

## 5.27 モンテカルロシミュレーション 分散減少法(5) 工事中

### 重点サンプリング法 Importance Sampling

この方法は、Truncate された分布や分布のある部分しか事実上必要とならないケース、それに非常に稀な部位を含む分布をシミュレーションする際に用いられる手法です。ある意味で、この手法はウエイト化されたサンプリングととも言えるものです。

下の例は、ヨーロピアンオプションの解を求める際に、コールならば  $S > K$  である部分のみのサンプリングをして解を求め、最後にそのサンプリングの起き得る確率をかけ合せたものです。同様にして、プットならば  $S < K$  である部分についてサンプリングして解を求め、最後にそのサンプリングの起き得る確率をかけ合せて最終的な解を求めています。

プログラム

```
new; cls;
S0=50;
K=50;
r=0.10;
sig=0.40;
T=5/12;
times=5000;
print "C=" Cimpsamp(S0,K,r,sig,T,times);
print "P=" Pimpsamp(S0,K,r,sig,T,times);
```

```
proc Cimpsamp(S0,K,r,sig,T,times);
    local e,S,payoffs,Epayoff,op,p;
    e=rndn(times,1);
    S=S0*exp((r-sig^2/2)*T+sig*e*sqrt(T));
    p=sumc(S.>K)/times;          /*      p=Prob(S>K)      */
    S=selif(S,(S.>K));           /*      sampling only if S>K      */
    payoffs=maxc(((S-K)~zeros(rows(S),1))');
    Epayoff=meanc(payoffs);
    op=exp(-r*T)*Epayoff;
    retp(p*op);                  /*      finally, multiplied by p      */
endp;
```

```
proc Pimpsamp(S0,K,r,sig,T,times);
    local e,S,payoffs,Epayoff,op,p;
```

```

e=rndn(times,1);
S=S0*exp((r-sig^2/2)*T+sig*e*sqrt(T));
p=sumc(S.<K)/times;          /*      p=Prob(S<K)      */
S=selif(S,(S.<K));           /*  sampling only if S<K  */
payoffs=maxc(((K-S)~zeros(rows(S),1))');
Epayoff=meanc(payoffs);
op=exp(-r*T)*Epayoff;
retp(p*op);                  /*  finally, multiplied by p  */
endp;
画面表示
C=      6.0407486
P=      4.1741524

```