Read Me

YE Fan

October 9, 2014

The included ***makefile*** is designed for ***Linux***.
In order to compile the code, run[1]

make all

In order to run all benchmarks, run[2]

make run

File list

The *BSERROR* directory contains all the files intended for PAC'14. The history of development is kept using *GIT*. Here's a brief description for each file.

BSERROR
.git: git repository
.gitignore: instruct git to ignore untracked files
makefile: makefile for the programs
run.sh: a script called by makefile, please not to run it directly
seq.c: serial version of the model
par_omp.c: 1st OpenMP version
par_omp2.c: 2nd OpenMP version
par_mpi.c: Hybrid MPI/OpenMP version, which is the final goal of this project

BSERROR/inc header files
misc.h: implementations of some auxiliary routines
onetimesimu.h: implementation of one-time Monte Carlo simulation taking the par_omp2.c approach, along with some relative routines

BSERROR/doc documents
readme: tex files for generating README.pdf
report: tex files for generating report.pdf

BSERROR/ParameterAnalysis various experimental results and figures

[1]Please note that it is required to have Intel compiler installed in the system.
[2]Also please note that the presence of Intel Xeon Phi coprocessor and a running MIC Platform Software Stack (MPSS) is indispensable to run *.mic

Mathematical description of the project

Equation 1 gives the discrete time hedging error for one time simulation. The goal is to simulate M times where M is sufficiently large so that the $Prob$ tends to be stable. The $Prob$ is defined as the number of times out of M when error is less than an accepted value ϵ.

$$\begin{aligned}
M_T^N &= e^{-rT} f(X_T) - (u(0,x) + \int_0^T \frac{\partial u}{\partial x}(\varphi(t), X_{\varphi_t})) d\widetilde{X}_t \\
&= \int_0^T \frac{\partial u}{\partial x}(t, X_t) d\widetilde{X}_t - \int_0^T \frac{\partial u}{\partial x}(\varphi(t), X_{\varphi(t)}) d\widetilde{X}_t \\
&= (X_T - K)^+ - \mathbb{E}[(X_T - K)^+] - \int_0^T \frac{\partial u}{\partial x}(\varphi(t), X_{\varphi(t)}) dX_t \\
&= (X_T - K)^+ - x_0 N(d_1(0)) + K N(d_2(0)) - \sum_{i=0}^{n-1} N(d_1(t_i))(X_{t_{i+1}} - X_{t_i}) \\
&= (X_T - K)^+ - \frac{x_0}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\log(\frac{x_0}{K}) + \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}}} e^{-\frac{v^2}{2}} dv + \frac{K}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\log(\frac{x_0}{K}) - \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}}} e^{-\frac{v^2}{2}} dv \\
&- \sum_{i=0}^{n-1} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\log(\frac{X_{t_i}}{K}) + \frac{1}{2}\sigma^2 (T - t_i)}{\sigma\sqrt{T - t_i}}} e^{-\frac{v^2}{2}} dv (X_{t_{i+1}} - X_{t_i})
\end{aligned} \tag{1}$$

The X_t is the price of the underlying asset for the option, which is assumed to evolve in time according to the stochastic equation

$$dX(t) = \mu X(t) dt + \sigma X(t) dB(t) \tag{2}$$

In this equation, μ is the drift of the asset, σ is the option volatility, and $B(t)$ is a standard Brownian motion.

The solution of this stochastic differential equation can be written as

$$X_{t_i} = X_{t_{i-1}} e^{(\mu - \sigma^2/2)\delta t + \sigma\sqrt{\delta t}\chi} \tag{3}$$

where χ is a normally distributed random variable with zero mean and unit standard deviation, and $\delta t = t_i - t_{i-1}$.

By our unpublished theory, the upper bound of N (discrete intervals of hedging) is given by Equation 4, which's the best estimation (the lowest upper bound) for the time being.

$$N_{max} = \log^3(1 - Prob) e^{\frac{1}{4}} \frac{1}{T^{\frac{1}{4}} \sqrt{\frac{\epsilon}{(\log \frac{X_0}{K} + 0.5\sigma^2 T)\sqrt{2\pi}}}} \cdot \left(-\frac{8e^3 X_0^2 \cdot 16 \cdot e^{\sigma^2}}{27\epsilon^2 \pi}\right) \tag{4}$$

There's one more thing to mention, we've implemented our own gaussian integral function using Simpson's rule. See equations 5 and 6.

$$\int_a^b f(x) dx \approx \frac{h}{3} \left[f(x_0) + 2 \sum_{j=1}^{n/2-1} f(x_{2j}) + 4 \sum_{j=1}^{n/2} f(x_{2j-1}) + f(x_n) \right] \tag{5}$$

$$f(x) = \int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt = 0.5 \times \sqrt{2\pi} + \int_0^x e^{-\frac{t^2}{2}} dt \tag{6}$$