Monte Carlo forecasting from CIR square root diffusion models

Dario Cziraky
Email: Dario.Ciraki@ubs.com

Sergei Kucherenko
Email: s.kucherenko@broda.co.uk

Abstract

We compare empirical convergence of Monte Carlo and quasi Monte Carlo estimates of the *H*-period forecasts using the Cox-Ingersoll-Ross square root diffusion model. The behaviour of the quasi Monte Carlo estimates in high dimensions ($H = 250$) using both Euler scheme and the Brownian bridge discretisation is analysed. We find that quasi Monte Carlo estimator displays much higher convergence rate compared to Monte Carlo regardless of the forecasting horizon (dimension) and discretisation method used. We show that for the Cox-Ingersoll-Ross square root diffusion model quasi Monte Carlo outperforms Monte Carlo without using the Brownian bridge technique even in high dimensions.

Keywords: Cox Ingersoll Ross model, Diffusion models, quasi Monte Carlo, Sobol sequence

1 Introduction

The square root diffusion model was initially proposed by Cox, Ingersoll and Ross (1985) as an interest rate model, given by a univariate continuous-time diffusion process

$$dy_t = (\alpha + \beta y_t)\,dt + \sigma\sqrt{y_t}\,dW_t, \tag{1}$$

where $\lambda > 0$, $\mu > 0$, $\sigma > 0$, and $W(t)$ is a scalar Brownian motion. The model (1) is frequently used in the literature to simulate future bond prices, interest rates, or financial derivatives using Monte Carlo (MC) methods (Andersen et al, 2002; Andersen and Lund, 1997; Bacinello, 2000; Barone-Adesi and Sorwar, 2002; Broadie and Kaya, 2003; Gkamas, 2001; Zhang and Shu, 2003).

The MC simulation is typically implemented by using Euler discretisation of the form

$$y_{t+\Delta t} - y_t = (\alpha + \beta y_t)\Delta t + \sigma\sqrt{y_t}\sqrt{\Delta t}\varepsilon_t. \tag{2}$$

While the diffusion coefficient in the square root diffusion model does not satisfy a global Lipschitz condition, Higham and Mao (2005) have shown that Euler discretisation provide correct approximation to the first and second moments. They also derived a strong convergence result that justifies MC simulations using square root diffusion model.

In this paper we are interested in the empirical convergence properties of the QMC methods (QMC) comparative to pseudo MC for the square root diffusion process. Moreover, we investigate the behaviour of the QMC estimates in high dimensions (H = 250) using both Euler scheme and the Brownian bridge discretisation.

The paper is organised as follows. The second section describes the forecasting algorithm based on MC simulations from the square root diffusion model and describes a generalised method of moments estimation procedure for calibration of the model coefficients. Third section presents an empirical application to short-term interest rate forecasting. In the fourth section we compare the root mean square error

convergence rate of MC and QMC estimators. Conclusions are presented in the fifth section.

2 Monte Carlo forecasting

Consider the task of forecasting the short-term interest rate (y_t) H-period ahead from the square root diffusion model (1). Given the starting value y_T, coefficient estimates $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\sigma}$, along with an *i.i.d.* standard Gaussian vector of innovations $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_H$, the values $y_{T+1}, y_{T+1}, \ldots, y_{T+H}$ are recursively calculated as

$$\begin{aligned} E\left[y_{T+1} \mid y_T\right] &= \left(\hat{\alpha} + \hat{\beta} y_T\right)\Delta t + \hat{\sigma}\sqrt{y_T}\sqrt{\Delta t}\varepsilon_1 \\ E\left[y_{T+2} \mid y_T, y_{T+1}\right] &= \left(\hat{\alpha} + \hat{\beta} y_{T+1}\right)\Delta t + \hat{\sigma}\sqrt{y_{T+1}}\sqrt{\Delta t}\varepsilon_2 \\ &\vdots \\ E\left[y_{T+H} \mid y_T, y_{T+H-1}\right] &= \left(\hat{\alpha} + \hat{\beta} y_{T+H}\right)\Delta t + \hat{\sigma}\sqrt{y_{T+H}}\sqrt{\Delta t}\varepsilon_H . \end{aligned} \tag{3}$$

Repeating the above recursion using N independent realisations of innovations vectors $\left(\varepsilon_1^{(1)}, \varepsilon_2^{(1)}, \ldots, \varepsilon_H^{(1)}\right)$, $\left(\varepsilon_1^{(2)}, \varepsilon_2^{(2)}, \ldots, \varepsilon_H^{(2)}\right)$, …, $\left(\varepsilon_1^{(N)}, \varepsilon_2^{(N)}, \ldots, \varepsilon_H^{(N)}\right)$ we obtain N simulated H-period ahead forecasts $\hat{y}_H^{(1)}, \hat{y}_H^{(2)}, \ldots, \hat{y}_H^{(N)}$. The MC estimator of y_H is then given by the average

$$\tilde{y}_H = \frac{1}{N}\sum_{i=1}^{N} \hat{y}_H^{(i)} . \tag{4}$$

The convergence rate of MC methods is $O(N^{-1/2})$. It does not depend on the number of variables (dimensions) but it is rather low.

The efficiency of MC methods is determined by the properties of random numbers. It is known that random number sampling is prone to clustering: for any sampling there are always empty areas as well as regions in which random points are wasted due to clustering. As new points are added randomly, they do not necessarily fill the gaps between already sampled points.

A higher rate of convergence can be obtained by using deterministic uniformly distributed sequences also known as low-discrepancy sequences (LDS) instead of pseudo-random numbers. Methods based on the usage of such sequences are known as QMC methods.

LDS are specifically designed to place sample points as uniformly as possible. Unlike random numbers, successive LDS points "know" about the position of previously sampled points and "fill" the gaps between them. LDS are also known as quasi random numbers. The QMC estimator of y_H has a form similar to (4) but it this case $\{\varepsilon_t\}_{t=T+N}^{t=T}$ is a vector of LDS while in the case of MC, $\{\varepsilon_t\}_{t=T+N}^{t=T}$ is a vector of pseudo random numbers.

There are a few well-known and commonly used LDS. Different principles were used for their construction by Holton, Faure, Sobol', Niederreiter and others. Many practical studies have proven that the Sobol' LDS is in many aspects superior to other LDS (see Bratley et al, 1992, Paskov et al, 1995, Sobol', 1998). For this reason it was used in this work.

For the best known LDS the estimate for the rate of convergence $I_N \to I$ is known to be $O(\ln^H N)/N$. This rate of convergence is much faster than that for the MC method, although it depends on the dimensionality *H*. (see Glasserman, 2004, Jäckel, 2003).

We note that the recursion given in (3) is based on the Euler scheme (2). We also consider the Milstein scheme which has the discretisation of the form

$$y_{t+\Delta t} - y_t = \left(\alpha(\beta - y_t) - \frac{1}{4}\sigma \right) \Delta t + \sigma \sqrt{y_t \Delta t} \cdot \varepsilon_t + \frac{1}{4}\sigma^2 \Delta t \cdot \varepsilon_t^2 . \qquad (5)$$

Figure 1 gives an S+ function that can be used to simulate paths $y_{T+1}, y_{T+1}, \ldots, y_{T+N}$ for either Euler (**`method="euler"`**) or Milstein (**`method="milstein"`**). Note that the vector of innovations $\{\varepsilon_t\}_{t=T+N}^{t=T}$ is supplied via the **`z`** argument; hence it can be generated either as pseudo (default) or as a quasi random sequence, which needs to be pre-generated and supplied in the form of a vector.

```
CIR.sim <- function(alpha, beta, sigma, y0, dt, scheme, method, N, ToReturn,
                    index, z=rnorm(N))
{
y <- rep(NA,N);
if(scheme=="euler")
{
y[1] = y0 + alpha*(beta - y0)*dt + sigma*sqrt(y0*dt)*z[1];
       for(i in 2:N)
       {
       y[i] = y[i-1] + alpha*(beta - y[i-1])*dt + sigma*sqrt(y[i-1]*dt)*z[i];
       }
}
if(scheme=="milstein")
{
y[1] = y0 + (alpha*(beta - y0) - .25*sigma)*dt + sigma*sqrt(y0*dt)*z[1]
          + .25*(sigma^2)*dt*z[1]^2;

       for(i in 2:N)
       {
       y[i] = y[i-1]+ (alpha*(beta - y[i-1]) - .25*sigma)*dt
                    + sigma*sqrt(y[i-1]*dt)*z[i] + .25*(sigma^2)*dt*z[i]^2;
       }
}
       if(ToReturn=="last")    return(y[N]);
       if(ToReturn=="average") return(mean(y));
       if(ToReturn=="path")    return(y);
}
```

Figure 1. S+ code for simulating paths of square root diffusion process

So far we have assumed we have the coefficient estimates $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\sigma}$ available for simulation. In the following, we briefly describe the generalised method of moments (GMM) estimation procedure that can be used to obtain these estimates (see Hall, 2005). We assume that a sample of observations $y_1, y_2, \ldots, y_T$ is available for the GMM estimation.

The moment conditions we can use in GMM estimation follow from (2). In particular, we have $E\left[\varepsilon_{t+\Delta t}\right] = 0$, $E\left[\varepsilon_{t+\Delta t}^2\right] = \sigma^2 y_t^2 \Delta t$, $E\left[\varepsilon_{t+\Delta t} y_t\right] = 0$, and $E\left[\varepsilon_{t+\Delta t}^2 y_t\right] = \sigma^2 y_t^2 \Delta t y_t$.

The moment conditions must satisfy the nonlinear system of equations $E\left[g(\mathbf{w}_{t+\Delta t}, \boldsymbol{\theta}\right] = 0$, which in this case specialises to

$$g\left(\mathbf{w}_{t+\Delta t}, \boldsymbol{\theta}\right) = \begin{pmatrix} \left(y_{t+\Delta t} - y_t\right) - \left(\alpha + \beta y_t\right)\Delta t \\ \left(\left(y_{t+\Delta t} - y_t\right) - \left(\alpha + \beta y_t\right)\Delta t\right)^2 - \sigma^2 y_t^2 \Delta t \\ \left(\left(y_{t+\Delta t} - y_t\right) - \left(\alpha + \beta y_t\right)\Delta t\right) y_t \\ \left(\left(y_{t+\Delta t} - y_t\right) - \left(\alpha + \beta y_t\right)\Delta t\right)^2 y_t - \sigma^2 y_t^2 \Delta t y_t \end{pmatrix}, \tag{6}$$

where $\mathbf{w}_{t+\Delta t} = \left(y_{t-\Delta t} - y_t, y_t, y_t^2\right)^T$ and $\boldsymbol{\theta} = \left(\alpha, \beta, \sigma, \gamma\right)^T$. The moment conditions (6) can be coded in S+ as shown in Figure 2.

```
CIR.moments <- function(parm, data, dt)
{
e.hat <- as.vector(data[,1] - (parm[1] + parm[2] * data[,2])* dt)
m2    <- e.hat * as.vector(data[, 2])
m3    <- e.hat^2 - dt * parm[3] * parm[3] * (as.vector(data[, 2]))
m4    <- m3 * data[, 2]
cbind(e.hat, m2, m3, m4)
}
```

Figure 2. S+ code for the moment conditions of the square root diffusion model

Note that we have four moment conditions and three parameters to estimate, hence $\boldsymbol{\theta}$ is overidentified and thus the GMM objective function is given by $J\left(\boldsymbol{\theta}\right) = T\, g_T\left(\boldsymbol{\theta}\right)^{\mathrm{T}}\, \hat{\mathbf{W}} g_T\left(\boldsymbol{\theta}\right)$, where

$$g_T\left(\boldsymbol{\theta}\right) = \frac{1}{T}\sum_{t=1}^{T} g\left(\mathbf{w}_t, \boldsymbol{\theta}\right), \tag{7}$$

and $\hat{\mathbf{W}}$ is a 3 × 3 positive definite symmetric weight matrix such that $p\lim_{n\to\infty} \hat{\mathbf{W}} = \mathbf{W}$. The efficient nonlinear GMM estimate of $\boldsymbol{\theta}$ is then $\hat{\boldsymbol{\theta}}_{GMM} = \arg\min_{\boldsymbol{\theta}} J\left(\boldsymbol{\theta}, \hat{\mathbf{W}}\right)$, where $\hat{\mathbf{W}} = \hat{\mathbf{S}}^{-1}$. An autocorrelation consistent estimate of $\mathbf{S}$ can be obtained as

$$\mathbf{S} = \boldsymbol{\Gamma}_0 + \sum_{j=1}^{\infty}\left(\boldsymbol{\Gamma}_j + \boldsymbol{\Gamma}_j^T\right) \tag{8}$$

where

$$\boldsymbol{\Gamma}_j = \frac{1}{T}\sum_{t=j+1}^{T} g_t\left(\hat{\boldsymbol{\theta}}\right) \cdot g_{t-j}\left(\hat{\boldsymbol{\theta}}\right)^T \tag{9}$$

An S+ function that implements GMM estimation for the square root diffusion model (2) is given in Figure 3. It is based on the **S+FinMetrics** function **GMM()** of Zivot and Wang (2006). Note that the data argument **x** should be a **timeSeries** object with the relevant series $y_1, y_2, \ldots, y_T$ in the j^{th} column.

```
CCIR.gmm <- function(x, dt, start.vals=runif(3), j=1)
{
data.ts  <- seriesMerge(diff(x[, j]), tslag(x[, j]))
data.mat <- as.matrix(seriesData(data.ts))
gmm.mod  <- GMM(start.vals, CIR.moments, ts=T, data=data.mat,
            dt, control=nlregb.control(iter.max=1000, eval.max=1000),
            lower=rep(0,3))
return(t(as.matrix(gmm.mod$parameters)))
}
```

Figure 3. S+ code for GMM estimation of square root diffusion model coefficients

3 An example: Forecasting short-term interest rates

To illustrate the above procedure, consider forecasting a short-term interest rate. Using the 9-month Euribor interest rate daily time series we estimate the coefficients of the square root diffusion model (2) as α = 1e-005, β = 0.1109, σ = 0.1929 initially using 250 daily observations, starting at 30 Dec 1998 and finishing on 14 Dec 1999. Using the value of the 9-month rate on 14 Dec 1999 as the starting value, y_T = 3.634, we can simulate *H*-period ahead using the MC estimator (4).

With these coefficient estimates values, for H = 5 and N = 50, we obtain the forecast for 21 Dec 1999 of $\tilde{y}_T$ = 3.607, which can be compared to the actual value on that day of 3.719. Repeating this procedure by moving one day forward till, for example, 13 Oct 2006, we obtain the backtest results plotted in Figure 4. Note that GMM calibration should also be repeated for each 250-day window, moving one day forward at each step. We can observe that sharp rising of the Euribor rates in the first half of 2000 resulted in several months of too low forecasts, which changed in February 2001 hence afterwards the forecasts errors sharply decreased.

What is of particular interest at this point is the convergence speed of MC estimates. Namely, simple backtests such as the one showed in Figure 4 can be prohibitively

expensive if ran over a large time span or if needing a large number of MC runs (N). Figure 5 shows convergence in mean for the example of the 5 and 50 day forecasts of the 9-month Euribor rate starting from 30 Dec 1998. The comparison shows that QMC produces much smother and much faster convergence than MC. There is no apparent advance in using the higher order accuracy Milstein scheme over the Euler scheme.

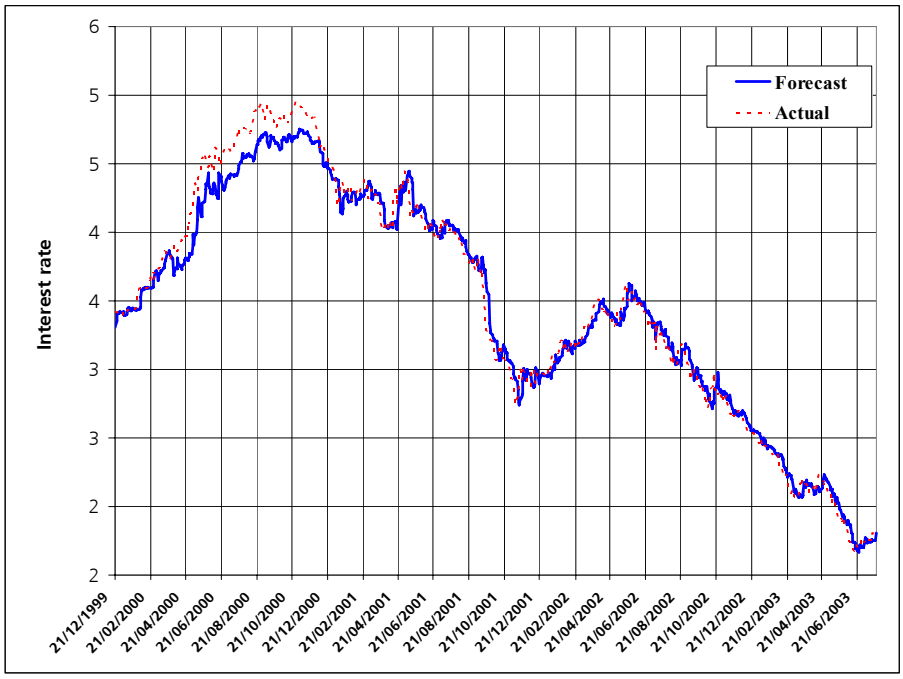

Figure 4. 5-days a-ahead backtest for the 9-month Euribor interest rate series

We compared MC with QMC for both Euler and Milstein discretisation. We found no advantage of using either of the two discretisation schemes; however, we obtained notably faster convergence using QMC compared to (pseudo) MC. This finding suggests we would need considerably less simulation runs if using QMC compared to (pseudo) MC to achieve convergence. In the next section we look further into the convergence issue by comparing the root mean square error (RMSE) convergence of QMC and MC estimators.

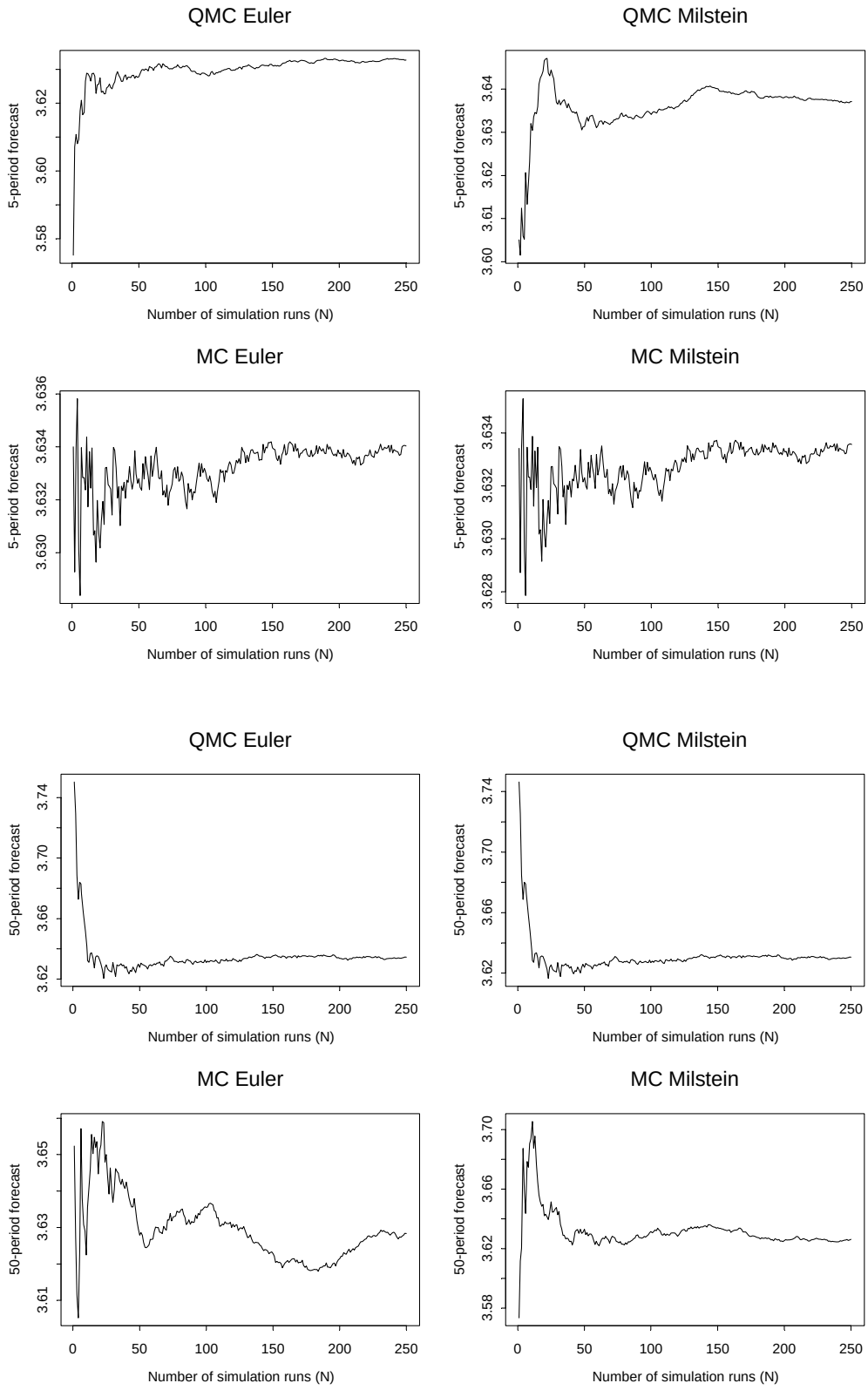

Figure 5. Convergence in mean of MC and QMC estimators with using Euler and Milstein schemes

4 RMSE convergence

We study the root mean square error (RMSE) convergence rates for short (H = 5), medium (H = 50), and long (H = 250) term forecasts starting at 14 Dec 1999, using α = 1e-005, β = 0.1109, σ = 0.1929 and y_0 = 3.634. We calculate the RMSE over K = 20 independent replications as

$$\varepsilon_{RMSE} = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(y(T)^{(k)} - \bar{y}(T)\right)^2} \quad . \tag{10}$$

Here $y(T)^{(k)}$ is the expected value of $y(T)$ at k^{th} run. $\bar{y}(T)$ is the corresponding converged value of $y(T)$. In our test $\bar{y}(T)$ was taken to be $E\left[\, y\left(T, N = 2^{20}\right)\right]$. For the MC method all runs were statistically independent. For QMC simulation for each run a different part of the Sobol' sequence was used.

RMSE can be approximated by a trend line: $cN^{-\alpha}$. For MC methods the convergence rate α is typically ~ 0.5 while for QMC methods is ~ 1.0. Values $(-\alpha)$ are given in brackets on the plots of convergence rates.

The RMSE convergence results for H = 5 indicates faster convergence rate of the QMC compared to MC algorithm (Figure 7), which is a similar result to the well known case of the geometric Brownian motion. For example, to achieve the same reduction in RMSE with 2^{10} replications of QMC it would take approximately 2^{17} replications with MC using pseudo random numbers, which translates into the reduction of CPU time of more than 100 times. Such convergence speed difference would be particularly noticeable if a MC simulation is used in backtesting.

Figure 6 outlines the S+ function we used to compute the simulations reported here. Note that this function uses the Sobol quasi random number generator **`sobolD()`**, which is an S+ wrapper for the SobolSeq370 generator distributed by Broda Ltd (Broda Ltd, 2007).

Figure 8 and Figure 9 show the RMSE convergence results for forecasting horizons of $H = 50$ and $H = 250$. We can observe similar convergence supremacy of QMC even in high dimensions.

```
MonteCarlo <-  function(alpha, beta, sigma, r0, dt, scheme="euler",
method="MC", N, ToReturn="last")
{
K.rep <- matrix(NA,13,20);
for(k in 1:20)
{
k.index <- (k-1)*((2^13-1)*2^6+1)-((k-1)-1)+cumsum(c(0,2^(6:17)));
N.trial <- rep(NA,13);
       if(method=="MC")
       {
       for(j in 6:18)
       {
       N.j.mean <- rep(NA,2^j);
       for(i in 0:(2^j-1))
       {
       N.j.mean[i+1] <- CIR.sim(alpha, beta, sigma, r0, dt, scheme,
                         method="MC", N, ToReturn, index=(k.index[j-5]+i));
              }
       N.trial[j-5] <- mean(N.j.mean);
       print(paste("N.trial = ",N.trial[j-5],"| K = ",k,"| j = ",j));
       }
       }
       if(method=="QMC")
       {
       for(j in 6:18)
       {
       Sim.s <- t(sobolD(k.index[j-5], 2^j, N))
       N.j.mean <- apply(Sim.s, 2, FUN=function(x){CIR.sim(alpha, beta,
                    sigma, r0, dt, scheme, method="QMC", N, ToReturn, z=x)});
       N.trial[j-5] <- mean(N.j.mean);
       print(paste("N.trial = ",N.trial[j-5],"| K = ",k,"| j = ",j));
       }
       }
K.rep[,k] <- (N.trial - N.trial[13])^2;
}
rm(alpha,beta,sigma,r0,dt,scheme,method,N,ToReturn);
return( sqrt(rowMeans(K.rep)) );
}
```

Figure 6. S+ code for computing MC RMSE

We also compare the RMSE convergence of the QMC estimator using Euler discretisation and the Brownian bridge algorithm. Figure 10 shows the results, which find no notable advantage of Brownian bridge over the Euler scheme. It is in contrast to the standard geometric diffusion model for which the Brownian bridge algorithm produces a much higher convergence rate comparing to the standard (Euler) discretisation scheme (Kucherenko et al, 2007).

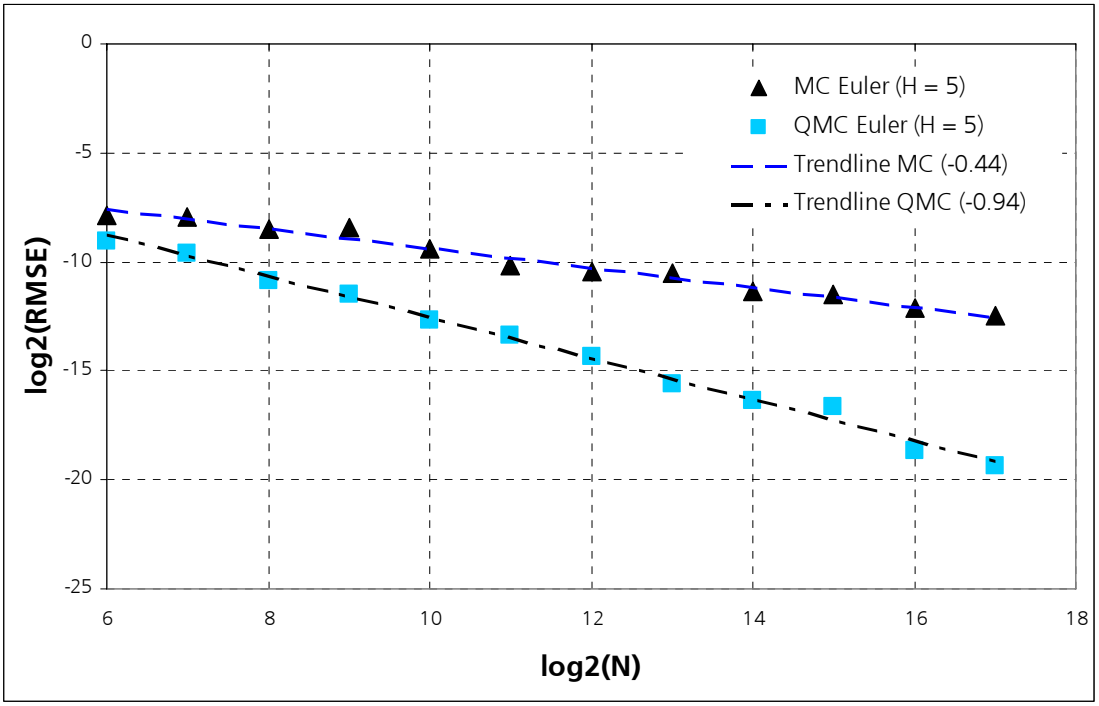

Figure 7. RMSE convergence for 5-days forecasting horizon

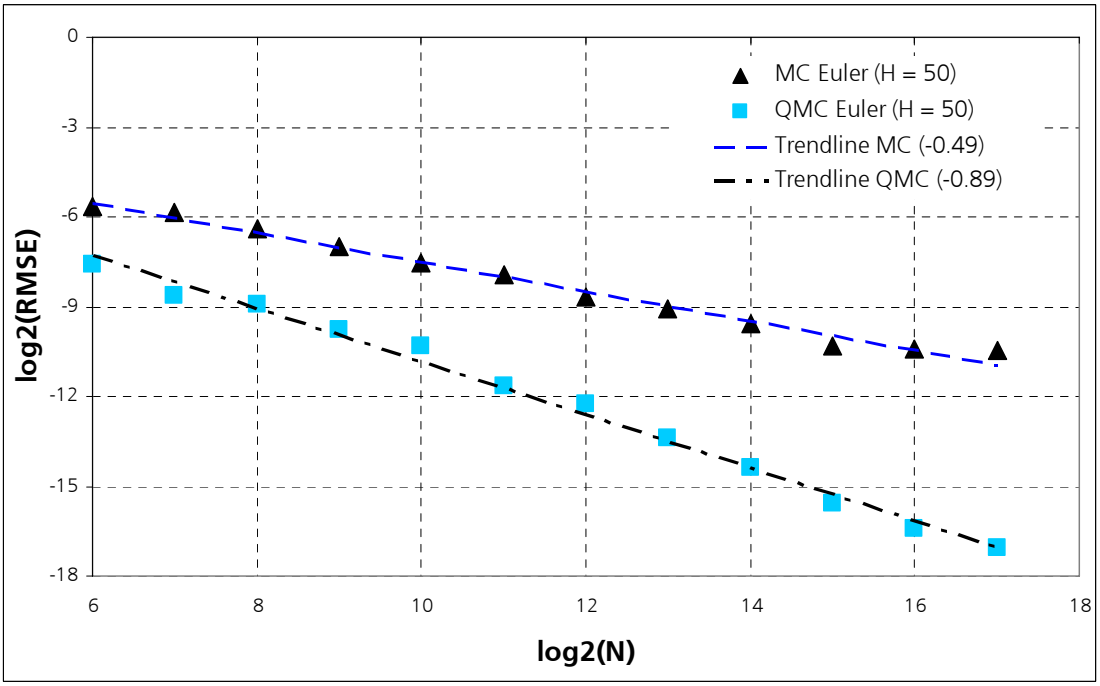

Figure 8. RMSE convergence for 50-days forecasting horizon

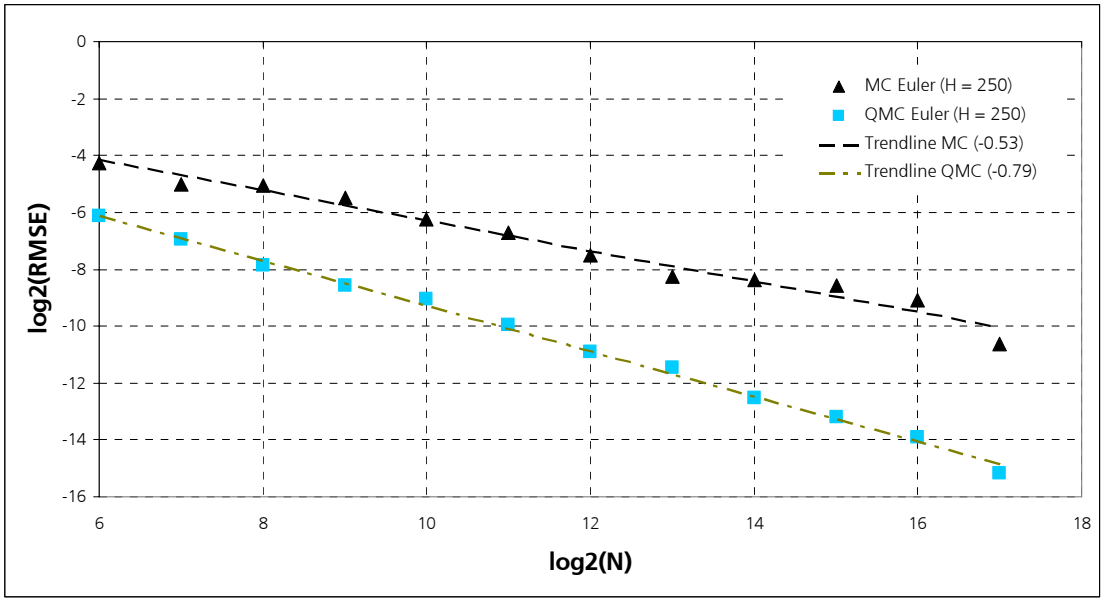

Figure 9. RMSE convergence for 250-days forecasting horizon

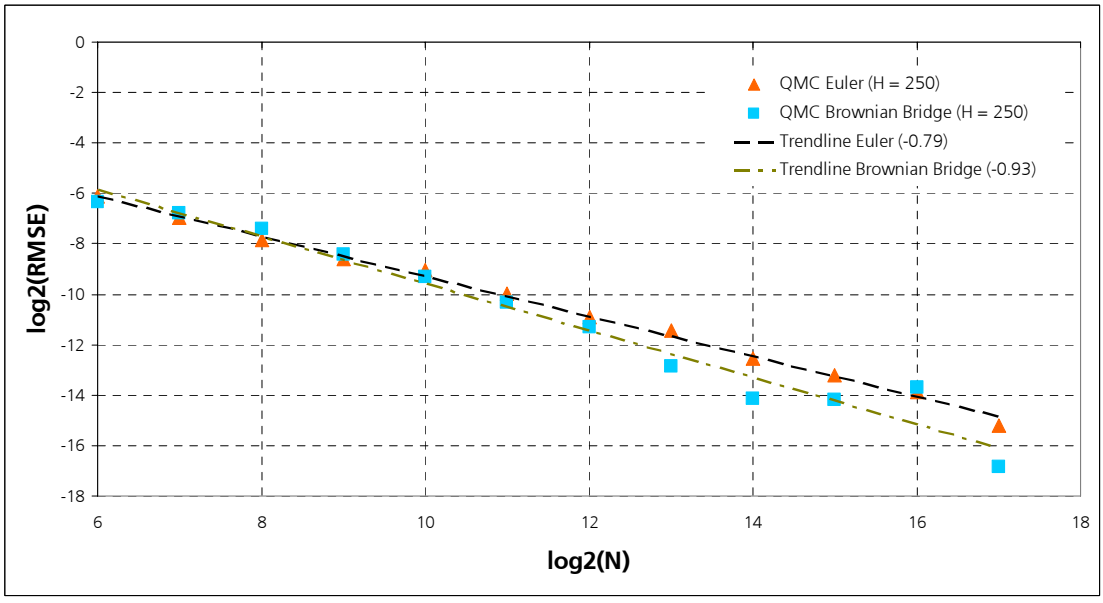

Figure 10. Brownian bridge compared to Euler scheme for 250-days horizon

4 Conclusion

We considered forecasting from the square root diffusion model, and outlined a MC simulation forecasting procedure. We compared empirical convergence of the MC and QMC estimates of the *H*-period forecasts for the square root diffusion model. We find that QMC estimator displays much faster convergence compared to MC

regardless of the forecasting horizon (dimension) and discretisation method used. In addition, we find that for the square root diffusion model the QMC outperforms (pseudo random) MC using Euler discretisation scheme, without the need to use the Brownian bridge algorithm.

References

Andersen, T. G., Benzoni, L., and Lund, J. (2002). An empirical investigation of continuous time equity return models. Journal of Finance 57, 1239–84.

Andersen, T.G. and Lund, J. (1997). "Estimating Continuous-Time Stochastic Volatility Models of the Short-Term Interest Rate. *Journal of Econometrics*, 77, 343—377.

Bacinello, A. R. (2000). Valuation of contingent-claims characterising particular pension schemes. Insurance: Mathematics and Economics 27, 177–88.

Barone-Adesi, G., and Sorwar, G. (2002). Interest rate barrier options. In Computational Methods in DecisioH-Making, Economics and Finance: Applied Optimization 74, E. J.

Bratley P, Fox B, Niederreiter H. (1992), Implementation and tests of low-discrepancy sequences, ACM Transactions on Modeling and Computer Simulation, **2**(3), 195—213.

BRODA Ltd. http://www.broda.co.uk

Cox, J. C., Ingersoll, J. E., and Ross, S. A. (1985). A theory of the term structure of interest rates. Econometrica 53, 385–407.

Gkamas, D. (2001). Stochastic volatility and option pricing. Quantitative Finance 1, 292–7.

Glasserman, P. (2004). Monte Carlo Methods in Financial Engineering. Springer, Berlin.

Hall, A. (2005), *Generalized Method of Moments*. Oxford University Press.

Higham, D.J. and Mao, X. (2005), Convergence of Monte Carlo simulations involving the meaH-reverting square root process. *Journal of Computational Finance*. 8(3), 35—61.

Jäckel, P. (2002), *Monte Carlo Methods in Finance*. John Wiley.

Kucherenko S., Shah N. (2007), The Importance of being Global.Application of Global Sensitivity Analysis in Monte Carlo option Pricing *Wilmott*.

Paskov S, Traub J. (1995), Faster evaluation of financial derivatives. J. Portfolio Management, **22**(1), 113—120.

Sobol' I. (1998), On quasi- Monte-Carlo integrations. Mathematics and Computers in Simulation, **47**, 103—112.

Zivot, E. and Wang, J. (2006), Modelling Financial Time Series with S-PLUS. Springer.