposal is the pricing of the prominent path-dependent derivative: the **Asian option**. Polynomial-time pricing algorithms are not available for this option and in fact are not expected to be available. The purpose of this proposal is therefore efficient approximation algorithms that converge to the true price.

The payoff of an Asian option is $\max\left(\frac{\sum_{i=0}^{n} S_i}{n+1} - X, 0\right)$. Although there are many variations, an algorithm that can handle this option can usually be revised only slightly to handle others.

A tremendous amount of research has been carried out to attack the pricing problem of Asian options. It can be said that this is one of the most important problems in computational finance and financial economics [Lyuu02].

In the **analytical approach**, the goal is to come up with a closed-form or semi-closed-form formula. The problem with this approach is two-fold. First, none of the formula in the literature is exact. Hence, they will fail on some input almost surely and the accuracy cannot be improved, being a formula. Second, some Asian options can be **American-style**, which means they can be terminated at any time. Formulae cannot handle this early exercise feature.

In the **Monte Carlo simulation** approach, simulation of the asset price is conducted with the resulting payoffs then averaged. This method is quite general and when coupled with **variance-reduction techniques** and **quasi-Monte Carlo sequences** can be quite effective. The difficulty with this methodology is again two-fold. First, the computed result has no a priori guarantee. Even though the computed value can be proved to converge to the true value as n goes to infinity, the random noise means only *stochastic* bounds are available. Second, again, it cannot handle early exercise.

Now we come to the **approximation-algorithm** approach [Hoch97]. *Exact* values can often be computed given $O(2^n)$ or $O(3^n)$ time bound. The complexity makes such algorithms useless. Our purpose is then to design an efficient approximation algorithm that gives *good* results. Work in this area is surveyed in [Lyuu02].

We intend to design approximation algorithms that are efficient in practice and that converge weakly to the exact value.

三、實驗方法

The major contribution of our result is a novel trinomial lattice that is general-purpose but most useful for pricing Asian options. Pricing on the lattice is efficient and accurate. Furthermore, unlike most other schemes, interpolation is not needed in backward induction. The algorithm is therefore an exact discrete-time algorithm. This characteristic is in sharp contrast to existing discrete-time algorithms; these algorithms attempt to approximate the naive but convergent exponential-time algorithm, which simply evaluates each of the 3^n paths on an n-period trinomial model. Our algorithm can price both European- and American-style options. Convergence to the continuous-time value is guaranteed as the lattice matches the first and second moments of the continuous-time model at each node [Duffie96]. Such theoretical guarantee is lacking in many other approximation schemes.

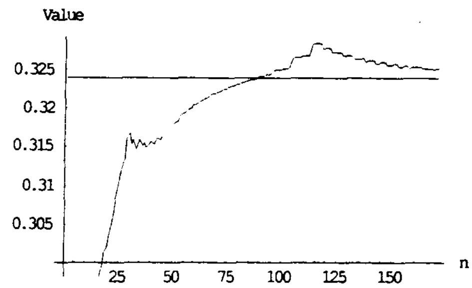
The idea is deceptively simple. It is well-known that the option value is homogeneous of degree one in the asset price. We can thus multiply the exercise price and all the asset prices on the lattice by some number x, price the option, and finally divide the calculated option value by x. This, together with the extra degrees of freedom afforded by the trinomial model, is exploited to construct a trinomial lattice with *integral* asset prices. This means that the price sum of any path on the lattice will be integers as well. (Recall that the payoff is determined by the price sum.) The key insight can now be stated: The price sums of paths reaching any given node are finite in number and *enumerable*, being integers between some integral minimum and maximum price sums. The integral property will eventually allow backward induction to dispense with approximations. The algorithm is thus exact.

In practice the algorithm will not really be multiplying asset prices on the lattice as there exists a more efficient, yet equivalent, implementation. By insisting that the multiplication factor x be a power of two, multiplication amounts to shifting the prices by j bits to the left. Thus asset prices will simply be rational numbers of finite precision.

It is not necessary to apply the extension of precision mentioned above to all the lattice nodes. When a node N has extended precision, those nodes reachable from it need to have extended precision as well. The reason is that a path passing through node N adds N's price, which has extended precision, to the price sum. On the other hand, nodes not reachable from node N need not have the same precision as N. This observation will be employed to reduce the complexity further and result in nodes with varying precisions, thus the term **multiresolution** (MR).

Nothing is gained with the above manipulations unless the number of states is reduced from exponential to a much more manageable number. It turns out that limiting the stock prices to finite-precision rational numbers does drastically reduce the possible number of price sums. For example, with 160 time periods, the total number of paths to the middle node at expiration is the astronomical 10^{74}. No computers are expected to face down this number now, or in the future. But the total number of price sums at that node is at most 57887 under the typical scenario Thus a lot of paths produce the same price sums on the MR lattice. That is why the MR algorithm can price Asian options in the case of 160 periods. We are led to conclude that the MR algorithm has broken the exponential barrier while remaining an exact algorithm. It may be the first exact discrete-time algorithm to break the exponential barrier [Dai02].

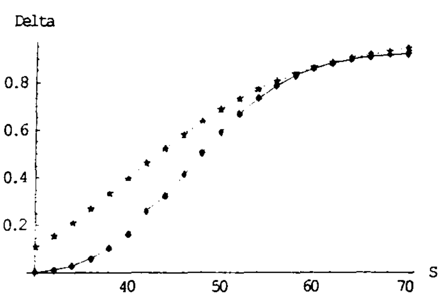
Extensive computer experiments suggest that the MR algorithm is superior to the approaches in the literature sampled above in terms of both convergence and accuracy. It is also a practical algorithm. Experiments suggest that the algorithm runs in time $3^{\sqrt{n}}$. It quickly converges to the continuous-time value (see the figure that follows). Sensitivity measures such as delta are also straightforward to compute with the MR algorithm. Hence hedging the Asian option presents no fundamental problems.



四、結論與討論

The running time of the MR algorithm is reasonable and much less than 3^n. For example, the algorithm finishes in 145 seconds for n=141, whereas 3^{141} is about 10^{67}, making the naive 3^n-time algorithm hopeless.

Delta is key to hedging and replication. It is therefore important that the MR algorithm compute the option delta accurately. The computing of delta is a straightforward by-product of option pricing in the MR algorithm. The figure below shows further the numerical delta as determined by the MR algorithm varies smoothly with asset price. We therefore do not expect problems in constructing hedge portfolios.



With n=30, the running time is about 2 seconds on an Intel Pentium II 233MHz computer. Most of the values computed by the MR algorithm are close to the value computed by Monte Carlo simulation.

We conclude that MR's practical running time means that it may be the first exact algorithm to break the exponential barrier. The source of the tremendous reduction in running time is the dramatic decrease in the possible number of price sums. Future work will look into whether the MR algorithm can be made to run in polynomial time, provably.

五、參考文獻

[Dai02] Tian-Shyr Dai and Yuh-Dauh Lyuu. ``Efficient, Exact Algorithms for Asian Options with Multiresolution Lattices." Accepted by *Review of Derivatives Research*, April 2002.

[Duffie96] Duffie, Darrell. *Dynamic Asset Pricing Theory*. 2nd ed. Princeton, New Jersey: Princeton University Press, 1996.

[Hoch97] Hochbaum, Dorit S. (Ed.) *Approximation Algorithms for NP-Hard Problems*. Boston: PWS, 1997.

[Lyuu02] Lyuu, Yuh-Dauh. *Financial Engineering & Computation: Principles, Mathematics, and Algorithms*. Cambridge: Cambridge University Press, 2002.