D－1

Black-Scholes 偏微分方程式の高速並列陰解法

Fast Parallel implicit Algorithm for Black-Scholes Equation

植松 郁哉　　李 磊
Ikuya UEMATSU　　Lei LI
法政大学大学院 理工学研究科
Graduate School of Science and Engineering, Hosei University

1. はじめに

近年の金融市場では,取引の高速化によって短時間の間に数多くの取引が行われる.したがって,計算が必要な金融商品についてはより高速で算出を行う必要が生じている.本稿では並列アルゴリズムを用いることによって金融商品のひとつであるオプションの高速価格算出手法を提案する.

2. Black-Scholes 偏微分方程式

Black- Scholes 偏微分方程式とは一定の条件下においてオプションの価格を求めることのできる偏微分方程式である.

$$\frac{\partial V}{\partial t}+\frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}+\mathrm{r}S\frac{\partial V}{\partial S}-rV=0 \qquad (1)$$

このとき V:オプション価格, σ :原資産ボラリティ, S:資産価値,T:満期時点,r:利子率である.

3. 計算手法

Black-Scholes 偏微分方程式は有限差分法によって解の算出を行うことができるが,オプションは金融商品であるため,安定であることが望ましく,陰解法によって算出を行うのが妥当である.

陰解法を用いて有限差分法を解く際には三対角 Toeplitz 型連立一次方程式 (2)を解く必要があり,この行列式を高速に計算するためにさまざまな手法が提案されている[1].

$$\begin{pmatrix} b & c & & & \\ a & b & c & & \\ & \ddots & \ddots & \ddots & \\ & & a & b & c \\ & & & a & b \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{pmatrix}=\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix} \qquad (2)$$

4. 提案手法

三対角 Toeplitz 行列に対する並列アルゴリズム[2]を用いることによって方程式の計算を行う.

式(2)の三対角 Toeplitz 行列を含む方程式を

$$\mathrm{Au}=\mathrm{f} \qquad (3)$$

とする.

行列 A に対して式(4)のように分解を行う

$$\mathrm{A}=\alpha[LU+\beta\gamma e_1 e_1{}^T] \qquad (4)$$

このとき,$e_1=(1,\quad 0,\quad \cdots\quad ,0)^T$

$L=Tridiag\{\beta,1,0\},\quad U=Tridiag\{0,1,\gamma\}$

$\alpha=\frac{1}{2}\left(b+\sqrt{b^2-4ac}\right),\ \beta=\frac{a}{\alpha},\ \ \gamma=\frac{c}{\alpha}$ である.

ここで$\mathrm{d}=\frac{f}{\alpha}, M=LU$

$$\mathrm{Mv}=\mathrm{d}, \mathrm{Mw}=e_1 \qquad (5)$$

を定義することで

$\mathrm{u}=\mathrm{v}-\xi \mathrm{v}_1\mathrm{w}_1$,$\xi=(\mathrm{w}_1+(\beta\gamma)^{-1})^{-1}$

となり,式(5)を解くことで方程式を解くことができる.

解 u を求めるうえで主な計算は式(5)になる.

式(5)は対角要素が 1 の 2 対角行列であるため

$$x_1=b_1,\ x_i=ax_{i-1}+b_i,\ \ i=2,3,\cdots,n \qquad (6)$$

一次線形漸化式(6)を解くことによって解を求めることができる.

一次線形漸化式は図 1.のアルゴリズムによって求めることができる.

$l=[\log_2 n]$
$for\ i=n+1,n+2,\cdots,l\ do\ b[i]=0,$
$for\ k=1,2,\cdots,l\ do\{$
$\quad for\ i=n-2^{k-1},\cdots,2,1\ do\{$
$\quad\quad b[i+2^{k-1}]=a*b[i]+b[i+2^{k-1}],$
$\quad\quad a=a*a.\}$
$for\ i=1,2,,\cdots,n\ \ do\ x[i]=b[i],$
end

図 1.一次線形漸化式の並列算出手法

5. 実験結果

GPU を用いた並列計算プラットフォーム CUDA を用いて 2500step の数値実験を行った結果を表 1 に示す.

表 1.提案アルゴリズム数値実験結果

行列次元数	1024	8192	65536
所有時間[ms]	263.16	333.97	418.02

6. これから

実装コードにはまだ最適化を行う余地があり,さらなる高速化のために検討及び改良を行う必要がある.

7. 参考文献

[1] I.Chiorean, Parallel Algorithms for solving Black-Scholes equation, Kragujevac J Math. 27(2005) 91-100.
[2] L.Li, et al., A Parallel Algorithm for Solving the Implicit Diffusion Difference Equation, *Intern. J. Computer Math*, 67(1998), 435-440.