

# バケッティングを用いた格子モデルにおける アジアンオプションの価格評価

中島 宏  
指導教員 藤井 眞理子教授

## 概要

格子モデルを用いたアジアンオプションのアルゴリズム評価を行った。アジアンオプションとは各ステップの原資産価格に依存するオプションである。格子モデルを用いてアジアンオプションを評価しようとすると、計算時間は指数的に増大するので、ある一定以上の観測数における厳密解を出すのは現実的に困難である。そこでバケッティングを用いた近似アルゴリズムが提案されている。今回の研究では改良型の格子モデルに近似アルゴリズムを適用したところ、従来のアルゴリズムより誤差が少ない結果が得られた。

## 1 はじめに

オプションとは満期までの原資産価格の変動によってペイオフが確定する派生商品である。オプションには大きく分けてコール、プットの2種類がある。コールオプションはある定められた日(満期)に定められた価格(行使価格)で原資産を購入する権利、プットオプションはある定められた日に定められた価格で原資産を売却する権利である。今回の研究で扱うアジアンオプションはある期間の原資産価格の平均値にペイオフが依存するオプションである。

本論文で扱うアジアン・コールオプションのペイオフ  $A(T)$  は現時点を  $t=0$ 、オプションの満期を  $T$ 、行使価格を  $X$  として観測数  $N$  個の原資産価格  $S(t)$  の平均に依存するとすると

$$A(T) \equiv \max\left(\left(\frac{1}{N} \sum_{i=1}^N S(t_i) - X\right), 0\right) \quad (1)$$

となる。今回の研究では特に断らない限り  $t_i = i \cdot (T/N) = i\Delta t$  として考える。

オプション価格を計算する場合、一般的に原資産価格が対数正規分布に従うと仮定する。しかし、この仮定のもとではアジアンオプションの価格は解析的には求めることができない。そこで価格を求める手法として格子モデル、モンテカルロ法、有限差分法等、様々な計算方法が提案されている。今回は格子モデルを用いて価格評価を行った。

格子モデルで最も単純なモデルが二項モデルである。現時点での原資産価格を  $S_0$ 、満期までの期間を  $T$ 、期間の分割数を  $N$  とする。以下では期間の分割数と原資産価格の観測数が等しい場合を考える。二項モデルにおいて  $\Delta t (= T/N)$  期間後の原資産価格は、現時点 ( $t=0$ ) の原資産価格が  $S_0$  であるとするば確率  $p$  で  $uS_0$ 、確率  $q (= 1-p)$  で  $dS_0$  になる。以下、二項モデルにおいてノード  $(i, j)$  は  $i\Delta t$  において通算  $j$  回下降したノードであるとする。ノード  $(i, j)$  における原資産価格は  $u^{i-j} d^j S_0$  となる。

## 2 アジアンオプションのプライシング

アジアンオプションの場合、ペイオフが原資産の累積和に依存する。よって格子モデルを用いて厳密に価格評価を行う場合、パスごとに価格評価を行う必要がある。しかし、観測数が増えればパスの数は指数的にふえるため、計算時間が指数的に増えてしまう。普通のPCだと  $N = 30 \sim 40$  程度が限界である。この事より  $N$  が大きい場合、格子モデルにおける厳密解を求めるのは現実的ではない。そこで Aingworth et al[1] はバケッティングによって累積和をいくつかの代表値で近似するアルゴリズム(以下 AMO アルゴリズムと略す)を提案した。

ここでいうバケッティングとは、原資産価格の累積和を適当なバケット(区間)で近似する手法のことである。図1は切り上げバケッティングの様子を表している。全区間  $[0, 200]$ 、バケット幅 50 の場合の切り上げバケッティングの

上限 下限	(累積和, 確率)	⇒ 累積和を近似して 確率を足し合わせる	上限 下限	(近似値, 確率)
200			200	
150			150	
150	(140, 0.05)		150	
100	(120, 0.1)		100	(150, 0.15)
100	(100, 0.1)		100	
50	(80, 0.2)		50	(100, 0.3)
50			50	
0	(40, 0.1)		0	(50, 0.1)

図 1. バケッティングの様子

様子を示している。左側では区間  $[100, 150]$  に累積和 140, 120 が含まれている。どちらの値も区間の最高値 150 に切り上げられる。切り上げバケッティング後の 150 の確率は 140 の確率 0.05 と 120 の確率 0.10 を足し合わせた 0.15 となる。この作業を全ての区間において行っている。

原資産の累積和が  $NX$  を超えているときの満期におけるペイオフの期待値  $\left(\frac{1}{N} \sum_{i=1}^N S(i\Delta t) - X\right)$  は現在のノード

$S(i+1, j)$  と  $(i+1)\Delta t$  までの原資産の累積和  $\sum_{j=1}^{i+1} S(j\Delta t)$  が分かっているならば簡単に算出することが出来る。

表 1 が AMO アルゴリズムの擬似コードである。AMO アルゴリズムでは累積和が  $NX$  を超えている場合はペイオフの期待値を算出し、ペイオフの期待値とバケットの持つ確率を掛けたものを満期におけるペイオフの期待値に足し合わせる。累積和が  $NX$  に達していない場合は、対応するバケットに確率を足しこむ。この操作を  $t = 0, \Delta t, \dots, T$  という順番で全てのノードにおいて行う。

AMO アルゴリズムの近似誤差について考える。各時点における近似誤差を  $|D_i|$  とすると全体での近似誤差は  $\frac{1}{N-1} \sum_{j=1}^{N-1} |D_j|$  である。 $|D_i| < NX/k$  ( $k$ : ノードのバケット数) より全体での最大誤差は  $O(NX/k)$  となる。

Dai et al[3] は総バケット数が一定の時にノードへの割り当てバケット数を最適化させることで最大誤差が  $O(\sqrt{NX}/k)$  となるアルゴリズム(以下 DHL アルゴリズムと略す)を提案した。具体的には近似誤差はノード  $(i, j)$  における誤差を

$|D_{ij}|$  とすると全体の近似誤差は  $\frac{1}{N-1} \sum_{i=1}^{N-1} \sum_{j=0}^i \binom{i}{j} p^{i-j} (1-p)^j |D_{ij}|$  となる。 $(i, j)$  におけるバケット数  $k_{ij}$  が

$$k_{ij} = \text{総バケット数} \times \frac{\sqrt{\binom{i}{j} p^{i-j} (1-p)^j}}{\sum_{0 \leq j \leq i \leq N-1} \sqrt{\binom{i}{j} p^{i-j} (1-p)^j}}$$

するとき全体の誤差は最小になり  $O(\sqrt{NX}/k)$  となる。

## 3 格子モデル

通常の二項モデルは分割数(観測数)が大きくなった場合、非効率であることが知られている。経過時間を  $t$  とすると原資産の分布の標準偏差は  $\sqrt{t}$  に比例する。一方、通常の格

表 1. AMO アルゴリズムの擬似コード

$S(i, j)$ : ノード  $(i, j)$  における原資産価格,  $B(i, j, h)$ : ノード  $(i, j)$  の  $h$  番目のバケット  $w(i, j, h)$ :  $B(i, j, h)$  の確率,  $k$ : 各ノードにおけるバケット数,  $X$ : 行使価格,  $p$ : リスク中立確率,  $k$ : 各ノードにおけるバケット数,  $u$ : 上昇幅,  $d$ : 下降幅,

begin

満期での期待ペイオフ  $\Phi$  に初期値 0,  $S(i, j) = S_0 u^{i-j} d^j$ ,  
バケット  $B(0, 0, 0)$  の確率  $w(0, 0, 0)$  に初期値 1,  
それ以外の全ての確率に 0 を代入 .

for all  $i = 0, \dots, n-1$

for all ノード  $(i, j) (j = 0, \dots, i)$

for all バケット  $B(i, j, h) (h = 0, \dots, k)$

if  $\frac{NX}{k} h + S(i+1, j) < NX$

バケット  $B(i+1, j, b)$  の確率  $w(i+1, j, b)$  に  
 $w(i, j, h) \cdot p$  を加算

ただし  $b = \lceil \frac{(NX/k) \cdot h + S(i+1, j)}{NX/k} \rceil$

else

状態  $(S(i+1, j), \frac{NX}{k} h + S(i+1, j))$  における  
ペイオフの条件付期待値  $\Phi'$  を計算

$\Phi$  に  $\Phi' \cdot p \cdot w(i, j, h)$  を加算

if  $\frac{NX}{k} h + S(i+1, j+1) < NX$

バケット  $B(i+1, j+1, c)$  の確率  $w(i+1, j+1, c)$   
に  $w(i, j, h) \cdot (1-p)$  を加算

ただし  $c = \lceil \frac{(NX/k) \cdot h + S(i+1, j+1)}{NX/k} \rceil$

else

状態  $(S(i+1, j+1), \frac{NX}{k} h + S(i+1, j+1))$  に  
おけるペイオフの条件付期待値  $\Phi''$  を計算

$\Phi$  に  $\Phi'' \cdot (1-p) \cdot w(i, j, h)$  を加算

オプションの満期におけるペイオフの期待値として  $\Phi$  を出力  
end

[ $X$ ] は  $X$  以上の最小の整数を表す .

子モデルにおいては上昇幅, 下降幅が時間によらず一定なので原資産価格の分布幅は  $t$  に比例する . 分割数 (観測数) が大きい場合, 殆ど利用されないノードにおける価格評価をするために計算時間の大部分が費やされる事になってしまう .

Baule and Wilkens[2] は, ノード密度を変化させたリーンモデルを提案した . 図 2 がリーンモデルの例である . リーンモデルは中央部は通常の三項モデルとノード密度が変わらないが, あまり利用されない外縁部のノード密度を減らしている . ノード密度を減らす外縁部の規模を適切に選ぶことで, オプション価格は通常の三項モデルとほぼ変わらない値になる . また, 計算時間は三項モデルより短縮される .

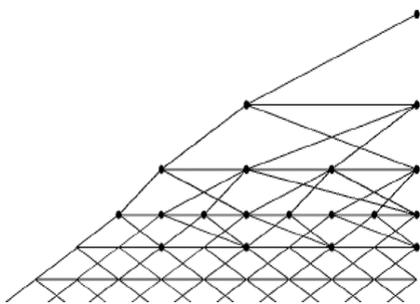


図 2. リーンモデル

## 4 数値計算結果

実装した格子モデルは二項モデル, 三項モデル, リーンモデルの 3 種類, アルゴリズムは AMO, DAL の 2 種類 . よって合計 6 種類の手法を比較した .

数値計算結果の基準値として, 今回はモンテカルロ法で得られた値を用いた . また, 二項モデルと三項モデルでは厳密値 (全てのパスを考慮して求めた値) が違うと思われ

る . 厳密値の差が大きければ二項モデルと三項モデルの近似アルゴリズムを比較することはできない . 今回はモンテカルロ法の結果から, 分割数が大きければ二項モデルと三項モデルの値に大きな差はないと判断した . また, リーンモデルにおいて粗い部分の規模を適切に選べば値はほぼ三項+AMO と一致し, 粗い部分の規模を減らしていけば三項モデルの値に収束することが確認できた .

総バケット数が同じであれば, どの格子モデルでも AMO より DHL のアルゴリズムの方が効率的であった . 各格子モデルに DHL のアルゴリズムを適用した結果が表 2, 図 3 である . モンテカルロ法で求めた値を真の値として, 相対誤差を求めた . 総バケット数が一定であればリーンモデルが最もモンテカルロ法との差が少ない . また計算時間でみてもリーンモデルが最も効率的であることが分かる .

表 2. 各格子モデル (DHL) の相対誤差

総バケット数	$10^7$	$2 \cdot 10^7$	$3 \cdot 10^7$	$4 \cdot 10^7$	$5 \cdot 10^7$
リーンモデル (時間)	4.17 (12.7)	2.16 (23.5)	1.43 (116)	1.07 (295)	0.85 (346)
三項モデル (時間)	9.50 (12.2)	4.71 (23.8)	3.12 (101)	2.35 (220)	1.88 (302)
二項モデル (時間)	8.18 (6.1)	4.08 (36.9)	2.73 (139)	2.06 (246)	1.65 (256)

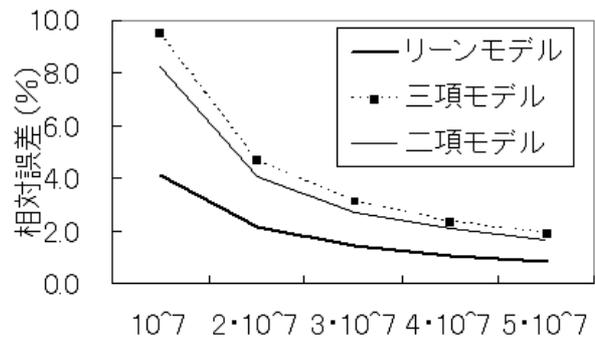


図 3. 各格子モデル (DHL) の相対誤差

## 5 まとめ

観測数が大きい場合, リーンモデルに DHL アルゴリズムを適用する方法が最も効率的であることが分かった . 今後の課題としてはアメリカン・アジアンオプションの価格評価等が考えられる .

## 参考文献

- [1] Aingworth.D, Morwani.R and Oldman.J, "Accurate approximations for Asian options", *Proc .11th ACM-SIAM Symposium on Discrete Algorithms*, 2000, pp891-900.
- [2] Baule.R, and Wilkens.M, "Lean Tree -A General Approach for Improving Performance of Lattice Models for Option Pricing", *Review of Derivative Research*, vol.7(2004), pp.53-72.
- [3] Dai.T.S, Huang.G.S and Lyuu.Y.D, "Extremely accurate and efficient tree algorithms for Asian options with range bounds", *2002 NTU International Conference on Finance, National Taiwan University, Taiwan*, 2002.