COMPARISON OF VALUE-AT-RISK (VAR) USING DELTA-GAMMA APPROXIMATION WITH HIGHER ORDER APPROACH

TEO LI HUI

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF MATHEMATICS

NATIONAL UNIVERSITY OF SINGAPORE

2006

ACKNOWLEDGEMENT

Many individuals contributed to the success of this thesis. Everyone was particularly helpful in the progress of this report too. First of all, I would like to thank Assoc Prof Jin Xing, as my supervisor who guided me patiently and willingly to share his knowledge with me. A special thank is extended to him for his construction guidance.

Million of thanks are dedicated to my family who giving me fully support in producing this thesis. Thank you for their encouragement all the way in doing this thesis.

Last but not least, I would like to thank all of my friends. A special thank to Lim and Wu who give me corporations in completing this study. Also, I would like to express my gratitude to all the others that had indirectly helped me in producing this thesis.

Once again, thank you so much.

TABLE OF CONTENTS

SUMMARY

Value-at-Risk (VaR) has emerged as a popular method to measure financial market risk that was developed in response to the financial disasters in the early 1990s. There had been frequent debates about the accuracy of various methodologies.

In this dissertation, we propose a new methodology which include third and forth moment into existing Delta-Gamma approximation in calculating VaR for non-linear portfolios.

We also consider the application of this new method to standard Monte Carlo simulation and Quasi Monte Carlo simulation. A computer implementation of Value-at-Risk simulation was carried out to verify the faster convergence rate of this approach.

We will provide numerical examples to demonstrate the faster convergence rate and do the comparison with other approaches.

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Introduction to Value-at-Risk (VaR)

Financial corporate are always faced with various kind of risk. Generally, risk itself can be defined as the degree of uncertainty about the future net returns. While there are many sources of financial risk, the most prominent is the market risk which estimates the uncertainty of future earnings, due to the changes in market. Hence value-at-risk (VaR) has become an important tool in measuring the portfolio risk.

In most common way, VaR can be defined as the maximum potential loss that will occur over a given time horizon (under normal market condition) with a certain confidence level α. In other words, it is a number that indicates how much an institution can lose with probability α over a given time horizon. The reason VaR become so popular nowadays is that it successfully reduces the market risk associated with any portfolio to just a single number, which is the loss associated with a given probability.

From the view point of statistics, VaR estimation is the estimation of a quantile of the distribution of the returns. For instance, a daily VaR of $30 million at 95%

confidence level suggest that a 5% chance for a loss greater than $30 million to occur during any single day.

1.2 Background

As VaR become a powerful tool to measure risk, there are various methodologies to calculate VaR. The common approaches of VaR calculation include historical simulation, variance-covariance approach, Monte Carlo simulation and Delta-Gamma approximation.

1.2.1 Historical simulation

The historical simulation involved using past data to predict future. First of all, we have to identify the market variables that will affect the portfolio. Then, the data will be collected on the movements in these market variables over a certain time period. This provides us the alternative scenarios for what can happen between today and tomorrow. For each scenario, we calculate the changes in the dollar value of portfolio between today and tomorrow. This defines a probability distribution for changes in the value of portfolio. For instance, VaR for a portfolio using 1-day time horizon with 99% confidence level for 500 days data is nothing but an estimation of the loss when we are at the fifth-worst daily change.

Basically, historical simulation is extremely different from other type of simulation in that estimation of a covariance matrix is avoided. Therefore, this approach has simplified the computations especially for the cases of complicated portfolio.

The core of this approach is the time series of the aggregate portfolio return. More importantly, this approach can account for fat tails and is not prone to the accuracy of the model due to being independent of model risk. As this method is very powerful and intuitive, it is then become the most widely used methods to compute VaR.

1.2.2 Variance-covariance Approach

Variance-covariance approach which is known as delta-normal model was firstly proposed by J.P.Morgan Chase. Over the time interval, the portfolio return can be written as

$$R_{p,t+1} = \sum_{i=1}^{N} w_{i,t} R_{i,t+1} \quad ,$$

where the weights $w_{i,t}$ are indexed by time to recognize the dynamic nature of trading portfolios. Under the variance-covariance framework, we assume that all assets returns are normally distributed, which means that the return of the portfolio, being a linear combination of normal variables, is also normally distributed. Hence, the portfolios variance can be given by

$$V(R_{p,t+1}) = w_t^{'} \sum w_t \quad .$$

In this situation, risk is given by a combination of linear relationship of many risk factors which are assumed to be normally distributed and by the forecast of covariance matrix $\sum$. Generally, variance-covariance approach can accommodate a large number of assets and is easily implementable. As we made the assumption of normal distribution, portfolios of normal variables are themselves normally distributed. Consequently, since the portfolios are linear combinations of assets, the variance-covariance approach turns out to be linear.

Formally, the potential loss in value V is computed as $V = \beta_0 \times \Delta S$ which in other words it is the product of β_0 and ΔS whereas β_0 is the portfolio sensitivity to changes in prices, evaluated at current position V_0 and ΔS is the potential change in prices.

Obviously, the normality assumption allows us to estimate the portfolio β simply as the average of individual betas.

This model is ideally suited to large portfolios which are exposed to many risk factors as this method only requires computing the portfolio value once. As a result, the utilization of time to compute VaR can be reduced.

1.2.3 Monte Carlo simulation

Monte Carlo simulation is another popular method to calculate VaR. It is a very natural methodology to deal with a portfolio which is nonlinear. We will cover the procedure of this well-known method in the followings.

Firstly, we assume that the portfolio consists of d risk factors and $S(t) = (S_1(t), ..., S_d(t))'$ denotes their value at time t. Assume that $S_1(t), ..., S_d(t)$ follows Geometric Brownian Motion, their discrete price path can be described as

$$S_i(t) = S_i(0)\exp\left[(\mu_i - \frac{\sigma_i^2}{2})t + \sigma_i \sqrt{t}\varepsilon_i \right], i = 1, ..., d \ ,$$

where μ_i is the drift, σ_i is the volatility, t is the time horizon and ε_i is a standard normal random variable. In matrix forms,

$$S(t) = S(0)\exp\left[(\mu - \frac{\sigma^2}{2})t + \sigma\sqrt{t} .* P'\varepsilon \right] ,$$

where

$$S(0) = (S_1(0), ..., S_d(0))' \ ,$$
$$\mu = (\mu_1, ..., \mu_d)' \ ,$$
$$\sigma = (\sigma_1, ..., \sigma_d)' \ ,$$
$$\varepsilon = (\varepsilon_1, ..., \varepsilon_d)' \ ,$$

$P'P = \sum$ is the covariance matrix with variance unity.

Then the portfolio value $v_k(t)$ for each simulation can be obtained. The next step is assume that the initial time is 0 and then calculates the portfolio gain $V_k(t)$ for each simulation using the followings:

$$V_k(t) = v_k(t) - v(0) \ ,$$

where $v(0)$ is the portfolio value at the initial time.

The procedure is then continued by sorting $V_k(t)$ in ascending order. VaR is the αth-quantile of a portfolio's gain distribution function.

To get a better estimation of VaR, we have to repeat the above procedure for m times. VaR is then given by a pool of estimation

$$VaR = \frac{1}{m}\sum_{j=1}^{m} VaR(j) \ .$$

Monte Carlo simulation is by far the most powerful method to compute value-at-risk. It can be used to evaluate a wide range of risks, including nonlinear price risk, volatility risk and even model risk.

However, this method suffers from two drawbacks. First, it requires a large number of evaluations. For large or complex portfolios this can be extremely time-demanding. Second and more importantly, traditional Monte Carlo, utilizing independent sampling of pseudo-random numbers, undesirably tends to form clusters in the sample space which leads to gap where sample space may not be explored at all, so the accuracy is adversely affected by clustering and gaping of the sample.

Overall, this method is probably the most comprehensive approach to measuring market risk if the model is done correctly.

1.2.4 Delta-Gamma Approximation

Delta-Gamma approximation is one of the most popular tools in measuring VaR for a non-linear portfolio. The coefficients used in this approach are the 1st and 2nd order sensitivities of the present values with respect to the changes in the underlying risk factors.

First of all, assume that we have d risk factors and that $S(t) = (S_1(t), \ldots, S_d(t))'$ denotes the value of these factors at time t. Defining $\Delta S = S(t + \Delta t) - S(t)$ to be the change in the risk factors during the interval $[t, t + \Delta t]$.

The Delta-Gamma approximation is then given by

$$\Delta V \approx \Theta \Delta t + \delta ' \Delta S + \frac{1}{2} \Delta S ' \Gamma \Delta S$$

$\Theta = \frac{\partial V}{\partial t}, \delta = \frac{\partial V_i}{\partial S_i(t)}, \Gamma_{ij} = \frac{\partial^2 V}{\partial S_i \partial S_j}$ (All partial derivatives being evaluated at *S(t)*)

Hence, for a given probability α, the VaR denoted by ξ_α is then

$$P\{-\Delta V \geq \xi_\alpha\} = \alpha \ .$$

This approach is much less time-consuming compared to a full simulation as it avoids repricing the whole portfolio on each simulation trial. It is also very easy to implement. However, it gives a poor convergence rate for portfolio which contains highly non-linear responses to risk for example, out-of-money option.

As a result, the higher moments of risk factors should be included in VaR calculation. This sparks the idea of this study.

1.3 The Scope of the Study

In this study, we will introduce the third and fourth moments to Delta-Gamma approximation to obtain a more accurate result in VaR calculation and show that why this two moments is included and the fifth and sixth moments are neglected. Then we will implement this new model to existing Monte-Carlo simulation and Quasi Monte Carlo simulation. Lastly, comparison between the new model and other methodologies will be carried out.

1.4 Outline

The rest of the thesis is organized as follows. In the next section, we will introduce the new model, Delta-Gamma-Skewness-Kurtosis approach in calculating Value-at-Risk for non-linear portfolio. Numerical examples are discussed in Section 3 to illustrate and compare the performance of various approaches. Section 4 concludes the paper. Appendices A and B include the proof of the theoretical results.

CHAPTER 2 DELTA-GAMMA-SKEWNESS-KURTOSIS APPROXIMATION

2.1 Literature Review

Many researchers have looked at the method of producing an accurate value-at-risk. We now review some of the recent paper.

Jamshidian and Zhu (1997) presented a factor-based scenario simulation in which they discretize the multivariate distribution of market variables into a limited number of scenarios.

However, **Abken (2000)** found that scenario simulation only converges slowly to the correct limiting values and convexity of the derivative values significantly weakens the performance of scenario simulation compare to standard Monte Carlo simulation.

At the same time, **Michael and Matthew (2000**) argued that factor-based scenario simulation failed to estimate VaR for some fixed-income portfolios. They proposed generating risk factors with a statistical technique called partial least squares instead of generating them with principal components analysis. They have suggested using “Grid Monte Carlo” method to compute VaR.

Meanwhile some of the researchers found that variance reduction technique was successfully increased the accuracy of standard Monte Carlo. In both **Hsu and Nelson (1990)** and **Hesterberg and Nelson (1998)** paper, control variates are used to reduce variance in simulation-based estimation for quantile which is equivalent to the estimation of VaR in a financial setting.

Avramidis and Wilson (1998) applied the correlation-induction techniques and Latin hypercube sampling to improve quantile approximation.

Glasserman et al. (2000) used stratified sampling and importance sampling in delta-gamma approximation. They combined these two methods to obtain further variance reduction. They extended their work by combining the speed of the delta-gamma approach and the accuracy of Monte Carlo simulation. By using delta-gamma approximation to guide the sampling of scenarios and through the combination of importance sampling and stratified sampling, they successfully reduced the number of scenarios needed in a simulation to achieve a specified precision.

Also, **Owen and Zhou (1998)**, **Avramidis and Wilson (1996)** are good references for the method of using conditional expectation to reduce variance. **Jin Xing et al. (2004)** improved the method by focusing on Quasi Monte Carlo which is as not sophisticated as Monte Carlo simulation.

Britten-Jones et al. (1999) proposed an alternative approach where the changes in value of an assets is approximated as a linear-quadratic function. Compared to delta-only approach, this gives a better estimation of the true distribution. Also, it is less time-consuming than a full valuation. This approach is also discussed in **Wilson (1994), Fallon (1996), Rouvinez (1997) and Jahel, Perrauddin and Sellin (1997).**

Using Imhof's numerical technique, **Rouvinez** invert the characteristic function of the quadratic approximation and so recover the exact distribution.

Jahel et al. used the characteristic function to compute the moment of approximation and fit the moments with a parametric distribution.

Fallon uses an approximation to the distribution derived from the moments.

Wilson (1994) used a linear-quadratic approach but the statistic he derived, "capital-at-risk" [CAR] differs significantly from the standard definition of VaR.

2.2 Delta-Gamma-Skewness-Kurtosis Model (DGSK)

As mentioned before, delta-gamma approximation gives a poor approximation for a portfolio which consists of highly non-linear responses. To overcome this problem, we introduce third and forth moments into the existing delta-gamma approximation and it will be proved that with these added moments, a more accurate result can be obtained. Here and after, we named this new model as Delta-Gamma-Skewness-Kurtosis model or in short as DGSK model.

To set up our model, we begin with the Taylor series approximation. The Taylor series relates the value of a differentiable function at any point to its first and higher order derivatives at a reference point. Mathematically, we can write it as

$$f_k = f_0 + (kT) f_0^{(1)} + \frac{(kT)^2}{2!} f_0^{(2)} + \frac{(kT)^3}{3!} f_0^{(3)} + \ldots + \frac{(kT)^n}{n!} f_0^{(n)} + O(T^{n+1}) \quad , \qquad \text{-(2.1)}$$

where f_k denotes the value of $f(t)$ at $t = kT$, $k = 0, \pm 1, \pm 2, \ldots, T$ is the sampling period, $f_0^{(k)}$ denotes the kth derivative of f at $t = 0$ and $O(T^{n+1})$ coming from the truncation of the series after *n+1* terms. Here the central difference method is used to approximate the derivatives.

By using central difference approximation, equation (2.1) becomes

$$f_k = f_0 + (kT) f_0^{(1)} + \frac{(kT)^2}{2!} f_0^{(2)} + \frac{(kT)^3}{3!} f_0^{(3)} + \ldots + \frac{(kT)^{2n}}{2n!} f_0^{(2n)} + O(T^{2n+1}) \ . \qquad \text{-(2.2)}$$

In DGSK model, we have n=2 as the first four moments are included in pricing the portfolio. Hence we have

$$f_k = f_0 + (kT) f_0^{(1)} + \frac{(kT)^2}{2!} f_0^{(2)} + \frac{(kT)^3}{3!} f_0^{(3)} + \frac{(kT)^4}{4!} f_0^{(4)} + O(T^5) \ . \qquad \text{-(2.3)}$$

Due to the derivative is obtained by solving a set of 2n equations, the last term of equation (2.1) has become $O(T^{2n+1})$.

Using these notations, a set of Taylor series can be written in matrix form as the followings:

$$F_c = A_c D_c + O(T^{2n+1}) \ ,$$

where F_c and D_c are the vectors of length 2n. A_c is a 2n *x* 2n square matrix and they are defined as

$$F_c = \begin{bmatrix} f_1 - f_0 \\ f_{-1} - f_0 \\ f_2 - f_0 \\ f_{-2} - f_0 \\ \vdots \\ f_4 - f_0 \\ f_{-4} - f_0 \end{bmatrix}, \ D_c = \begin{bmatrix} f_0^{(1)} \\ f_0^{(2)} \\ \vdots \\ f_0^{(4)} \end{bmatrix},$$

$$A_c = \begin{bmatrix} T & \frac{T^2}{2!} & \frac{T^3}{3!} & \frac{T^4}{4!} \\ -T & \frac{(-T)^2}{2!} & \frac{(-T)^3}{3!} & \frac{(-T)^4}{4!} \\ 2T & \frac{(2T)^2}{2!} & \frac{(2T)^3}{3!} & \frac{(2T)^4}{4!} \\ -2T & \frac{(-2T)^2}{2!} & \frac{(-2T)^3}{3!} & \frac{(-2T)^4}{4!} \end{bmatrix}$$

The rest of VaR calculation is exactly the same as in Delta-Gamma approach. We will see in details later.

2.3 Methodology

This study consists of a few steps as follows:

a) Understand the problem of existing Delta-Gamma approximation in calculating VaR.

b) Seek the closed-form solution for European call option based on Heston (1993).

c) Obtain the closed-form solution for the finite difference approximations of first and higher order derivatives based on Taylor series.

d) Compare the result for these two methods.

e) Make conclusions and suggestions.

2.4 VaR Simulation

In this section, we focus on the VaR simulation. First of all, let $v(t)$ be the value of a portfolio at time t, for instance $v(t) = v(s(t), t)$. Assume that the initial time is 0, the portfolio changes over time t is then given by

$$\Delta v(t) = v(s(t), t) - v(s(0), 0) \ .$$

For a given probability α, the VaR denoted by ξ_α is then defined as

$$P\{v(s(0), 0) - v(s(t), t) \geq \xi_\alpha\} = \alpha \ .$$

Also, we can write it as

$$P\{v(s(t), t) - v(s(0), 0) \leq -\xi_\alpha\} = \alpha \ .$$

The confidence level α is usually close to zero and typically set to 0.01 or 0.05. Meanwhile, the holding period t is in between 1 day or a few weeks. These two variables are always depending on the needs of users.

Now we introduce the algorithm of this research. Firstly, we obtained the closed-form solution for European call option with volatilities based on Heston (1993) as stated in methodology. The core steps are shown as follows:

Assume that K and T is the strike price and maturity date for a European call option respectively, $v(t)$ is the variance, the option satisfies the following partial differential equation (PDE):

$$\frac{1}{2}vS^2\frac{\partial^2 U}{\partial S^2}+\rho\sigma vS\frac{\partial^2 U}{\partial S\partial v}+\frac{1}{2}\sigma^2 v\frac{\partial^2 U}{\partial v^2}+rS\frac{\partial U}{\partial S}+\left\{\kappa\left[\theta-v(t)\right]-\lambda(S,v,t)\right\}\frac{\partial U}{\partial v}-rU+\frac{\partial U}{\partial t}=0\ .$$

The term $\lambda(S,v,t)$ represents the price of volatility risk, and must be independent of the assets.

Subject to

$$\begin{aligned}
&U(S,v,t)=\max(0,S-K),\\
&U(0,v,t)=0,\\
&\frac{\partial U}{\partial S}(\infty,v,t)=1,\\
&rS\frac{\partial U}{\partial S}(S,0,t)+\kappa\theta\frac{\partial U}{\partial v}(S,0,t)-rU(S,0,t)+U(S,0,t)=0,\\
&U(S,\infty,t)=S\ .
\end{aligned}$$

By analogy with the Black-Scholes formula, a guessed solution of the form is shown.

$$C(S,v,t)=SP_1-KP(t,T)P_2\ ,$$

where the first term is the present value of the spot asset upon optimal exercise and the second term is the present value of the strike price payment. Both of these terms must satisfy the above PDE. By using the change of variables, we can get the characteristic function and its solution. Then we can invert the characteristic function to get the desired probabilities. By combing all the steps above we can get the solution for European call option. To see in details please refer to Heston (1993).

This method is very time-consuming especially when the number of samples is large. It is not practical for a company to spend such a long time to calculate VaR. However, we used the results from this method as the true value to compare with the results using Delta-Gamma approximation and Delta-Gamma-Skewness-Kurtosis model. The numerical examples will be shown in next chapter.

Besides that, I have applied the Delta-Gamma-Skewness-Kurtosis approach to Monte Carlo simulation and Quasi Monte Carlo simulation. We will not discuss much about the VaR calculation using Monte Carlo simulation and Quasi Monte Carlo simulation but will present some of the numerical examples.

We could now re-establish Glasserman (2003)'s result on the convergence rate and optimal holding period to our Quasi Monte Carlo simulation for VaR.

Theorem 1(Convergence Rate)

Assume the followings hold:

(1) $x_i^j - x_0^j$ independent but not i.i.d;

(2) $y_j^{(k)}$ i.i.d, $j = 1, \ldots, m$; -(2.4)

(3) $E(x_i^j - x_0^j)^2 = T\sigma_i^2 + o(T), j = 1, 2, \ldots, m$.

Then, the convergence rate is given by

a) $o(m^{-\frac{2n+1-k}{4n+1}}), k$ odd ;

b) $o(m^{-\frac{2n+2-k}{4n+3}}), k$ even .

Proof See Appendix A

Theorem 2(Optimal Δt)

Assume the followings hold:

(1) $x_i^j - x_0^j$ independent and i.i.d;

(2) $y_j^{(1)}$ i.i.d, $j = 1,...,m$; -(2.5)

(3) $E(x_i^j - x_0^j)^2 = T\sigma_i^2 + o(T), j = 1,2,...,m$.

Then, the optimal value of Δt * is given by

$$\Delta t^* = \left[\frac{1800\sigma^2}{(\sum_{\substack{i=-2 \\ i\neq 0}}^{2} \frac{\Delta_i}{\Delta} i^5)^2 (f_0^{(5)})^2} \right]^{\frac{1}{9}}$$

Proof See Appendix B

CHAPTER 3 NUMERICAL RESULTS

In this chapter we will present some of the numerical examples that we have been carried out. As mentioned before, we obtained the target VaR based on Heston (1993). Then we performed the same experiments using Delta-Gamma approach and Delta-Gamma-Skewness-Kurtosis approximation. After that, we compared the results from these three methods and make some analysis.

Besides that, we applied the Delta-Gamma-Skewness-Kurtosis model to standard Monte Carlo simulation and proved that there is a fluctuation in the results. Hence we have improved it by using Quasi Monte Carlo simulation with Sobol sequence. Also we will display why the fifth and sixth moments are not considered in pricing the option.

For all experiments, the confidence level of VaR is set at 99%, corresponding to $\alpha = 0.01$. Additionally, we assume there are 250 trading days in a year and instantaneous short rate of 5%. Options will mature in one year and holding period Δt is one day or $\frac{1}{250}$ a years. All the experiments have been done using different initial stock prices, $s0 = 80, 90, 100, 110, 120$ and number of simulation path, $n = 50000$ for target VaR and $n = 1000, 4000, 16000$ for experiments.

Table 1: Comparison DG and DGSK

s0=80 TrueVaR=0.2177(std=0.0007)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	0.1554	0.0002	0.003881	0.1642	0.2176	0.0036	0.0000	0.3305	299.2544
4000	0.1554	0.0001	0.003881	0.1646	0.2158	0.0020	7.61E-06	0.3220	510.0263
16000	0.1554	0.0000	0.003881	0.1717	0.2161	0.0008	0.0000	0.3315	1212.9000

s0=90 TrueVaR=1.1340(std=0.0054)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	1.0199	0.0754	0.0133	0.1803	1.1359	0.0210	0.0004	0.3300	29.8148
4000	1.0215	0.0077	0.0127	0.1662	1.1395	0.0146	0.0002	0.3195	52.2392
16000	1.0207	0.0034	0.0128	0.1828	1.1365	0.0121	0.0002	0.3345	84.1638

s0=100 TrueVaR=3.1459(std=0.0167)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	3.1606	0.1136	0.0131	0.1717	3.1463	0.1079	0.0116	0.3245	1.1270
4000	3.1547	0.0696	0.0049	0.1652	3.1316	0.0798	0.0066	0.3245	0.7488
16000	3.1403	0.0317	0.0010	0.1798	3.1409	0.0341	0.0012	0.3365	0.8724

s0=110 TrueVaR=5.3575(std=0.0327)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	5.4069	0.2025	0.0434	0.1707	5.4605	0.2572	0.0768	0.3260	0.5660
4000	5.5141	0.1322	0.0420	0.1657	5.3485	0.1393	0.0195	0.3195	2.1555
16000	5.4877	0.0682	0.0216	0.1763	5.3681	0.0367	0.0015	0.3300	14.8044

s0=120 TrueVaR=6.7144(std=0.0437)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	6.8631	0.2901	0.1063	0.2248	6.7913	0.2764	0.0823	0.4376	1.2911
4000	6.7816	0.1773	0.0360	0.2278	6.7466	0.1443	0.0219	0.4386	1.6447
16000	6.7922	0.0843	0.0132	0.2373	6.7200	0.0861	0.0074	0.4506	1.7676

Table 1 shows that the comparison between Delta-Gamma approximation and Delta-Gamma-Skewness-Kurtosis approach. The column named VaR indicates the value-at-risk of the portfolio; Std represents the standard deviation of VaR. For your information, we have repeated these experiments 20 times and the VaR here was the mean of 20 experiments. Meanwhile, M is the measure of method X and it is obtained by using the following equation:

$$measure_X = (mean_X - mean_{truevalue})^2 + std_X^2 \quad ,$$

where $mean_{truevalue}$ is obtained based on Heston(1993)

and

$$ratio = \frac{measure_{DG}}{measure_{DGSK}} \quad .$$

Here, the column Cpu refers to the time used to calculate VaR. Correspondingly, it can refer to the speed of my method. All the experiments have been done by using Intel Pentium M processor 715 with 1.5Ghz.

From table 1, we found that in most of the cases Delta-Gamma-Skewness-Kurtosis approach gave us more accurate results than Delta-Gamma approximation. Obviously, by adding the third and forth moments into the existing Delta-Gamma approach, the weaknesses of Delta-Gamma approximation has been improved. Hence, the problem of calculating the VaR of non-linear portfolio is solved and it is clear that Delta-Gamma-Skewness-Kurtosis model has successfully overcome the problem of poor convergence rate of existing Delta-Gamma approach, for example when the initial stock price is 80.

Table 2: Comparison TRUE VALUE and DGSK

Here $\kappa = 2, \theta = 0.01, v = 0.01, \rho = 0, \sigma = 0.1, T = 0.5yr, r = 0, K = 100$.

s0=80 TrueVaR=0.2177(0.0007)

n	Heston				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	perfomance
1000	0.2177	0.0036	0.0000	14.8504	0.2176	0.0036	0.0000	0.3305	44.8985
4000	0.2173	0.0014	0.0000	58.6674	0.2158	0.0020	0.0000	0.3220	50.7566
16000	0.2180	0.0010	0.0000	237.3753	0.2161	0.0008	0.0000	0.3315	243.9094

s0=90 TrueVaR=1.1340(0.0054)

n	Heston				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	perfomance
1000	1.1349	0.0371	0.0014	10.8156	1.1359	0.0210	0.0004	0.3300	101.5221
4000	1.1320	0.0171	0.0003	42.7895	1.1395	0.0146	0.0002	0.3195	163.0875
16000	1.1308	0.0068	0.0001	173.2366	1.1365	0.0121	0.0002	0.3345	191.6077

s0=100 TrueVaR=3.1459(0.0167)

n	Heston				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	perfomance
1000	3.1248	0.1154	0.0138	10.9117	3.1463	0.1079	0.0116	0.3245	39.7486
4000	3.1523	0.0683	0.0047	42.6453	3.1316	0.0798	0.0066	0.3245	94.0940
16000	3.1457	0.0311	0.0010	172.6127	3.1409	0.0341	0.0012	0.3365	417.7144

s0=110 TrueVaR=5.3575(0.0327)

n	Heston				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	perfomance
1000	5.2868	0.1957	0.0433	10.9447	5.4605	0.2572	0.0768	0.3260	18.9367
4000	5.3626	0.1029	0.0106	42.7334	5.3485	0.1393	0.0195	0.3195	72.8587
16000	5.3740	0.0638	0.0043	172.5611	5.3681	0.0367	0.0015	0.3300	1556.1738

s0=120 TrueVaR=6.7144(0.0437)

n	Heston				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	perfomance
1000	6.7543	0.2435	0.0609	15.5023	6.7913	0.2764	0.0823	0.4376	26.2040
4000	6.7470	0.1760	0.0320	60.8495	6.7466	0.1443	0.0219	0.4386	203.3421
16000	6.6979	0.0785	0.0064	243.0931	6.7200	0.0861	0.0074	0.4506	466.2906

The main purpose of table 2 is to compare the accuracy of Delta-Gamma-Skewness-Kurtosis model and the true value. The column named performance is calculated as the followings:

$$performance = \frac{measure_H \times Cpu_H}{measure_{DGSK} \times Cpu_{DGSK}} \ .$$

As we can see from table 2, Heston (1993) approach is still applicable when the number of sample size is small. The problem appears when the number of sample size becomes large. It is clear that when the number of sample size is increasing, more time is required to calculate VaR. However, Delta-Gamma-Skewness-Kurtosis approach does not encounter with this kind of problem. The speed of this new approach is much faster than Heston (1993). For the performance column, we can notice that the performance of the new approach is hundred times better than Heston (1993) as it is pretty less time-consuming.

True value=Black scholes

Table 3:Comparison DG and DGSK(Monte Carlo simulation)

s0=80 true value=0.1386(0.0003)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	0.0885	0.0000	0.0025	21.3377	0.2021	0.0147	0.0042	41.2989	0.5908
4000	0.0885	0.0000	0.0025	21.2621	0.1927	0.0054	0.0030	41.7746	0.8491
16000	0.0885	0.0001	0.0025	21.1534	0.1960	0.0032	0.0033	41.4366	0.7595

s0=90 true value=1.3603(0.0056)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	1.1970	0.0054	0.0267	21.4909	1.2359	0.0081	0.0155	41.6148	1.7178
4000	1.1941	0.0035	0.0276	21.2591	1.2366	0.0048	0.0153	41.5903	1.8033
16000	1.1955	0.0020	0.0272	21.2275	1.2373	0.0027	0.0151	42.3659	1.7946

s0=100 true value=4.1777(0.0271)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	4.2551	0.1536	0.0296	21.3892	4.0813	0.1074	0.0208	41.3450	1.4204
4000	4.2438	0.0760	0.0101	21.2716	4.0333	0.0691	0.0256	41.7320	0.3959
16000	4.2938	0.0494	0.0159	21.2185	4.0476	0.0336	0.0181	41.9318	0.8817

s0=110 true value=6.4172(0.0447)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	6.5851	0.3312	0.1379	21.3152	6.4529	0.3819	0.1471	41.8487	0.9372
4000	6.5573	0.1795	0.0518	21.179	6.5229	0.2043	0.0529	42.6383	0.9799
16000	6.5366	0.0771	0.0202	21.3913	6.5108	0.0841	0.0158	42.2473	1.2758

s0=120 true value=7.1985(0.0496)

n	Delta-Gamma				Delta-Gamma-Skewness-Kurtosis				
	VaR	Std	M	Cpu	VaR	Std	M	Cpu	Ratio
1000	7.355	0.3432	0.1423	21.3347	7.2475	0.3839	0.1498	41.689	0.9499
4000	7.1967	0.1958	0.0383	21.1769	7.0654	0.1732	0.0477	41.4591	0.8036
16000	7.2292	0.0965	0.0103	21.4759	7.0847	0.0815	0.0196	41.392	0.5234

Besides that, we applied the Delta-Gamma-Skewness-Kurtosis approach to standard Monte Carlo simulation. Here, the true value of VaR is obtained by using original Black-Scholes formula. Call option price of the portfolio is calculated based on the following equation:

$$c(s) = sN(d_1) - e^{-rt} KN(d_2)$$

with

$$d_1 = \frac{\log(\frac{s}{K}) + (r + \frac{\sigma^2}{2} t)}{\sigma\sqrt{t}} \ ,$$

$$d_2 = d_1 - \sigma\sqrt{t} \ ,$$

where K is the strike price, r is the interest rate, σ is volatility, t is the maturity date, N is cumulative normal distribution function.

As we can see from table 3, Delta-Gamma-Skewness-Kurtosis approach in Monte Carlo simulation only shows slightly better results than the Delta-Gamma approach. However, in certain case as the initial stock price is 120, Delta-Gamma approach converges to true value faster than the Delta-Gamma-Skewness-Kurtosis approach.

Hence, we further the experiments by using Delta-Gamma-Skewness-Kurtosis model and Sobol sequence. As before, we perform experiments on Delta-Gamma approximation and Delta-Gamma-Skewness-Kurtosis approach and compare the results from these two methods. We also carried out an additional experiment by adding two more moments into the Delta-Gamma-Skewness-Kurtosis approach; correspondingly six moments are included in pricing the option.

The last column in table 4 which is DGSK56 gives us the value-at-risk of the portfolio by adding 5^{th} and 6^{th} moments in pricing the option. It is proved that by using four moments we can successfully obtain the result which converge to true value while added 5^{th} and 6^{th} moments are redundant.

Table 4: Comparison original Black-Scholes and Quasi Monte Carlo simulation n=50000, confidence level=0.01

s0	Original bls	DG(QMC)	DGSK(QMC)	DGSK56(QMC)
80	0.1387	0.0885	0.1405	0.1397
90	1.3602	1.1957	1.3724	1.3606
100	4.1744	4.2851	4.1588	4.1703
110	6.4091	6.5548	6.4208	6.422
120	7.2017	7.2255	7.1895	7.1871

Table 5a: Initial stock price 80, n=16000

true value=0.1386(0.0003)

	Δt =1		Δt =8.5			
	quasi monte carlo		quasi monte carlo		monte carlo	
s0=80	DG	DGSK	DG	DGSK	DG	DGSK
1	0.0890	0.1584	0.0875	0.1422	0.0877	0.1402
2	0.0880	0.1190	0.0875	0.1419	0.0872	0.1376
3	0.0892	0.1306	0.0875	0.1417	0.0861	0.1395
4	0.0881	0.1174	0.0875	0.1422	0.0856	0.1371
5	0.0876	0.1940	0.0875	0.1424	0.0867	0.1438
6	0.0893	0.1769	0.0875	0.1421	0.0853	0.1405
7	0.0880	0.1083	0.0875	0.1419	0.0884	0.1441
8	0.0879	0.1247	0.0875	0.1412	0.0862	0.1349
9	0.0880	0.1329	0.0875	0.1423	0.088	0.1454
10	0.0881	0.1645	0.0875	0.1409	0.0874	0.1419
11	0.0883	0.1198	0.0875	0.1413	0.0879	0.1417
12	0.0884	0.1108	0.0875	0.1416	0.0864	0.137
13	0.0892	0.2370	0.0875	0.1406	0.0865	0.1411
14	0.0886	0.1342	0.0875	0.1403	0.0863	0.1358
15	0.0885	0.1446	0.0875	0.1408	0.0868	0.1397
16	0.0885	0.1325	0.0875	0.1428	0.0851	0.1372
17	0.0885	0.2173	0.0875	0.1419	0.0885	0.1437
18	0.0888	0.1105	0.0875	0.1408	0.0877	0.1417
19	0.0877	0.1495	0.0875	0.1416	0.0908	0.1506
20	0.0881	0.1094	0.0875	0.1415	0.088	0.1422
mean	0.0884	0.1446	0.0875	0.1416	0.0871	0.1408
std	0.0005	0.0368	0.0000	0.0007	0.0013	0.0037
MSE	**0.0025**	**0.0013926**	**0.00261**	**0.00001**	**0.00265**	**0.00002**

Remark: MSE refers to Mean Square Error.

Table 5b: Initial stock price 90, n=16000

true value=1.3603(0.0056)

	Δt =1		Δt =8.5			
	quasi monte carlo		quasi monte carlo		monte carlo	
s0=90	DG	DGSK	DG	DGSK	DG	DGSK
1	1.1958	1.3877	1.2104	1.3704	1.2105	1.3824
2	1.1945	1.4085	1.2127	1.3725	1.2143	1.3695
3	1.1918	1.3381	1.2115	1.3690	1.2119	1.3871
4	1.1965	1.3581	1.2089	1.3668	1.1989	1.3516
5	1.1935	1.2759	1.2100	1.3528	1.2067	1.3718
6	1.1912	1.3091	1.2064	1.3524	1.2149	1.3779
7	1.1964	1.3788	1.2123	1.3968	1.1931	1.3477
8	1.1939	1.3283	1.2103	1.3705	1.2053	1.3801
9	1.1951	1.4013	1.2134	1.3712	1.2118	1.3732
10	1.1965	1.3909	1.2098	1.3654	1.2083	1.367
11	1.1957	1.2873	1.2112	1.3734	1.2048	1.3706
12	1.1966	1.3145	1.2116	1.3661	1.2173	1.3728
13	1.1964	1.4211	1.2133	1.3785	1.2193	1.3733
14	1.1976	1.4366	1.2100	1.3604	1.2048	1.3588
15	1.1976	1.3673	1.2104	1.3752	1.198	1.3533
16	1.1957	1.3881	1.2114	1.3688	1.2212	1.3842
17	1.1941	1.3433	1.2111	1.3701	1.1957	1.338
18	1.1952	1.3873	1.2127	1.3701	1.2086	1.3655
19	1.1963	1.3624	1.2145	1.3713	1.2016	1.3561
20	1.2001	1.3563	1.2145	1.3751	1.1998	1.3538
mean	1.1955	1.3620	1.2113	1.3698	1.2073	1.3667
std	0.0020	0.0436	0.0019	0.0092	0.0079	0.0133
MSE	**0.0272**	**0.00190012**	**0.02220**	**0.00018**	**0.02346**	**0.00022**

Table 5c: Initial stock price 100, n=16000

true value=4.1777(0.0271)

	Δt =1		Δt =8.5			
	quasi monte carlo		quasi monte carlo		monte carlo	
s0=100	DG	DGSK	DG	DGSK	DG	DGSK
1	4.2907	4.1929	4.2681	4.1601	4.2355	4.1262
2	4.2796	4.1878	4.2661	4.1583	4.2693	4.1559
3	4.2963	4.1803	4.2828	4.1735	4.2188	4.117
4	4.2928	4.1565	4.2780	4.1691	4.2206	4.1089
5	4.2907	4.0354	4.2775	4.1686	4.2829	4.1752
6	4.2203	4.0906	4.2031	4.1008	4.2777	4.1754
7	4.2793	4.2298	4.2562	4.1492	4.2129	4.1072
8	4.2266	4.0152	4.2096	4.1068	4.2573	4.1455
9	4.3679	4.1624	4.3455	4.2303	4.2236	4.1146
10	4.2494	4.1105	4.2301	4.1255	4.2725	4.1685
11	4.3235	3.9997	4.3088	4.1971	4.315	4.2121
12	4.3511	4.2594	4.3379	4.2234	4.2563	4.1449
13	4.3030	4.2143	4.2854	4.1759	4.2584	4.1499
14	4.2059	4.0648	4.1844	4.0837	4.274	4.1673
15	4.3615	4.1738	4.3430	4.2281	4.3167	4.2062
16	4.2924	4.0921	4.2741	4.1655	4.2894	4.1797
17	4.2699	4.0908	4.2498	4.1434	4.3309	4.2125
18	4.2859	4.1681	4.2628	4.1552	4.2317	4.118
19	4.3716	4.1357	4.3504	4.2348	4.2779	4.1802
20	4.3958	4.2807	4.3806	4.2620	4.2119	4.1137
mean	4.2977	4.1420	4.2797	4.1706	4.2617	4.1539
std	0.0519	0.0784	0.0524	0.0476	0.0357	0.0349
MSE	**0.0171**	**0.0074225**	**0.01315**	**0.00232**	**0.00832**	**0.00178**

Table 5d: Initial stock price 110, n=16000

true value=6.4172(0.0447)

	Δt =1		Δt =8.5			
	quasi monte carlo		quasi monte carlo		monte carlo	
s0=110	DG	DGSK	DG	DGSK	DG	DGSK
1	6.5867	6.3125	6.5950	6.4511	6.5085	6.3626
2	6.4845	6.3571	6.4977	6.3608	6.9286	6.7616
3	6.4834	6.3852	6.4881	6.3518	6.4556	6.3198
4	6.5425	6.4472	6.5462	6.4058	6.6304	6.4847
5	6.5703	6.5274	6.5706	6.4285	6.5798	6.4402
6	6.6409	6.4852	6.6503	6.5024	6.6866	6.533
7	6.6184	6.4150	6.6266	6.4804	6.4292	6.3021
8	6.6216	6.5894	6.6320	6.4854	6.4713	6.3354
9	6.5329	6.4148	6.5373	6.3976	6.5037	6.376
10	6.3150	6.1661	6.3190	6.1941	6.5177	6.3774
11	6.5440	6.3716	6.5504	6.4097	6.5003	6.3632
12	6.4856	6.2208	6.4878	6.3515	6.6568	6.5103
13	6.6359	6.5163	6.6475	6.4998	6.5947	6.4499
14	6.5752	6.4990	6.5848	6.4417	6.6408	6.4911
15	6.5855	6.3741	6.5869	6.4437	6.5105	6.3758
16	6.4674	6.2318	6.4761	6.3405	6.7085	6.556
17	6.4655	6.4253	6.4695	6.3345	6.5235	6.3829
18	6.5870	6.3897	6.5922	6.4485	6.5266	6.3887
19	6.7403	6.4568	6.7511	6.5956	6.5745	6.4256
20	6.5032	6.5004	6.5081	6.3704	6.5182	6.379
mean	6.5493	6.4043	6.5559	6.4147	6.5733	6.4308
std	0.0892	0.1089	0.0907	0.0843	0.1140	0.1056
MSE	**0.0254**	**0.0120274**	**0.02746**	**0.00711**	**0.03735**	**0.01133**

Table 5e: Initial stock price 120, n=16000

true value=7.1985(0.0496)

	Δt =1		Δt =8.5			
	quasi monte carlo		quasi monte carlo		monte carlo	
s0=120	DG	DGSK	DG	DGSK	DG	DGSK
1	7.3153	7.4356	7.3140	7.2758	7.1099	7.069
2	7.2991	7.3473	7.2996	7.2617	7.0124	6.976
3	7.2946	7.3366	7.2960	7.2582	7.4399	7.3993
4	7.1913	7.3091	7.1981	7.1619	7.1374	7.1119
5	7.3337	7.3736	7.3291	7.2907	7.1334	7.0939
6	7.1459	6.9631	7.1444	7.1091	7.2796	7.234
7	7.2684	6.9268	7.2758	7.2382	7.274	7.2418
8	7.2290	7.0963	7.2331	7.1963	7.2403	7.2078
9	7.2171	7.2768	7.2275	7.1908	7.3833	7.3372
10	7.0643	7.0516	7.0664	7.0323	7.1092	7.0776
11	7.2317	7.1884	7.2274	7.1907	7.3956	7.351
12	7.0761	6.8253	7.0775	7.0432	7.3572	7.3173
13	7.2021	7.0837	7.2009	7.1646	7.1968	7.1656
14	7.1814	6.9615	7.1803	7.1443	7.1204	7.0957
15	7.2840	7.2907	7.2899	7.2522	7.2568	7.2193
16	7.1634	7.0790	7.1611	7.1254	7.2079	7.167
17	7.1293	7.0670	7.1328	7.0977	7.2756	7.2372
18	7.4267	7.3380	7.4260	7.3858	7.4531	7.4157
19	7.1400	7.1964	7.1477	7.1123	7.1832	7.1388
20	7.3269	7.4126	7.3361	7.2976	7.0503	7.0175
mean	7.2260	7.1780	7.2282	7.1914	7.2308	7.1937
std	0.0931	0.1797	0.0929	0.0914	0.1279	0.1250
MSE	**0.0094**	**0.03270499**	**0.00952**	**0.00840**	**0.01739**	**0.01566**

Table 5a, 5b, 5c, 5d and 5e show the results using different perturbation for $\Delta t = 1$ and optimal $\Delta t = 8.5$ (from the theorem in chapter 2). Here, we make a comparison between Quasi Monte Carlo simulation with $\Delta t = 1$ and both Quasi Monte Carlo simulation and Monte Carlo simulation with optimal $\Delta t = 8.5$.

For your information, we carried out the experiment by using both Delta-Gamma approach and Delta-Gamma-Skewness-Kurtosis approach for each simulation. For example, if we perform 20 times Quasi Monte Carlo simulation with Delta-Gamma approximation, we will repeat the same experiments by changing Delta-Gamma approach to Delta-Gamma-Skewness-Kurtosis approach.

Firstly, we compare VaR using $\Delta t = 1$ and optimal $\Delta t = 8.5$. We focus on the first four columns in Table 5a, 5b, 5c, 5d and 5e. Here we can see that when we fixed the method by choosing Quasi Monte Carlo simulation and for each Δt, we come out with Delta-Gamma approximation and Delta-Gamma-Skewness-Kurtosis approach. From the results, it showed that Delta-Gamma-Skewness-Kurtosis approach with optimal perturbation has the smallest MSE compared to others. Also, when we look at the experiments with optimal $\Delta t = 8.5$, again Quasi Monte Carlo simulation with Delta-Gamma-Skewness-Kurtosis performs very well compared to Monte Carlo simulation with Delta-Gamma-Skewness-Kurtosis approach.

Both theoretical and numerical results show that Delta-Gamma-Skewness-Kurtosis model together with Quasi Monte Carlo simulation provides a more accurate result than Delta-Gamma only. This new method may perform better when we choose the perturbation as introduced.

CHAPTER 4 CONCLUSIONS

In general, Delta-Gamma-Skewness-Kurtosis model is a very good approach in calculating VaR for non-linear portfolio. It overcomes the problem of poor convergence rate which faced by Delta-Gamma approach. Additionally, it is less-time-consuming compare to other traditional approach like Heston (1993) approach.

Besides that, it leads to a faster convergence rate when Quasi Monte Carlo simulation is chosen. The numerical results suggest that Quasi Monte Carlo method coupled with Sobol sequence can lead to a great variance reduction effects over standard Monte Carlo.

This Delta-Gamma-Skewness-Kurtosis approach is very straightforward and easy to implement in practice. Numerical examples show the proposed method performs very well when perturbation is chosen as suggested in theory. Moreover, the speed of this approach is another attraction to the user. We do not have to use a couple of days to simulate the results and in contrast we can get all the results in a few minutes time.

In this thesis, we focused on one dimension portfolio and it can be furthered to moderate high dimension portfolio in future. Also, this thesis can be extended to consider the effect of other variance reduction technique on Monte Carlo and Sobol sequence.

APPENDIX A

Proof of convergence rate

In general, we have

$$f_1 = f_0 + f_0^{(1)}T + f_0^{(2)}\frac{T^2}{2!} + \cdots + f_0^{(2n)}\frac{T^{(2n)}}{(2n)!} + f_0^{(2n+1)}\frac{T^{(2n+1)}}{(2n+1)!} + f_0^{(2n+2)}\frac{T^{(2n+2)}}{(2n+2)!} + o(T^{(2n+2)}) \ ,$$

$$f_{-1} = f_0 - f_0^{(1)}T + f_0^{(2)}\frac{T^2}{2!} - \cdots + f_0^{(2n)}\frac{T^{(2n)}}{(2n)!} - f_0^{(2n+1)}\frac{T^{(2n+1)}}{(2n+1)!} + f_0^{(2n+2)}\frac{T^{(2n+2)}}{(2n+2)!} + o(T^{(2n+2)}) \ ,$$

$$\vdots$$

$$f_n = f_0 + f_0^{(1)}(nT) + f_0^{(2)}\frac{(nT)^2}{2!} + \cdots + f_0^{(2n)}\frac{(nT)^{(2n)}}{(2n)!} + f_0^{(2n+1)}\frac{(nT)^{(2n+1)}}{(2n+1)!} + f_0^{(2n+2)}\frac{(nT)^{(2n+2)}}{(2n+2)!}$$
$$+ o(T^{(2n+2)}),$$

$$f_{-n} = f_0 - f_0^{(1)}(nT) + f_0^{(2)}\frac{(nT)^2}{2!} - \cdots + f_0^{(2n)}\frac{(nT)^{(2n)}}{(2n)!} - f_0^{(2n+1)}\frac{(nT)^{(2n+1)}}{(2n+1)!} + f_0^{(2n+2)}\frac{(nT)^{(2n+2)}}{(2n+2)!}$$
$$+ o(T^{(2n+2)}) \ ,$$

- (A1)

Rewrite the equation **(A1)** in the following form:

$$
\begin{bmatrix}
f_1 - f_0 - \dfrac{T^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \dfrac{T^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \\
f_{-1} - f_0 + \dfrac{T^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \dfrac{T^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \\
\vdots \\
f_n - f_0 - \dfrac{(nT)^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \dfrac{(nT)^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \\
f_{-n} - f_0 + \dfrac{(nT)^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \dfrac{(nT)^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)})
\end{bmatrix}
=
\begin{bmatrix}
T & \dfrac{T^2}{2!} & \cdots & \dfrac{T^{2n}}{(2n)!} \\
-T & \dfrac{(-T)^2}{2!} & \cdots & \dfrac{(-T)^{2n}}{(2n)!} \\
 & \vdots & & \vdots \\
nT & \dfrac{(nT)^2}{2!} & \cdots & \dfrac{(nT)^{2n}}{(2n)!} \\
-nT & \dfrac{(-nT)^2}{2!} & \cdots & \dfrac{(-nT)^{2n}}{(2n)!}
\end{bmatrix}
\begin{bmatrix}
f_0^{(1)} \\ f_0^{(2)} \\ \vdots \\ f_0^{(2n-1)} \\ f_0^{(2n)}
\end{bmatrix}
$$

Denote

$$x_1 = f_1 - f_0 - \frac{T^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \frac{T^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \ ,$$

$$x_{-1} = f_{-1} - f_0 + \frac{T^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \frac{T^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \ ,$$

$$\vdots$$

$$x_n = f_n - f_0 - \frac{(nT)^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \frac{(nT)^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \ ,$$

$$x_{-n} = f_{-n} - f_0 + \frac{(nT)^{(2n+1)}}{(2n+1)!} f_0^{(2n+1)} - \frac{(nT)^{(2n+2)}}{(2n+2)!} f_0^{(2n+2)} + o(T^{(2n+2)}) \ .$$

Using Cramer rule we can solve $f_0^{(k)}$ by replacing *kth* column with $x_1, \cdots, x_{-n}$

$$
f_0^{(k)} = \frac{\begin{vmatrix}
T & \frac{T^2}{2!} & \cdots & x_1 & \cdots & \frac{T^{2n}}{(2n)!} \\
-T & \frac{(-T)^2}{2!} & \cdots & x_{-1} & \cdots & \frac{(-T)^{2n}}{(2n)!} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
nT & \frac{(nT)^2}{2!} & \cdots & x_n & \cdots & \frac{(nT)^{2n}}{(2n)!} \\
-nT & \frac{(-nT)^2}{2!} & \cdots & x_{-n} & \cdots & \frac{(-nT)^{2n}}{(2n)!}
\end{vmatrix}}{\begin{vmatrix}
T & \frac{T^2}{2!} & \frac{T^3}{3!} & \cdots & \frac{T^{2n}}{(2n)!} \\
-T & \frac{(-T)^2}{2!} & \frac{(-T)^3}{3!} & \cdots & \frac{(-T)^{2n}}{(2n)!} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
nT & \frac{(nT)^2}{2!} & \frac{(nT)^3}{3!} & \cdots & \frac{(nT)^{2n}}{(2n)!} \\
-nT & \frac{(-nT)^2}{2!} & \frac{(-nT)^3}{3!} & \cdots & \frac{(-nT)^{2n}}{(2n)!}
\end{vmatrix}}
$$

$$
f_0^{(k)} = \frac{1}{T^k} \frac{\begin{vmatrix} 1 & \frac{1^2}{2!} & \cdots & x_1 & \cdots & \frac{1^{2n}}{(2n)!} \\ -1 & \frac{(-1)^2}{2!} & \cdots & x_{-1} & \cdots & \frac{(-1)^{2n}}{(2n)!} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ n & \frac{(n)^2}{2!} & \cdots & x_n & \cdots & \frac{(n)^{2n}}{(2n)!} \\ -n & \frac{(-n)^2}{2!} & \cdots & x_{-n} & \cdots & \frac{(-n)^{2n}}{(2n)!} \end{vmatrix}}{\begin{vmatrix} 1 & \frac{1^2}{2!} & \frac{1^3}{3!} & \cdots & \frac{1^{2n}}{(2n)!} \\ -1 & \frac{(-1)^2}{2!} & \frac{(-1)^3}{3!} & \cdots & \frac{(-1)^{2n}}{(2n)!} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ n & \frac{(n)^2}{2!} & \frac{(n)^3}{3!} & \cdots & \frac{(n)^{2n}}{(2n)!} \\ -n & \frac{(-n)^2}{2!} & \frac{(-n)^3}{3!} & \cdots & \frac{(-n)^{2n}}{(2n)!} \end{vmatrix}}
$$

$$
f_0^{(k)} = \frac{k!}{T^k} \frac{\begin{vmatrix} 1 & 1 & \cdots & x_1 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & x_{-1} & \cdots & (-1)^{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & x_n & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & x_{-n} & \cdots & (-n)^{2n} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & & 1 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & (-1)^k & \cdots & (-1)^{2n} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & (n)^k & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & (-n)^k & \cdots & (-n)^{2n} \end{vmatrix}}
$$

$$
f_0^{(k)} = \frac{k!}{T^k} \sum_{\substack{i=-n \\ i \neq 0}}^{n} \frac{\Delta_i}{\Delta} x_i \ ,
$$

where $\Delta_i = (-1)^{k+i}\left|\Delta(-i,-k)\right|$,

$$\Delta = \begin{vmatrix} 1 & 1 & \cdots & 1^k & \cdots & 1^{2n} \\ -1 & (-1)^2 & \cdots & (-1)^k & \cdots & (-1)^{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ n & n^2 & \cdots & n^k & \cdots & n^{2n} \\ -n & (-n)^2 & \cdots & (-n)^k & \cdots & (-n)^{2n} \end{vmatrix},$$

$$f_0^{(k)} = \frac{k!}{T^k}\frac{\begin{vmatrix} 1 & 1 & \cdots & f_1-f_0 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & f_{-1}-f_0 & \cdots & (-1)^{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & f_n-f_0 & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & f_{-n}-f_0 & \cdots & (-n)^{2n} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & & 1 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & (-1)^k & \cdots & (-1)^{2n} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & (n)^k & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & (-n)^k & \cdots & (-n)^{2n} \end{vmatrix}} + \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\displaystyle\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta} i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd}, \\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\displaystyle\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta} i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even}, \end{cases}$$

Our simulation

$$\overline{Y}^{(k)} = \frac{k!}{T^k}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta}(x_i - x_0)\ .$$

Monte Carlo

$$\hat{\Delta}_c = Y^{(k)} = \frac{k!}{T^k}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta}\left[\sum_{j=1}^{m}\frac{(x_i^j - x_0^j)}{m}\right]$$

$$= \frac{k!}{T^k}\frac{1}{m}\left[\sum_{j=1}^{m}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta}(x_i^j - x_0^j)\right].$$

(A2)

Consider

$Y_j^{(k)} = \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(x_i^j - x_0^j)$ and $\hat{Y}_j^{(k)} = Y_j^{(k)} - EY_j^{(k)}$ we need to find variance of $\hat{Y}_j^{(k)}$ where

$Var\ \hat{Y}_j^{(k)} = E\left[Y_j^{(k)}\right]^2 - (EY_j^{(k)})^2$.

$$
\begin{aligned}
E\left[Y_j^{(k)}\right] &= E\left[\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(x_i^j - x_0^j)\right] \\
&= \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(E(x_i^j) - E(x_0^j)) \\
&= \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(f_i - f_0) \\
&= \frac{\begin{vmatrix} 1 & 1 & \cdots & f_1 - f_0 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & f_{-1} - f_0 & \cdots & (-1)^{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & f_n - f_0 & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & f_{-n} - f_0 & \cdots & (-n)^{2n} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & & 1 & \cdots & 1 \\ -1 & (-1)^2 & \cdots & (-1)^k & \cdots & (-1)^{2n} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ n & (n)^2 & \cdots & (n)^k & \cdots & (n)^{2n} \\ -n & (-n)^2 & \cdots & (-n)^k & \cdots & (-n)^{2n} \end{vmatrix}},
\end{aligned}
$$

$$
E\left[Y_j^{(k)}\right] = \frac{T^k}{k!}\left[f_0^{(k)} - \begin{cases} \dfrac{k!\,f_0^{(2n+1)}}{(2n+1)!}\displaystyle\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta} i^{2n+1} T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd} \\ -\dfrac{k!\,f_0^{(2n+2)}}{(2n+2)!}\displaystyle\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta} i^{2n+2} T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even} \end{cases} \right].
$$

From $Y_j^{(k)} = \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(x_i^j - x_0^j)$,

$$\left[Y_j^{(k)}\right]^2 = \left[\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta}(x_i^j - x_0^j)\right]^2$$

$$= \sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i}{\Delta}(x_i^j - x_0^j)\frac{\Delta_p}{\Delta}(x_p^j - x_0^j) + \sum_{\substack{i=-n \\ i\neq 0}}^{n} (\frac{\Delta_i}{\Delta}(x_i^j - x_0^j))^2 \ .$$

Take expectation for both side, we get

$$E\left[Y_j^{(k)}\right]^2 = \sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i}{\Delta}(Ex_i^j - Ex_0^j)\frac{\Delta_p}{\Delta}(Ex_p^j - Ex_0^j) + \sum_{\substack{i=-n \\ i\neq 0}}^{n} (\frac{\Delta_i}{\Delta}E(x_i^j - x_0^j))^2$$

$$= \sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i\Delta_p}{\Delta^2}(Ex_i^j - Ex_0^j)(Ex_p^j - Ex_0^j) + \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2}T\sigma_i^2 + o(T)$$

$$= \sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i\Delta_p}{\Delta^2}[f(x_0+iT) - f(x_0)][f(x_0+pT) - f(x_0)] + \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2}T\sigma_i^2 + o(T) \ .$$

Equivalently,

$$E\left[Y_j^{(k)}\right]^2 = \sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i\Delta_p}{\Delta^2}[f_0^{(1)}iT + o(T)][f_0^{(1)}pT + o(T)] + \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2}T\sigma_i^2 + o(T) \ . \quad \textbf{-(A3)}$$

As $\sum_{\substack{i\neq p \\ i,p=-n \\ i,p\neq 0}}^{n} \frac{\Delta_i\Delta_p}{\Delta^2}[f_0^{(1)}iT + o(T)][f_0^{(1)}pT + o(T)] \to o(T)$

Equation **(A3)** is then become

$$E\left[Y_j^{(k)}\right]^2 = \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2}T\sigma_i^2 + o(T) \ .$$

So we have

$$Var\ \hat{Y}_j^{(k)} = \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2} T\sigma_i^2 + o(T) + \left(\frac{T^k}{k!} \left[f_0^{(k)} - \begin{cases} \dfrac{k!\,f_0^{(2n+1)}}{(2n+1)!} \displaystyle\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta} i^{2n+1} T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd} \\ -\dfrac{k!\,f_0^{(2n+2)}}{(2n+2)!} \displaystyle\sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i}{\Delta} i^{2n+2} T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even} \end{cases} \right] \right)^2$$

$$= \sum_{\substack{i=-n \\ i\neq 0}}^{n} \frac{\Delta_i^2}{\Delta^2} T\sigma_i^2 + o(T)\ .$$

From equation **(A2)**

$$Y^{(k)} = \frac{k!}{T^k}\frac{1}{m}\left[\sum_{j=1}^{m}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta}(x_i^j - x_0^j)\right].$$

It also can be written as

$$\begin{aligned} Y^{(k)} &= \frac{k!}{T^k}\frac{1}{m}\sum_{j=1}^{m} Y_j^{(k)} \\ &= \frac{k!}{T^k}\frac{1}{m}\left[\sum_{j=1}^{m}(\hat{Y}_j^{(k)} + EY_j^{(k)})\right] \\ &= \frac{k!}{T^k}\frac{1}{m}\sum_{j=1}^{m}\hat{Y}_j^{(k)} + \frac{k!}{T^k}EY_j^{(k)} \ . \end{aligned}$$

$$\begin{aligned} \frac{k!}{T^k}EY_j^{(k)} &= \frac{k!}{T^k}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\frac{\Delta_i}{\Delta}(f_i - f_0) \\ &= \frac{k!}{T^k}\left[\frac{T^k}{k!}\left(f_0^{(k)} - \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd} \\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even} \end{cases}\right)\right] \\ &= f_0^{(k)} - \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd}\ , \\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even}\ . \end{cases} \end{aligned}$$

$$Y^{(k)} = \frac{k!}{T^k}\frac{1}{m}\sum_{j=1}^{m}\hat{Y}_j^{(k)} + f_0^{(k)} - \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd}\ , \\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\sum_{\substack{i=-n\\ i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even}\ . \end{cases}$$

$$Y^{(k)} - f_0^{(k)} = \frac{k!}{T^k}\frac{1}{m}\sum_{j=1}^{m}\hat{Y}_j^{(k)} - \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\sum_{\substack{i=-n\\i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd },\\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\sum_{\substack{i=-n\\i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even },\end{cases}$$

$$= \frac{k!}{T^{k-\frac{1}{2}}m^{\frac{1}{2}}}\frac{\sum_{j=1}^{m}\hat{Y}_j^{(k)}}{\sqrt{T}\sqrt{m}} - \begin{cases} \dfrac{k!f_0^{(2n+1)}}{(2n+1)!}\sum_{\substack{i=-n\\i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+1}T^{2n+1-k} + o(T^{2n+1-k}), k \text{ odd },\\ -\dfrac{k!f_0^{(2n+2)}}{(2n+2)!}\sum_{\substack{i=-n\\i\neq 0}}^{n}\dfrac{\Delta_i}{\Delta}i^{2n+2}T^{2n+2-k} + o(T^{2n+2-k}), k \text{ even },\end{cases}$$

$$T^{k-\frac{1}{2}}m^{\frac{1}{2}-z} = T^{2n+1-k}m^z,$$

$$z = \begin{cases} \dfrac{2n+1-k}{4n+1}, k \text{ odd },\\ \dfrac{2n+2-k}{4n+3}, k \text{ even }.\end{cases}$$

Hence, the convergence rate is

$o(m^{-\frac{2n+1-k}{4n+1}}), k$ odd ,

$o(m^{-\frac{2n+3-k}{4n+3}}), k$ even .

m is the sample size.

APPENDIX B

Proof of optimal Δt

Based on Taylor series, we can write the following equation for our DGSK model. (Assume that $f_0^{(5)} \neq 0$ and $f_0^{(6)} \neq 0$)

$$f_1 = f_0 + Tf_0^{(1)} + \frac{T^2}{2!} f_0^{(2)} + \frac{T^3}{3!} f_0^{(3)} + \frac{T^4}{4!} f_0^{(4)} + \frac{T^5}{5!} f_0^{(5)} + \frac{T^6}{6!} f_0^{(6)} + o(T^6) \ ,$$

$$f_{-1} = f_0 - Tf_0^{(1)} + \frac{T^2}{2!} f_0^{(2)} - \frac{T^3}{3!} f_0^{(3)} + \frac{T^4}{4!} f_0^{(4)} - \frac{T^5}{5!} f_0^{(5)} + \frac{T^6}{6!} f_0^{(6)} + o(T^6) \ ,$$

$$f_2 = f_0 + 2Tf_0^{(1)} + \frac{(2T)^2}{2!} f_0^{(2)} + \frac{(2T)^3}{3!} f_0^{(3)} + \frac{(2T)^4}{4!} f_0^{(4)} + \frac{(2T)^5}{5!} f_0^{(5)} + \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6),$$

$$f_{-2} = f_0 - 2Tf_0^{(1)} + \frac{(2T)^2}{2!} f_0^{(2)} - \frac{(2T)^3}{3!} f_0^{(3)} + \frac{(2T)^4}{4!} f_0^{(4)} - \frac{(2T)^5}{5!} f_0^{(5)} + \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6).$$

-(B1)

where f_k , $k = \pm 1, \pm 2, ..., \pm n$ denotes the value of $f(t)$ at $t = kT$, $f_0^{(k)}$ denotes the value of the kth derivatives of f at $t = 0$ and $o(T^{2n})$ is a term of the order of T^{2n} coming from the truncation after *2n* terms. Here I use n=2. Rewrite equation **(B1)** as

$$\begin{bmatrix} f_1 - f_0 - \frac{T^5}{5!} f_0^{(5)} - \frac{T^6}{6!} f_0^{(6)} + o(T^6) \\ f_{-1} - f_0 + \frac{T^5}{5!} f_0^{(5)} - \frac{T^6}{6!} f_0^{(6)} + o(T^6) \\ f_2 - f_0 - \frac{(2T)^5}{5!} f_0^{(5)} - \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6) \\ f_{-2} - f_0 + \frac{(2T)^5}{5!} f_0^{(5)} - \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6) \end{bmatrix} = \begin{bmatrix} T & \frac{T^2}{2!} & \frac{T^3}{3!} & \frac{T^4}{4!} \\ -T & \frac{(-T)^2}{2!} & \frac{(-T)^3}{3!} & \frac{(-T)^4}{4!} \\ 2T & \frac{(2T)^2}{2!} & \frac{(2T)^3}{3!} & \frac{(2T)^4}{4!} \\ -2T & \frac{(-2T)^2}{2!} & \frac{(-2T)^3}{3!} & \frac{(-2T)^4}{4!} \end{bmatrix} \begin{bmatrix} f_0^{(1)} \\ f_0^{(2)} \\ f_0^{(3)} \\ f_0^{(4)} \end{bmatrix},$$

-**(B2)**

For simplicity, let

$$x_1 = f_1 - f_0 - \frac{T^5}{5!} f_0^{(5)} - \frac{T^6}{6!} f_0^{(6)} + o(T^6) \ ,$$
$$x_{-1} = f_{-1} - f_0 + \frac{T^5}{5!} f_0^{(5)} - \frac{T^6}{6!} f_0^{(6)} + o(T^6) \ ,$$
$$x_2 = f_2 - f_0 - \frac{(2T)^5}{5!} f_0^{(5)} - \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6) \ ,$$
$$x_{-2} = f_{-2} - f_0 + \frac{(2T)^5}{5!} f_0^{(5)} - \frac{(2T)^6}{6!} f_0^{(6)} + o(T^6) \ .$$

Equation **(B2)** can be written as

$$f_0^{(1)} = \frac{\begin{vmatrix} x_1 & \frac{T^2}{2!} & \frac{T^3}{3!} & \frac{T^4}{4!} \\ x_{-1} & \frac{(-T)^2}{2!} & \frac{(-T)^3}{3!} & \frac{(-T)^4}{4!} \\ x_2 & \frac{(2T)^2}{2!} & \frac{(2T)^3}{3!} & \frac{(2T)^4}{4!} \\ x_{-2} & \frac{(-2T)^2}{2!} & \frac{(-2T)^3}{3!} & \frac{(-2T)^4}{4!} \end{vmatrix}}{\begin{vmatrix} T & \frac{T^2}{2!} & \frac{T^3}{3!} & \frac{T^4}{4!} \\ -T & \frac{(-T)^2}{2!} & \frac{(-T)^3}{3!} & \frac{(-T)^4}{4!} \\ 2T & \frac{(2T)^2}{2!} & \frac{(2T)^3}{3!} & \frac{(2T)^4}{4!} \\ -2T & \frac{(-2T)^2}{2!} & \frac{(-2T)^3}{3!} & \frac{(-2T)^4}{4!} \end{vmatrix}},$$

$$f_0^{(1)} = \frac{1}{T} \frac{\begin{vmatrix} x_1 & \frac{1^2}{2!} & \frac{1^3}{3!} & \frac{1^4}{4!} \\ x_{-1} & \frac{(-1)^2}{2!} & \frac{(-1)^3}{3!} & \frac{(-1)^4}{4!} \\ x_2 & \frac{(2)^2}{2!} & \frac{(2)^3}{3!} & \frac{(2)^4}{4!} \\ x_{-2} & \frac{(-2)^2}{2!} & \frac{(-2)^3}{3!} & \frac{(-2)^4}{4!} \end{vmatrix}}{\begin{vmatrix} 1 & \frac{1^2}{2!} & \frac{1^3}{3!} & \frac{1^4}{4!} \\ -1 & \frac{(-1)^2}{2!} & \frac{(-1)^3}{3!} & \frac{(-1)^4}{4!} \\ 2 & \frac{(2)^2}{2!} & \frac{(2)^3}{3!} & \frac{(2)^4}{4!} \\ -2 & \frac{(-2)^2}{2!} & \frac{(-2)^3}{3!} & \frac{(-2)^4}{4!} \end{vmatrix}},$$

$$
= \frac{1}{T} \frac{\begin{vmatrix} x_1 & 1 & 1 & 1 \\ x_{-1} & (-1)^2 & (-1)^3 & (-1)^4 \\ x_2 & (2)^2 & (2)^3 & (2)^4 \\ x_{-2} & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & (-1)^2 & (-1)^3 & (-1)^4 \\ 2 & (2)^2 & (2)^3 & (2)^4 \\ -2 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}}
$$

$$
= \frac{1}{T} \sum_{\substack{i=-2 \\ i \neq 0}}^{2} \frac{\Delta_i}{\Delta} x_i \ ,
$$

where

$$
\Delta_i = (-1)^{-1+i} \left| \Delta(-i, -1) \right|,
$$

$$
\Delta = \begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & (-1)^2 & (-1)^3 & (-1)^3 \\ 2 & 2^2 & 2^3 & 2^4 \\ -2 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}.
$$

Also it can be written as

$$
f_0^{(1)} = \frac{1}{T} \frac{\begin{vmatrix} f_1 - f_0 & 1 & 1 & 1 \\ f_{-1} - f_0 & (-1)^2 & (-1)^3 & (-1)^4 \\ f_2 - f_0 & 2^2 & 2^3 & 2^4 \\ f_{-2} - f_0 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & (-1)^2 & (-1)^3 & (-1)^3 \\ 2 & 2^2 & 2^3 & 2^4 \\ -2 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}} + \frac{f_0^{(5)}}{5!} \sum_{\substack{i=-2 \\ i \neq 0}}^{2} \frac{\Delta_i}{\Delta} i^5 T^4 + o(T^4) \ ,
$$

Denotes our stock price which generated by Monte Carlo simulation by x_i^j ,
$i = 0, \pm 1, \pm 2$,
$j = 1, ..., m$. (m is the number of sample path)

Assumptions
(1) $x_i^j - x_0^j$ independent and i.i.d ;
(2) $y_j^{(1)}$ i.i.d, $j = 1, ..., m$;
(3) $E(x_i^j - x_0^j) = T\sigma_i^2 + o(T), j = 1, 2, ..., m$.

$$\text{Let} \quad \hat{\Delta}_c = \frac{1}{T}\sum_{\substack{i=-2 \\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta}\left[\sum_{j=1}^{m}\frac{(x_i^j - x_0^j)}{m}\right]$$

$$= \frac{1}{Tm}\sum_{j=1}^{m}\sum_{\substack{i=-2 \\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta}(x_i^j - x_0^j)$$

$$= \frac{1}{Tm}\sum_{j=1}^{m} y_j^{(1)} ,$$

where $y_j^{(1)} = \sum_{\substack{i=-2 \\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta}(x_i^j - x_0^j)$.

We know that $E(y_j^{(1)}) = \sum_{\substack{i=-2 \\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta}(Ex_i^j - Ex_0^j)$

$$= \sum_{\substack{i=-2 \\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta}(f_i - f_0) .$$

$$\text{Also } E(y_j^{(1)}) = \frac{\begin{vmatrix} f_1 - f_0 & 1 & 1 & 1 \\ f_{-1} - f_0 & (-1)^2 & (-1)^3 & (-1)^4 \\ f_2 - f_0 & 2^2 & 2^3 & 2^4 \\ f_{-2} - f_0 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & (-1)^2 & (-1)^3 & (-1)^4 \\ 2 & 2^2 & 2^3 & 2^4 \\ -2 & (-2)^2 & (-2)^3 & (-2)^4 \end{vmatrix}} ,$$

$$E(y_j^{(1)}) = T\left[f_0^{(1)} - \frac{f_0^{(5)}}{5!}\sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta} i^5 T^4 + o(T^4)\right],$$

$$E(\hat{\Delta}_c) = \left[f_0^{(1)} - \frac{f_0^{(5)}}{5!}\sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta} i^5 T^4 + o(T^4)\right].$$

Then $E(\hat{\Delta}_c - f_0^{(1)}) = -\frac{f_0^{(5)}}{5!}\sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i}{\Delta} i^5 T^4 + o(T^4)$.

Now we calculate the variance

$$Var(\hat{\Delta}_c) = Var\left(\frac{1}{Tm}\sum_{j=1}^{m} y_j^{(1)}\right)$$

$$= \frac{1}{T^2} Var(y_j^{(1)}),$$

$$Var(y_j^{(1)}) = E(y_j^{(1)})^2 - (E(y_j^{(1)}))^2$$

$$= \sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i^2}{\Delta^2} T\sigma_i^2 + o(T),$$

$$Var(\hat{\Delta}_c) = \frac{1}{T^2}\left(\sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i^2}{\Delta^2} T\sigma_i^2 + o(T)\right)$$

$$= \sum_{\substack{i=-2\\ i\neq 0}}^{2}\frac{\Delta_i^2}{\Delta^2}\frac{\sigma_i^2}{T} + o(T^{-1}).$$

According to Chapter 7 in Glasserman (2003), optimal Δt can be obtained by the followings:

$$\Delta t = \left[\frac{\sigma^2}{8\left(\frac{f_0^{(5)}}{5!} \sum_{\substack{i=-2 \\ i \neq 0}}^{2} \frac{\Delta_i}{\Delta} i^5\right)^2} \right]^{\frac{1}{9}}$$

$$= \left[\frac{1800\sigma^2}{\left(\sum_{\substack{i=-2 \\ i \neq 0}}^{2} \frac{\Delta_i}{\Delta} i^5\right)^2 \left(f_0^{(5)}\right)^2} \right]^{\frac{1}{9}} .$$

BIBLIOGRAPHY

[1] Abken, P., 2000. An empirical evaluation of Value-at-Risk by scenario simulation. *Journal of Derivatives* 7 12-30.

[2] Avramidis, A. N. and J.R. Wilson, 1996. Integrated variance reduction strategies for simulation. *Operations Research* 44 327-346.

[3] Avramidis, A. N. and J.R. Wilson, 1998. Correlation-induction techniques for estimating quantiles in simulation experiments. *Operations Research* 46 574-591.

[4] Britten-Jones, M. and S.M.Schaefer, 1999. Non-linear Value-at-Risk. *European Finance Review* 2 161-187.

[5] Boyle, P., M. Broadie, P. Glausserman, 1997. Monte Carlo Methods for Security Pricing. *Journal of Economic Dynamics and Control*.

[6] Duffie, D. and J.Pan, 1997. An overview of Value at Risk. *The Journal of Derivatives* 4(3) 7-49.

[7] Duffie, D. and J.Pan, 1999. Analytical Value-at-Risk with Jumps and Credit Risk. *Working Paper, Graduate School of Business, Stanford University*.

[8] Fallon, William, 1996. Calculating Value-At-Risk. *Working Paper*. *Columbia University*.

[9] Glasserman, P., P.Heidelberger and P. Shahabuddin, 1999. Importance sampling and Stratification for Value-at-Risk. In *Computational Finance*, *(Proceeding of the Sixth International Conference on Computational Finance)*.

[10] Glasserman, P., P.Heidelberger and P. Shahabuddin, 2000. Variance ReductionTechniques for estimating Value-at-Risk. *Management Science* 46 1349-1364.

[11] Glasserman, P., P.Heidelberger and P. Shahabuddin, 2002. Portfolio Value - at-Risk with Heavy-Tailed Risk Factors. *Mathematical Finance* 12(3) 239-269.

[12] Glasserman, P., 2003. Monte Carlo Methods in Financial Engineering. *Springer-Verlag Berlin and Heidelberg GmbH and Co*.

[13] Glynn, P.W., 1989. Optimization of Stochastic System via Simulation. in *Proceeding of the Winter Simulation Conference, IEEE Press, New York* 180-185.

[14] Hesterberg, T.C. and B.L.Nelson, 1998. Control variates for probability and quantile estimation. *Management Science* 44 1295-1312.

[15] Heston, S., 1993. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies* 6 327-343.

[16] Hsu, J.C. and B.L. Nelson, 1990. Control variates for quantile estimation. *Management Science* 36 835-851.

[17] Jahel, L.E., W. Perraudin and P. Sellin, 1997. Value-At-Risk for derivatives. *Working Paper*. *Birkbeck College*, *University of London*.

[18] Jamshidian, F. and Y. Zhu, 1997. Scenario simulation: theory and methodology. *Finance and Stochastics* 1 43-67.

[19] Jin Xing, Allen Zhang, 2004. Reclaiming Quasi-Monte Carlo Efficiency in Portfolio Value-at-Risk Simulation. In forthcoming *Management Science*.

[20] Joy, C., P.P. Boyle , P.P. and K.S. Tan, 1996. Quasi-Monte Carlo methods in Numerical Finance. *Journal of Management Scien*ce 42(6).

[21] Khan, I.R. and R. Ohba, 1999. Closed form expressions for the finite difference approximations of first and higher derivatives based on Taylor series. *Journal of Computational and Applied Mathematics* 107 179-193.

[22] Khan, I.R. and R. Ohba, 2001. New finite difference formulas for numerical differentiation. *Journal of Computational and Applied Mathematics* 126 269-276.

[23] Khan, I.R. and R. Ohba, 2003. Taylor series based finite difference approximations of higher degree derivatives. *Journal of Computational and Applied Mathematics* 154 115-124.

[24] Michael, S.G. and Matthew, P., 2000. Improving Grid-based Methods for Estimating Value at Risk of Fixed-Income Portfolios.

[25] Owen, A.B. and Y. Zhou, 1998. Safe and effective importance sampling. *Working Paper*. *Stanford University*.

[26] Rounvinez, C., 1997. Going Greek with VaR. *Risk* 10(2) 57-65.

[27] Wilson, T., 1994. Plugging the gap. *Risk* 7 74-80.

[28] Zazanis, M.A. and Suri, R., 1993. Convergence Rates of Finite Difference Sensitivity Estimates for Stochastic Systems. *Operations Research* 41(4) 694-703.