Imputation, Estimation and Missing Data in Finance

by
Giuseppe DiCesare

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosphy
in
Statistics

Waterloo, Ontario, Canada, 2006

I hereby declare that I am the sole author of this thesis.

This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Suppose X is a diffusion process, possibly multivariate, and suppose that there are various segments of the components of X that are missing. This happens, for example, if X is the price of various assets and these prices are only observed at specific discrete trading times. Imputation (or conditional simulation) of the missing pieces of the sample paths of X is discussed in several settings. When X is a Brownian motion the conditioned process is a tied down Brownian motion or a Brownian bridge process. In the special case of Gaussian stochastic processes the problem is simplified since the conditional finite dimensional distributions of the process are multivariate Normal. For more general diffusion processes, including those with jump components, an acceptance-rejection simulation algorithm is introduced which enables one to sample from the exact conditional distribution without appealing to approximate time step methods such as the popular Euler or Milstein schemes. The method is referred to as *pathwise imputation*. Its practical implementation relies only on the basic elements of simulation while its theoretical justification depends on the pathwise properties of stochastic processes and in particular Girsanov's theorem. The novelty of the method is that it allows for the complete characterization of the bridge paths of complicated diffusions using only Brownian bridge interpolation. The imputation methods discussed are applied to estimation, variance reduction and exotic option pricing.

Acknowledgments

I would like to thank my supervisor, Dr. Don McLeish, for his effort in making my time as a graduate student a fantastic and memorable experience. It has been a great pleasure and privilege to complete this work under his guidance. I also wish to thank my thesis committee: Dr. Phelim Boyle, Dr. Adam Kolkiewicz, Dr. Andrew Heunis and Dr. Paul Glasserman for their time and valuable comments.

My thanks go out to the Natural Science and Engineering Research Council of Canada (NSERC), the Ontario Graduate Scholarship Program (OGS) and the University of Waterloo for their financial support.

Dedication

This thesis is dedicated to my parents. My mother has been the greatest example of perseverance in my life which is the most important quality I needed to complete this work. Without my father's interest and persistence in developing my education at an early age I would not be where I am today. Thank you both for your unconditional love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Since Rubin's (1987) seminal work on missing data, imputation to create complete datasets has become a common statistical practice. The purpose of this thesis is not to discuss the validity of such procedures; it has been well documented in the statistical literature that imputation is a useful tool for analyzing missing data problems, so long as the imputed data are treated as such and not as observed values. We wish to make such techniques available to both multivariate and univariate diffusion models, such as those commonly used in finance, for parameter and error estimation, as well as other problems in which stochastic interpolation is useful.

Missing or incomplete data is present in nearly every use of financial data. Covariates, often relevant to the study at hand are not considered for reasons of parsimony. Specific financial time series which may be useful predictors are omitted from the analysis because they are either entirely missing or only partially observed. Continuous time models in finance, typically diffusions may (or may not) provide a reasonable model for the fluctuations in prices of a given asset but in any case

these prices, even if the market were essentially perfect, are only observed at specific times when trades are made. This may be of little consequence for highly liquid equities and benchmark bonds, but is a much more significant issue with thinly traded or illiquid assets.

Although most modern models for interest rates, bond yields, equity prices, etc. are continuous time multivariate models, these are the very models most susceptible to problems associated with asynchronous trading and missing data. For example if we obtain tick trade (or quote) data for a number of stocks in nearly continuous time there is virtually no chance that two stocks are traded synchronously. For example in Figure 1.1, there are a total of ten trades (labelled as "■") on 4 different stocks but no two stocks are traded at the same time. If we wished to complete this data, we would need the thirty missing prices labelled "•"). Any analysis based on complete data estimators would first require the imputation of all of these values. In such a setting the missing data are an artificial construction introduced to simplify the problem. Another example is a stochastic volatility model for stock prices. Such a model is often a bivariate diffusion with stochastic differential equation of the form

$$dS_t = \mu(S_t)\,dt + \sqrt{v_t}\sigma(S_t)\,dW_t^1$$
$$dv_t = \alpha(v_t)\,dt + \varphi(v_t)\,dW_t^2,$$

where W^1 and W^2 are Brownian motions with instantaneous correlation given by ρ. S_t and v_t represent the stock's value and volatility, respectively, at time t. The volatility process is never observed, it is a latent variable. For the purposes of estimation it can be useful to consider the volatility process as missing data. Eraker (2001) estimates stochastic volatility models under this premise and finds

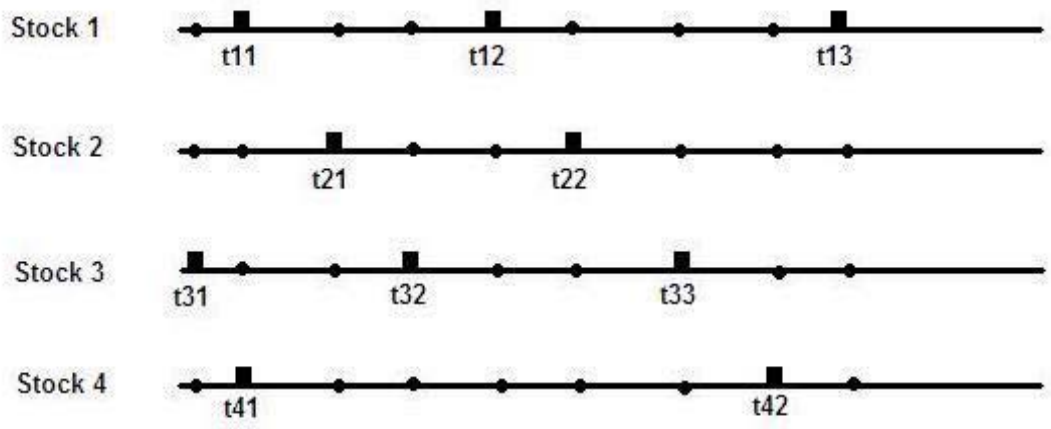

Figure 1.1: Discrete trading.

promising results. Further examples arise in the estimation and calibration of option pricing models. Stock and option quotes are often asynchronously observed which can serve to greatly complicate estimation procedures based on the joint observation of these quantities. Again, this can be viewed in missing data context.

There are a number of other applications of imputation methodology for financial time series. When we wish to price path-dependent options, in order to employ variance reduction techniques such as importance sampling and control variates. We are frequently faced with the task of simulating a diffusion process conditional on its value at endpoints, say the beginning and end of the life of the option. Because of the Markov property of diffusions, this can be done working "inwards" by simulating the value in the middle of an interval, conditional on the two endpoints. This is essentially the problem of imputation we deal with here.

Throughout the remainder of this thesis the term "imputation" will be used

synonymously with "conditional simulation". In the context of statistical estimation imputation refers to the simulation of missing data conditional on observed quantities. When discussing the simulation of diffusion processes (for other applications) imputation is taken to mean the stochastic interpolation of the process conditional on some given endpoint values.

1.1 Estimation and Incomplete Data

1.1.1 Simple Methods for Dealing with missing data.

Kofman and Sharpe (2003) provide a small survey of papers published in four recent volumes of five international journals in banking and finance, as summarized suggesting that missing observations is a common feature of many financial applications. In total, 175 articles (out of 1057) were identified where authors explicitly recognized their treatment of missing data, but since these were only cases where missing data was specifically mentioned, it likely underestimates the prevalence of missing data in these journals. They also give a summary of some of the most common methods of dealing with such data.

The *listwise deletion method* is perhaps most commonly used. It simply excludes any observations with missing data from the study and only uses complete records. This is often at considerable cost for multivariate data because there may be very few vectors without one or more components missing. The damage done is not restricted to the kind of efficiency loss expected with a reduction in the data. If the mechanism governing the "missingness" is related to the parameters of interest

or specific values of the dependent variable, then the exclusion of the missing data may lead to considerable systematic bias as well.

The *omitted variable* method is similar only it excludes the variables (rather than the cases) with missing data from the analysis. This may be possible for certain predictors in a regression, for example.

A third alternative is to impute, once or multiply, those values that are missing and while this is often done, the most common methods of imputation are largely subjective. For example one could randomly or subjectively assign values using an arbitrary guess at the distribution of the missing values (*ad hoc imputation*), use the mean of the values that were observed (*mean imputation*) or try to find observed values that match this missing ones in some respects and use these as proxies for the missing data (*proxy or hot deck imputation*).

1.1.2 Missing Data Mechanisms

It is difficult to devise a single principled way for dealing with missing data since the expediency of the above mentioned methods depend largely on how the data became missing in the first place. Was a data point merely not recorded because of human error? Was the data augmented just to form a complete data set, and the missing data simply invented for model flexibility and convenience? The key question is: what is the specific cause of the missingness? For example, in the estimation of stock price models with stochastic volatility, the volatility is an unobserved latent variable and hence never available for the estimation procedure at any time points. Perhaps the data is missing because market regulators have halted trading on a

particular stock since its price has reached a particular level. In another example the data could be the result of a designed experiment, say a response measuring the effect of a drug where some patients became ill and had to abandon the treatment leaving the experimenter with only partial data on such patients. It is important to consider the mechanism in which the data became missing before devising an imputation procedure. Such mechanisms have been studied in some generality. We give a brief description of each of them here. Suppose our data is $Y = (y_{ij})$, an $n \times p$ matrix with the columns of Y representing p variables. In many examples the rows of Y are considered to be independent observations on the vector of p variables, but we do not restrict ourselves to that case. In fact, we are concerned with stochastic processes, and hence the rows of Y will typically denote the time points at which our variables were observed. We assume that the distribution of Y depends on a vector of unknown parameters denoted by θ. In any event, one can consider another matrix random variable, $M = (m_{ij})$, the same size as Y , which determines which values of Y are observed. Such a random matrix is basically a collection of indicator variables with the property that y_{ij} is observed if $m_{ij} = 0$, and y_{ij} is missing whenever $m_{ij} = 1$. Y and M together determine a partition of the data into two components - an observed and a missing component, which we formally denote by $Y = (Y_{obs}, Y_{mis})$. Imputation refers to simulating the unobserved component of Y , Y_{mis}, to arrive at a complete data set which can then be analyzed using existing complete data methods. As described in Rubin (1987), the missing data mechanism is characterized by the conditional distribution of M given Y, say $f(M|Y, \phi)$ where ϕ is an unknown parameter. Data of this type falls into one of three possible classifications which is determined by properties of the conditional distribution of the missingness indicator:

- *Missing Completely at Random* (MCAR): $f(M|Y,\phi) = f(M|\phi)$
- *Missing at Random* (MAR): $f(M|Y,\phi) = f(M|Y_{obs},\phi)$
- Not Missing at Random (NMAR): $f(M|Y,\phi) = f(M|Y_{obs},Y_{mis},\phi)$

Note that these definitions are not restrictions on the pattern of missingness that may be present, they describe how the missingness depends on the values of the observed and missing data. Each classification has an impact on the potential of an estimation procedure to make valid inferences. When the missing data are MAR then the missing data mechanism is said to be ignorable for the purposes of likelihood estimation, and one can simply disregard the particular missing data mechanism by maximizing the likelihood:

$$L_{ign}(\theta|Y_{obs}) = \int f(Y_{obs},Y_{mis}|\theta)\,dY_{mis} = f(Y_{obs}|\theta)$$

whereas the full likelihood is defined to be

$$L_{full}(\theta,\phi|Y_{obs}) = \int f(Y_{obs},Y_{mis},M|\theta,\phi)\,dY_{mis} = f(Y_{obs},M|\theta,\phi)$$

Inferences regarding θ based on L_{ign} are equivalent to Maximum Likelihood estimation based on L_{full} when (i) the missing data are MAR, and (ii) θ and ϕ are distinct. For completeness we mention here that there is a further characterization of missing data, known as Coarsened at Random (CAR), which generalizes the ideas of MAR to a wider class of models exhibiting course data. For example, any type of rounded data is a special case of data coarsening, where the coarsening mechanism is non-stochastic. For a more complex example consider a one dimensional

diffusion observed at randomly spaced discrete time points, so that the underlying data is a sample path of the diffusion but the observable data is the sample path of a Markov jump process. For the remainder of this thesis all methods rely on assumption that the data are MAR and that θ and ϕ are distinct, so we may use L_{ign} for estimation purposes.

1.1.3 Multivariate Normal Data: Sweep Operator and EM Algorithm

We now describe the sweep operator, which is an algorithm involving a finite sequence of relatively simple steps which returns the conditional distribution of one set of multivariate normal random variables given another set. This permits easy imputation of missing values and the calculation of conditional expectations. A thorough summary of the sweep operator and its properties is given in Schafer (1997). Here we briefly summarize certain aspects of this useful tool. Suppose G is a $p \times p$ symmetric matrix whose $(i,j)th$ element is given by g_{ij}. For any $k \in \{1, ..., p\}$ the sweep operator on position k, $SWP[k]$, produces another $p \times p$ symmetric matrix H whose elements are given by

$$h_{kk} = -1/g_{kk}$$

$$h_{jk} = h_{kj} = g_{jk}/g_{kk} \text{ for } j \neq k$$

$$h_{jl} = h_{lj} = g_{jl} - g_{jk}g_{kl}/g_{kk} \text{ for } j \neq k \text{ and } l \neq k$$

One may notice that sweeping the matrix G in all its positions is equivalent to calculating $-G^{-1}$, ie.

$$SWP[1,...,p]G = SWP[1]...SWP[p]G = -G^{-1},$$

so long as none of the attempted sweeps involves a division by 0. Note that the order in which the variables are swept is not important since the sweep operator is commutative, $SWP[k_1]SWP[k_2]G = SWP[k_2]SWP[k_1]G$ for any $k_1 \neq k_2$, $k_1, k_2 \in \{1,...,p\}$. One may also define the reverse sweep operator on position k which is denoted by $H = RSW[k]G$ which is also given componentwise by

$$h_{kk} = -1/g_{kk}$$

$$h_{jk} = h_{kj} = -g_{jk}/g_{kk} \text{ for } j \neq k$$

$$h_{jl} = h_{lj} = g_{jl} - g_{jk}g_{kl}/g_{kk} \text{ for } j \neq k \text{ and } l \neq k.$$

It is worth mentioning that the reverse sweep is also commutative and, as it's name suggests,

$$RSW[k]SWP[k]G = G$$

for any $k \in \{1,...,p\}$.

The sweep operator can be used to turn response variables of a multivariate normal distribution into predictors. Suppose $z \sim MVN(\mu, \Sigma)$ is a p-dimensional random vector. Let $1 \leq p_1 < p$ and consider the partition of z into two random vectors, z_1 containing the first p_1 elements of z and z_2 containing the last $p - p_1$

elements of z, so that $z = (z_1, z_2)$. This induces a natural partition of μ and Σ

$$\mu = (\mu_1, \mu_2)$$
$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

where $E[z_i] = \mu_i$, $Cov[z_i] = \Sigma_{ii}$ and $Cov[z_1, z_2] = \Sigma_{12} = \Sigma_{21}'$. From basic results concerning the multivariate normal distribution, it is known that the conditional distribution of z_2 given z_1 is also normal with

$$E[z_2|z_1 = x] = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x - \mu_1) = \left(\mu_2 - \Sigma_{21}\Sigma_{11}^{-1}\mu_1\right) + \left(\Sigma_{21}\Sigma_{11}^{-1}\right)x$$

and

$$VAR[z_2|z_1 = x] = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$$

These results can be obtained using the sweep operator by cleverly arranging the parameters of the normal distribution into an alternate form, let

$$\theta = \begin{bmatrix} -1 & \mu \\ \mu' & \Sigma \end{bmatrix} = \begin{bmatrix} -1 & \mu_1 & \mu_2 \\ \mu_1' & \Sigma_{11} & \Sigma_{12} \\ \mu_2' & \Sigma_{21} & \Sigma_{22} \end{bmatrix}$$

Labelling the first position of θ as position 0, sweeping θ on positions $1, ..., p_1$ yields

$$SWP[0, ..., p_1]\theta = \begin{bmatrix} -1 - \mu_1\Sigma_{11}^{-1}\mu_1' & \mu_1\Sigma_{11}^{-1} & \mu_2 - \mu_1\Sigma_{11}^{-1}\Sigma_{12} \\ \Sigma_{11}^{-1}\mu_1' & -\Sigma_{11}^{-1} & \Sigma_{11}^{-1}\Sigma_{12} \\ \mu_2' - \Sigma_{21}\Sigma_{11}^{-1}\mu_1' & \Sigma_{21}\Sigma_{11}^{-1} & \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{bmatrix}$$

Notice that this swept matrix contains the parameters of the conditional distribution of z_2 given z_1 as given above. Also, the upper $(p_1+1)\times(p_1+1)$ matrix contains the parameters of the marginal distribution of z_1 in swept form. The significance of placing a -1 in the top left corner of θ is that the matrix θ can already be considered to be swept on position 0. A reverse sweep of θ on position 0 gives

$$RSW[0]\theta = \begin{bmatrix} 1 & \mu \\ \mu' & \Sigma + \mu'\mu \end{bmatrix},$$

the parameters of the unconditional multivariate normal distribution expressed in terms of the first two moments. This unswept form of the matrix is useful for computing maximum likelihood estimates. Suppose that $Y = (y_{ij})$ is an $n \times p$ matrix of independent observations on z, the sufficient statistics for this model can be written as $T_1 = Y'1$ and $T_2 = Y'Y$ which can be arranged into a $(p+1)\times(p+1)$ matrix as

$$T = \begin{bmatrix} n & T_1' \\ T_1 & T_2 \end{bmatrix}$$

Maximum likelihood estimation of this model then involves solving the moment equations: $RSW[0]\theta = n^{-1}T$ for θ, which is easily achieved by sweeping this equation on position 0 yielding $\widehat{\theta} = SWP[0]n^{-1}T$.

These results can be used for an economical implementation of the Expectation-Maximization (EM) algorithm which was first introduced in general by Dempster et. al. (1977). Suppose that only some of the data in Y are observed according to the missingness indicator $M = (m_{ij})$, which is a random matrix the same size as Y, where $m_{ij} = 0$ if y_{ij} is observed and $m_{ij} = 1$ otherwise. Y can be decomposed into its observed and missing parts: $Y = (Y_{obs}, Y_{mis})$. In general, there is no closed form

solution for the maximum likelihood estimates in the presence of missing data. In such circumstances the iterative EM algorithm is often the method of choice for solving likelihood based problems. After choosing an initial estimate $\theta^{(0)}$ for the parameter θ a single iteration of the EM algorithm is comprised of two steps. The first step consists of taking the conditional expectation of T over the distribution of $Y_{mis}|Y_{obs}$, the second step maximizes this result over the parameter θ. Using the sweep operator these two steps can be summarized by the equation

$$\theta^{(t+1)} = SWP[0]n^{-1}E\left[T|Y_{obs}, \theta^{(t)}\right]$$

where $\theta^{(t)}$ and $\theta^{(t+1)}$ denote successive parameter estimates. The above is essentially a linear regression of Y_{mis} on Y_{obs} whose calculation amounts to knowing the parameters of the distribution of $Y_{mis}|Y_{obs}$ which can be easily calculated using the sweep operator. One may ask why use the sweep operator when linear regression is an easy exercise? For higher dimensional normal data the sweep operator is more computationally efficient then running several separate regressions. Its main advantage is due to the fact that with one sweep of the parameter matrix a response variable can be converted to a predictor and vice versa.

We now review specific details on how to calculate $E[T|Y_{obs}, \theta]$ via sweep operator. The elements of T are of the form

$$\begin{aligned}
&\sum_{i=1}^{n} Y_{ij}, \ j = 1, ..., p \\
&\sum_{i=1}^{n} Y_{ij}^2, \ j = 1, ..., p \\
&\sum_{i=1}^{n} Y_{ij}Y_{ik}, \ j = 1, ..., p \ \ k = j+1, ..., p
\end{aligned}$$

Thus, calculating $E\left[T|Y_{obs},\theta\right]$ amounts to computing expectations of the form $E\left[Y_{ij}|Y_{obs},\theta\right]$, $E\left[Y_{ij}^2|Y_{obs},\theta\right]$ and $E\left[Y_{ij}Y_{ik}|Y_{obs},\theta\right]$, the random variables inside the expectation all belong to the same row of the data. Consider one row in Y, $Y_\alpha = (Y_{\alpha 1,...,}Y_{\alpha p})$, which contains at least one missing value. Suppose that the observed components of Y_α correspond to positions $1 \leq i_1, ..., i_s \leq p$, so that $M_{\alpha i_q} = 0$ for $q = 1, ..., s$. The conditional distribution $Y_{\alpha(mis)}|Y_{obs}$ which due to independence is equivalent to $Y_{\alpha(mis)}|Y_{\alpha(obs)}$ can be obtained by sweeping

$$\theta = \begin{bmatrix} -1 & \mu \\ \mu' & \Sigma \end{bmatrix}$$

on the corresponding observed components of Y_α, $i_1, ..., i_s$ (adopting the convention that the first row of θ is considered to be the 0th zero). Let $A = SWP[i_1, ..., i_k]\theta$ and denote the elements of A by a_{ij} $i, j = 0, ..., p$, then we have that

$$E\left[Y_{\alpha j}|Y_{obs}\right] = a_{oj} + \sum_{k\in\{i_1,...,i_s\}} a_{kj}Y_{\alpha k} \text{ for all } j \notin \{i_1, ..., i_s\}$$

$$Cov\left[Y_{\alpha i}, Y_{\alpha j}|Y_{obs}\right] = a_{ij} \text{ for all } i, j \notin \{i_1, ..., i_s\}$$

The observed components of Y_α can be regarded as fixed values over the distribution of $Y_{\alpha(mis)}|Y_{obs}$ and so we have that

$$E\left[Y_{\alpha j}|Y_{obs}\right] = Y_{\alpha j} \text{ for all } j \in \{i_1, ..., i_s\}$$

$$Cov\left[Y_{\alpha i}, Y_{\alpha j}|Y_{obs}\right] = 0 \text{ if } j \in \{i_1, ..., i_s\} \text{ for any } i$$

Using these results along with the known fact that

$$E\left[Y_{\alpha i}Y_{\alpha j}|Y_{obs}\right] = Cov\left[Y_{\alpha i}, Y_{\alpha j}|Y_{obs}\right] + E\left[Y_{\alpha i}|Y_{obs}\right]E\left[Y_{\alpha j}|Y_{obs}\right],$$

$E\left[T|Y_{obs}\right]$ becomes a routine calculation.

There are several methods for employing such ideas for a full scale implementation of the EM algorithm. Schafer (1997) shows one such implementation which organizes the data Y according to the missingness pattern exhibited by each row thereby minimizing the number of sweep operations required. However, this implementation is only useful when the rows of Y are independent random vectors, but can be generalized for data arising from a stochastic process as will be discussed in chapter 2 for Brownian motion.

1.2 Conditional Simulation of Diffusions: Estimation and Variance Reduction

Many estimation problems involving missing data for diffusion processes would be greatly simplified if one could obtain joint and conditional distributions of the observed and missing data. These conditional distributions are usually unknown and the current best practice is the use of Euler or Milstein approximations although it will be seen that these approximations can often add additional complexity. These approximations are more conducive to simulation than they are to calculating conditional explicitly conditional distributions such as is required by the EM algorithm. Perhaps for this reason simulation based estimation schemes have gained much popularity, especially but not exclusively in the Bayesian setting. These estimation methods rely on the imputation of the missing segments of the process. We review the popular approach to imputation for diffusion processes and point out serious drawbacks which arise in the context of estimation and simulation in general.

Suppose we are given a diffusion X with stochastic differential $dX_t = \mu(X_t)dt + \sigma(X_t)dW_t$ where W is a standard Brownian motion and X_0 is known. Consider the problem simulating $X_s|X_t, X_T$ for some $t < s < T$. In the few cases where the joint distribution of (X_t, X_s, X_T) is explicitly known this is usually not a difficult problem. For example, Gaussian processes such as Brownian motion or the Ornstein-Uhlenbeck process have well known joint and conditional distributions. In the general case where the transition density is not known there is no widely accepted method for tackling the problem. The natural approach taken by many

authors is to use an Euler approximation of the process which is of the form

$$X_\Delta | X_0 \sim N\left(X_0 + \Delta\mu(X_0), \Delta\sigma(X_0)'\sigma(X_0)\right).$$

This is only practical for small time steps $\Delta \approx 0$. Of course, $t < s < T$ can be times which are arbitrarily far from each other and taking Δ to be $s-t$ or $T-s$ will generally lead to a very poor approximation. To circumvent this problem while still retaining the simplicity of the Euler approximation many authors propose to introduce auxiliary times between the time points of interest. For example, we could introduce m (resp. n) time points between t and s (resp. between s and T), say $t < s_1 < s_2 < ... < s_m < s < s_{m+1} < s_{m+2} < ... < s_{m+n} < T$ chosen close enough to each other to justify an Euler approximation. This leaves us with simulating $X_{s_1}, ..., X_{s_m}, X_s, X_{s_{m+1}}, ..., X_{s_{m+n}} | X_t, X_T$. This can significantly increase the dimensionality of the problem with the added advantage of using the Euler scheme on a finer and hence a more appropriate discretized time scale. Though the increments of this discretized approximate process are normally distributed the conditional distribution need not be, and is only such in very special cases. For this reason, it is not clear how one should proceed at this point. Elerian et. al. (2001) and Eraker (2001) suggest MCMC methods which block the data according to time points. Due to the Markov property of diffusions this leaves one with simulating variates of the form $X_{s_i} | X_{s_{i-1}}, X_{s_{i+1}}$ based on the Euler approximation that $X_{s_i} - X_{s_{i-1}}$ and $X_{s_{i+1}} - X_{s_i}$ are normally distributed. In the case that the time points are equally spaced with a time spacing of $\Delta = s_i - s_{i-1} = s_{i+1} - s_i$ Eraker (2001) shows that the density of this conditional random variable is defined by the proportionality relationship

$$f\left(x_i | x_{i-1}, x_{i+1}\right) \propto$$

$$\left|\sigma\left(x_i\right)'\sigma\left(x_i\right)\right|^{-\frac{1}{2}}\exp\begin{array}{c}\left\{-\frac{1}{2}\left\|\left(\Delta x_i-\mu\left(x_{i-1}\right)\Delta\right)\sigma\left(x_{i-1}\right)^{-1}\Delta^{-\frac{1}{2}}\right\|^2\right.\\ \left.-\frac{1}{2}\left\|\left(\Delta x_{i+1}-\mu\left(x_i\right)\Delta\right)\sigma\left(x_i\right)^{-\frac{1}{2}}\Delta^{-1}\right\|^2\right\}\end{array},$$

where we use the following labels for convenience $x_i = x_{s_i}$ and $\Delta x_i = x_i - x_{i-1}$. This density is non-normal and can in fact be bimodal for large Δ. For small Δ he shows how one can simulate from this density using a hybrid accept/reject Metropolis-Hastings algorithm with a normal candidate density having mean $\frac{1}{2}\left(x_{i-1}+x_{i+1}\right)$ and variance $\frac{1}{2}\Delta\sigma\left(x_{i-1}\right)'\sigma\left(x_{i-1}\right)$. Elerian et. al. (2001) describes a similar scheme and both authors apply their methods in the context of Bayesian estimation involving missing data and latent variables. Such methods may not be the most convenient in practice. Since the Euler (or Milstein) approximation works better for smaller time steps one may be required to increase the number of augmented points. This comes at a cost. Not only is the dimensionality of problem increased in the sense that more auxiliary time points of the diffusion need to be simulated, but the resulting MCMC scheme will take longer to converge at the time points of interest. Also, monitoring convergence in such an algorithm requires additional diagnostic tests. It seems surprising that though the Euler approximation is simple and intuitive the simulation of conditional or bridge variates is not straight forward and one must resort to an iterative MCMC algorithm to produce a (non-exact) sample. There are a number of applications which require fast and accurate bridge sampling of a diffusion. In this section we will focus on two important problems: Estimation of diffusions and the use of stochastic interpolation in Monte Carlo simulation.

1.2.1 Bridge Sampling in Estimation Problems

There is an inherent problem in using the above simulation scheme for diffusion estimation problems. It depends on the fact that diffusion coefficient of a diffusion process can be estimated with exact precision if one had a continuous sample of the diffusion of any time length. Roberts and Stramer (2001) point out that data augmentation schemes such as those used by Eraker (2001) and Elerian et. al. (2001) break down as the number of augmented points gets arbitrarily large. To see this consider a simple diffusion of the form $dX_t = \mu(X_t)\,dt + \sigma dW_t$ where σ is unknown. If we could observe a continuous sample of X over the interval $[0,t]$ then the diffusion coefficient satisfies

$$\sigma^2 t = \lim_{n\to\infty} \sum_{i=0}^{n} \left(X_{\frac{i}{n}t} - X_{\frac{i-1}{n}t} \right)^2 .$$

Recall that the right hand side of this equation is the quadratic variation of the diffusion X over the interval $[0,t]$. Suppose we observe $X_t = x$ and consider an estimation scheme which iteratively updates the missing path of X up to time t. The imputed path determines the diffusion coefficient for the next imputation. Under such a scheme the estimated value of σ will be completely determined by the quadratic variation of the imputed path which is in turn completely determined by the previous value of σ. It is not hard to see that such a method produces a reducible Markov chain since the σ value used for the imputation will be the σ value inferred from the resulting path. Of course, in practice the path will be imputed at a discrete number of time points and so it seems that the reducibility of the estimation scheme should not be a problem so long as the number of augmented points in the path is chosen carefully. In fact, the larger the number of imputed time points the slower the convergence that one can expect where the limiting case of

continuous path imputation produces a reducible estimation scheme and hence no sensible convergence at all (Roberts and Stramer (2001)). This makes sense from the perspective that augmenting a dataset with "too much" missing data decreases the relative amount of information contained in the actual observed values. It is well known that missing data algorithms such as the EM converge more slowly as the proportion of incomplete data increases. A remedy to this is to decrease the number of imputed time points, however, this interferes directly with the accuracy of the Euler approximation which requires small time steps. Thus, there is a direct conflict between the accuracy required for the Euler approximation and the convergence of data augmentation schemes. For a more complete discussion of this relationship see Roberts and Stramer (2001). They study the properties of such algorithms in detail and propose a transformation of the missing data which eliminates the dependency of the imputation and the diffusion coefficient. This is one possible solution to such problems. Another solution is to produce exact simulations of the diffusion process. If one could produce exact imputations then the dependency between the data augmentation scheme and the convergence time of the algorithm could in theory be reduced or eliminated completely since unnecessary data augmentation can be avoided. Beskos et. al. (2005) propose an exact simulation algorithm for a class of one dimensional diffusions and use this to develop estimators of the transition density. They test their results in a Bayesian setting and find promising results.

1.2.2 Bridge Sampling and Variance Reduction

Another application of conditional simulation is in implementing variance reduction techniques in the context of Monte Carlo simulation. The most basic example of this in the finance literature is in the pricing of equity options. Consider a call option with strike price K and maturity time T on a stock whose time t value is given by S_t, $t \geq 0$. This financial contract gives its holder the right to buy a unit of the stock at time T at a price of K. Thus, the option will be exercised if the time T value of the stock is greater than K and so the payoff of this contract at maturity is given by $(S_T - K)^+ = \max[S_T - K, 0]$. The Black-Scholes model assumes that (in a risk-neutral world) the stock price process evolves according to a Geometric Brownian motion which solves the stochastic differential equation $dS_t = S_t(rdt + \sigma dW_t)$ where r is the riskless rate of interest and σ is the stock's volatility. From the theory of finance, it well known that the time t option price can be written as the expectation $C(r, \sigma, S_0, K, T-t) = E\left[e^{-r(T-t)}(S_T - K)^+\right]$. Under the assumed stock dynamics this expectation can be solved analytically, see Black and Scholes (1973) or Merton (1973). For some forms of the risk neutral stock dynamics this expectation in not tractable. Monte Carlo simulation is a common tool in analyzing such alternative dynamics which do not emit closed form solutions for option prices and other derivative securities. Boyle (1977) introduced the financial community to Monte Carlo methods in the context of the Black-Scholes model. This initial paper has sparked much interest in Monte Carlo methods for finance see Glasserman (2000) or McLeish (2005) for a summary of popular applications. A simulation based estimator of the call price is given by simulating N terminal (time T) values of the stock price S_{T_i} $i = 1, ..., n$ and forming the

average $\widehat{C} = e^{-r(T-t)} \sum_{i=1}^{N} (S_{T_i} - K)^+$. Some of these simulated terminal values will fall below K producing payoffs which take on a value of 0 contributing nothing to the sum $\widehat{C}$. These are wasted simulations. To avoid this computational waste one can simulate paths which are guaranteed to produce only nonzero payoff values. This is done by simulating terminal values which are guaranteed to fall above K and adjusting the estimator $\widehat{C}$ so that it remains unbiased. This is a form of stratified sampling. For this simple example the payoff depends only on the terminal value of the stock. Many other derivatives are path dependent in the sense that the payoff function depends several values of the asset price over the life of the contract. Simple examples of such contracts are barrier and lookback options whose payoff depends on the maximum (or minimum) value of the asset over a certain time period. Another example is an Asian style option whose payoff depends on the average value of the stock over subset of dates during the options life. Stratifying at the time points which enter into the payoff function of the derivative produce tends to produce optimal results. To employ the technique of stratified sampling for path dependent options one is often faced with the task of simulating the terminal value of a stock in some region and subsequently bridging the initial value to this terminal value at the time points of interest which enter into the payoff function. Thus, one needs to perform a conditional simulation experiment which can be viewed as a pathwise imputation.

In the case of geometric Brownian motion the solution to the stochastic differential equation is given by the process $S_t = S_0 \exp\left\{\left(r - \frac{\sigma^2}{2}\right)t + \sigma W_t\right\}, t \geq 0$. Since this is a one to one function of the Brownian motion W_t conditional simulation of geometric Brownian motion can be converted into a problem of conditional sim-

ulation of Brownian motion. The bridging process corresponding to a Brownian motion is the well known Brownian bridge process which can be easily simulated in terms of a standard Brownian motion. As a result, stratified sampling in the Black-Scholes framework can be carried out by simulating a Brownian bridge which is a relatively easy exercise.

Since the introduction of the Black-Scholes framework departures from the geometric Brownian motion stock model have been an active area of research. Also, other financial quantities are known to follow a certain behavioral pattern which is not implied by a geometric Brownian motion. For example, a common requirement of interest rate models is that they incorporate the possibility of mean reversion which is observed empirically. Though many alternative models are more realistic they often do not provide explicit solutions to most derivative prices. Numerical methods are often required to solve problems in the more complex scenarios. Monte Carlo simulation is one of the most widely used techniques although exact simulation or imputation in these models is not straight forward. Euler approximations are the simulation tool of choice in such settings, and due to the complexities mentioned above stratified sampling in these models is usually not feasible. The iterative MCMC techniques are computationally much more expensive than simulating a Brownian bridge and the convergence issues of such techniques can sometimes be problematic. One must study the efficiency of the resulting estimators all the while accounting for CPU time to ensure that the efficiency gains of stratifying outweigh computational costs. Many of these issues pertaining to MCMC are usually not well known nor understood by the non statistician and as such are relatively unused. Monte Carlo methods in finance are not restricted to one dimensional

problems but are also of interest in the multivariate setting. For example, several firm value default models are written in terms of diffusion processes whose hitting times to a certain barrier trigger a default or bankruptcy. Analyzing joint default probabilities in such models depends on the hitting time of a bivariate diffusion into a certain region. In such models one often employs importance sampling or stratified sampling to cause defaults more frequently leading to a more efficient estimates of joint default probabilities. Due to these concerns it is clear that simulating the bridging process for general diffusions in a more efficient yet simple way is of great practical importance.

1.3 Organization of this Thesis

This thesis is organized into six chapters. The results obtained throughout are mainly applications based, and the methods of experimentation used are based on new theory developed within. The original contributions are mainly are found in the middle three chapters. The main theme of this work is imputation or conditional simulation in the context of diffusion processes and its applications to estimation and pricing problems found in the financial and economic literature. Chapter 2 considers the estimation of an asynchronously observed multivariate Brownian motion. This problem is motivated, but not limited to, tick financial data which are trading data on financial securities that are recorded in continuous time. The asynchronous nature of the data allows us to approach the estimation as a missing data problem which is a popular topic in the statistical literature. This problem is attacked from every angle by employing Frequentist and Bayesian alternatives

in both deterministic and stochastic flavours. The main focus of Chapter 3 is the construction of an imputation technique for quite general diffusion processes. Conditional on the endpoints of a diffusion process an acceptance based sampling algorithm is developed which allows for the simulation of the path (or bridge) connecting the endpoints. The algorithm outputs a bridge path which is accepted on the basis of a finite number of points. Remarkably, upon acceptance the path can be filled in at arbitrary time points using only Brownian bridge interpolation. This method is extended to jump diffusion processes whose jump arrival intensity is not state dependent. Some applications involving the CIR square root diffusion process are also discussed. Chapter 4 is dedicated to applications of the pathwise imputation procedure developed in Chapter 3. The nature of the simulation technique allows for efficient generation of some functionals of a diffusion's path. Applications to barrier crossings, hitting times and extremes are discussed. These methods are then combined to price path dependent options in the CEV model with great success. A new model is introduced by adding a jump component to the traditional CEV process and pricing under this framework is shown to be easily accommodated by the same methods. Chapter 5 provides some closing remarks and Chapter 6 is an appendix of auxiliary results.

Chapter 2

Incomplete Data Methods for Brownian Motion

There is a class of models that is particularly well-suited to the treatment of missing or incomplete data, these are the multivariate processes which are transformations of a Gaussian process. This includes multivariate Brownian motion and Geometric Brownian motion, or either of these under a transformation of time (sometimes referred to as a subordinated Brownian motion), as well as common stationary parameter μ and diffusion matrix Σ. Arrange these observations into a matrix of data $X = (x_{ij})$, where $x_{ij} = X_i(t_j)$. Let $M = (m_{ij})$ be the missingness indicator of X, so that x_{ij} is observed whenever $m_{ij} = 0$ and is missing if $m_{ij} = 1$. Together X and M create a partition of the data into its observed and missing parts: $X = (X_{obs}, X_{mis})$. Directly maximizing the likelihood function for a general pattern of missingness under this framework is an impractical exercise. However, the EM algorithm is a widely accepted and proven alternative to finding estimates in this

framework in some circumstances. However, there are many cases which are not well suited to such a solution. For high dimensional data, such as asynchronously observed multivariate tick data, we will see that implementing the EM algorithm can be a computationally daunting task. There are a number of alternatives to the EM algorithm in analyzing such datasets which rely on imputation based techniques, such as single or multiple imputation. We provide yet another alternative based on imputation, a frequentist analogue of the popular Impute-Posterior (IP) algorithm, which can be considered to be a stochastic version of the EM algorithm.

2.1 High Dimensional Problems

The Brownian motion process X can be expressed with a stochastic differential equation

$$dX(t) = \mu dt + \Sigma^{1/2} dW(t)$$

for a standard $p-$dimensional Brownian motion process W, where $\Sigma^{1/2}$ denotes a matrix square root of Σ. If we had complete observations on every component of $X(t)$ at times $t_0 < t_1 < t_2 < ..., t_n$ then the maximum likelihood estimators of μ and Σ are easily determined,

$$\widehat{\mu} = \frac{1}{n} \sum_{i=1}^{n} \frac{X(t_i) - X(t_{i-1})}{t_i - t_{i-1}} \tag{2.1}$$

and

$$\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} \frac{[X(t_i) - X(t_{i-1}) - \widehat{\mu}(t_i - t_{i-1})]^T [X(t_i) - X(t_{i-1}) - \widehat{\mu}(t_i - t_{i-1})]}{t_i - t_{i-1}} \tag{2.2}$$

In the presence of a (possibly large) number of missing observations, these estimators can not be used. In theory, of course, since the joint distribution of all of the data (observed and unobserved) is multivariate normal, we could obtain the parameters of the conditional distribution of the unobserved given the observed data. These parameters could then be used to multiply impute the missing data or calculate the required conditional expectations for an implementation of the EM algorithm. The drawback to this technique is when we try and apply it to data X which has little or no completely observed time points, the distribution of a missing value can depend on the joint multivariate distribution of the entire observed dataset. For example, if we follow ten correlated stocks, and each are traded a total of 1000 times, then the chances that any two are traded at the same time is essentially zero. This means that at each of these 1000 trading times, we need to impute values of nine of the ten stocks. The multivariate normal distribution modelling these stock prices at 1000 times is of dimension 10000 and its covariance matrix of dimension $10^4 \times 10^4$, too large for convenient computer matrix operations or storage.

Although the continuous time process is Markovian, the observed data has no such property. For example, it is tempting to assume that, if we wish to impute the values of stock 1 at time t, it is sufficient to use only the joint distribution of stocks $1, 2, ..., p$ at those trading times that are nearest neighbours of t. Specifically, if we denote the trading times of stock i by $t_{i1} < t_{i2}, ...$ then for the imputation of $X_1(t)$ it is tempting to simulate from the conditional distribution of $X_1(t)$ given only the values $X_i(t_{ik_i}), X_i(t_{ik_i+1}), i = 1, 2, ...p$, where k_i is such that

$$t_{ik_i} < t < t_{ik_i+1}$$

are the nearest trading times before and after time t for stock i. A simple counter-example will convince that this is not the case. Consider, for example, two correlated standard Brownian motion process with observations as follows, where o denotes an observed value and * a missing value:

$$\begin{array}{lcccc} t = & 1 & 2 & 3 & 4 \\ X(t) = & * & \text{o} & * & \text{o} \\ Y(t) = & \text{o} & * & \text{o} & * \end{array}$$

Then with the data consisting of the vector

$$(X(2), X(4), Y(1), Y(3))$$

the covariance matrix of this data (assuming correlation ρ between the two Brownian motions) is

$$\begin{pmatrix} 2 & 2 & \rho & 2\rho \\ 2 & 4 & \rho & 3\rho \\ \rho & \rho & 1 & 1 \\ 2\rho & 3\rho & 1 & 3 \end{pmatrix}$$

Suppose we wish to impute the value of $X(3)$. Suppose we use only the nearest neighbours such as $Z = (X(2), X(4), Y(3))$ in constructing the imputation. Define

$$\begin{aligned} cov(X(3), Z) &= (cov(X(3), X(2)), cov(X(3), X(4)), cov(X(3), Y(3)) \\ &= (2, 3, 3\rho) = a, \text{ say} \end{aligned}$$

and the covariance matrix of Z is

$$A = \begin{pmatrix} 2 & 2 & 2\rho \\ 2 & 4 & 3\rho \\ 2\rho & 3\rho & 3 \end{pmatrix}$$

The conditional variance

$$\begin{aligned} var(X(3)|X(2), X(4), Y(3)) &= var(X(3)) - aAa' \\ &= 3 - aA^{-1}a' \end{aligned}$$

$$\begin{aligned} &3 - (2, 3, 3\rho) \begin{pmatrix} 2 & 2 & 2\rho \\ 2 & 4 & 3\rho \\ 2\rho & 3\rho & 3 \end{pmatrix}^{-1} \begin{pmatrix} 2 \\ 3 \\ 3\rho \end{pmatrix} \\ &= \frac{3\rho^2 - 3}{5\rho^2 - 6} \end{aligned}$$

On the other hand if we use all of the observations $(X(2), X(4), Y(1), Y(3))$ to impute $X(3)$, the conditional variance is

$$3 - \begin{pmatrix} 2 & 3 & \rho & 3\rho \end{pmatrix} \begin{pmatrix} 2 & 2 & \rho & 2\rho \\ 2 & 4 & \rho & 3\rho \\ \rho & \rho & 1 & 1 \\ 2\rho & 3\rho & 1 & 3 \end{pmatrix}^{-1} \begin{pmatrix} 2 \\ 3 \\ \rho \\ 3\rho \end{pmatrix} = \frac{\rho^4 - 5\rho^2 + 4}{\rho^4 - 8\rho^2 + 8}.$$

In Figure 2.1 we graph these two conditional variances against the value of ρ and although they are close for ρ near 0 or 1, there is evidently some loss of information in excluding $Y(1)$ as a predictor when $|\rho|$ is around $0.6 - 0.8$. We have considered only the conditional variance but the conditional mean is also dependent on which covariates we condition on, somewhat paradoxically in view of the Markovian nature of Brownian motion. Indeed this is generally true. If observations are at discrete and asynchronous times, although a bivariate (correlated) Brownian motion is

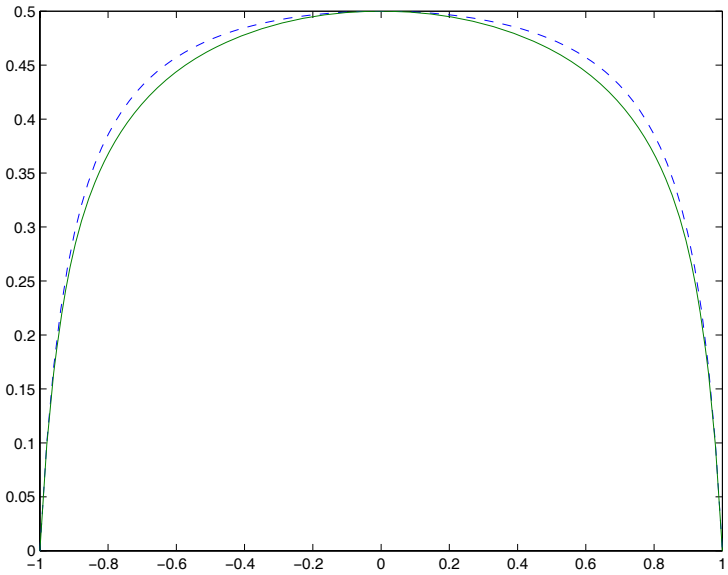

Figure 2.1: Comparison of conditional variances using the nearest neighbours (dashed line) and all covariate information (solid line).

Markovian, in order to impute the value of $X(t)$ we are unable to exclude any of the observations from the predictor set, no matter how far away they are from t without loss of some information. This means that we are required to treat the whole set of observed data as a giant multivariate normal vector, and for most datasets this is not feasible. Implementation of the EM algorithm in this scenario introduces the same type of difficulty: depending on the missingness pattern, calculation of the required conditional expectation could possibly depend on every observed value in the data. Using only the nearest neighbours to calculate such expectations provides an approximation at best, and can introduce bias and inefficiency. Fortunately, the Markov nature of this diffusion provides an alternative.

2.2 The EM algorithm for Multivariate Brownian Motion

Before we move on to imputation based techniques we derive an implementation of the EM algorithm for partially observed Brownian motion. The sweep operator can also be used to implement the EM algorithm for a process, but due to the issues mentioned in the previous section there are shortcomings to such an implementation. Consider a p - dimensional multivariate Brownian motion $X(t) = (X_1(t), ..., X_p(t))$ with drift and diffusion parameters given by $\theta = (\mu, \Sigma)$. Suppose we partially observe the process at times: $t_0, ..., t_N$. In other words, we observe at least one vector component of X at each of the given times and for simplicity we will take X_{t_0} to be observed. For $i = 0, .., N$ and $j = 1, ..., p$ let $X_{ij} = X_j(t_i)$ and define the missingness indicator random variables by $M_{ij} = 1$ if X_{ij} is missing and 0 if X_{ij} is observed. For i.i.d. multivariate Normal data Y recall the E and M steps of the EM algorithm can be expressed in one convenient calculation: $\theta_{t+1} = SWP[0]E[T|Y_{obs}, \theta_t]/n$, where T is the matrix of natural sufficient statistics for the Normal model and n is the number of observations. Also recall that the calculation of $E[T|Y_{obs}, \theta_t]$ was facilitated through the repeated use of the sweep operator for calculating conditional distributions. In constructing an EM algorithm implementation for a Brownian motion process with missing data we propose an analogous technique which employs the sweep operator.

For the moment assume that the time points $t_0, ..., t_n$ are equally spaced, and denote the common time increment as δ. This assumption is easy to relax and does not change much of what is discussed. In this setting the complete data arise from

an exponential family with the natural sufficient statistics similar to the multivariate Normal case. Due to the Markov property, conditional expectations involving functions of the missing data depend only on the observations which are inside the nearest completely observed rows of data. Thus, in determining the conditional expectation of the natural sufficient pertaining to the estimators of θ only subsets of the data need to be considered so long as there exists some completely observed rows in our data set. The observed rows of the data partitions the dataset into blocks. The distributional information of functionals from within each block are completely determined by the variables residing in the block. Once we have located the observed rows the sweep operator can be used to calculate the conditional distribution of the missing data within each block. If there are no completely observed rows no observations in the dataset can be ignored in calculating any expectations.

Given the data the matrix of natural sufficient statistics used in estimating $\theta = (\mu, \Sigma)$ is easily constructed by appealing to the independent increments property. Let $Y_{ij} = X_{ij} - X_{i-1,j}$. Then the random vectors $Y_{i*} = (Y_{i1}, ..., Y_{ip})$ $i = 1, ..., n$ are independent with $Y_{i*} \sim MVN(\delta\mu, \delta\Sigma)$ where $\delta = t_i - t_{i-1}$. Using this notation the natural sufficient statistic for the model is given as

$$T = \begin{pmatrix} \sum_{i=1}^{n} Y_{ij}, \ j = 1, ..., p \\ \sum_{i=1}^{n} Y_{ij}^2, \ j = 1, ..., p \\ \sum_{i=1}^{n} Y_{ij}Y_{ik}, \ j = 1, ..., p \quad k = j+1, ..., p \end{pmatrix}$$

Implementing the EM algorithm for this type of data amounts to calculating conditional expectations of the components of T with respect to the observed data. The algorithm is given as follows.

1. Identify the completely observed rows of X and label this set of rows as R. ie. $M_{r,j} = 0$ for all $r \in R$ and $j = 1, ..., p$. Augment the set R to include the first and last rows of the data whether they are observed or not, so that $R = \{r_1, ..., r_Q\}$ with $r_1 = 0$ and $r_Q = n$. This step essentially breaks the data up into blocks, where the first and last row of each block are completely observed (with possibly the exception of the first and last blocks). It also creates a natural partition of T into $Q-1$ components which can be summed to obtain T, ie. $T = \sum_{q=1}^{Q-1} T_q$, where

$$T_q = \begin{pmatrix} \sum_{i=r_q+1}^{r_{q+1}} Y_{ij}, \ j = 1, ..., p \\ \sum_{i=r_q+1}^{r_{q+1}} Y_{ij}^2, \ j = 1, ..., p \\ \sum_{i=r_q+1}^{r_{q+1}} Y_{ij}Y_{ik}, \ j = 1, ..., p, \quad k = j+1, ..., p \end{pmatrix}$$

Due to the Markovian nature of multivariate Brownian motion conditional expectations of functions of the missing data within each block depend only on observations that reside in the block. For each block $q = 1, ..., Q-1$, consider the random vector $X_q = \left(X_{r_q,1}, .., X_{r_{q+1},P}\right)'$ whose components are the stacked columns of the qth block, and its corresponding missingness indicator $M_q = \left(M_{r_q,1}, .., M_{r_{q+1},P}\right)'$. X_q has a multivariate normal distribution, denote its mean and covariance matrix by μ_q and Σ_q, respectively. To facilitate the use of the sweep operator arrange these parameters into a matrix

$$\theta_q = \begin{bmatrix} -1 & \mu_q' \\ \mu_q & \Sigma_q \end{bmatrix}$$

The conditional distribution of the missing observations given the observed

values in the qth block are easily calculated by sweeping θ_q on the corresponding observed data positions. Let

$$A_q = SWP\left[I_q\right]\theta_q$$

where $I_q = \left\{k : M_{q(k)} = 0,\ k = 1, ..., p(r_{q+1} - r_q + 1)\right\}$ is the set of positions which correspond to the observed variables in X_q (we label the first row of θ_q as the $0th$ row). The matrix A_q contains the parameters of the conditional distribution of the missing data given the observed for the qth block. Denote by a_{ij}^q the $(i,j)th$ entry of A_q.

2. To complete the EM algorithm implementation we need to calculate

$$E\left[T|X_{obs}, \mu, \Sigma\right].$$

This is achieved blockwise

$$E\left[T|X_{obs}, \mu, \Sigma\right] = E\left[\sum_{q=1}^{Q-1} T_q|X_{obs}, \mu, \Sigma\right] = \sum_{q=1}^{Q-1} E\left[T_q|X_{obs}, \mu, \Sigma\right]$$

For each pair of indices (i,j) such that $X_{ij} \notin X_{obs}$ let q_{ij} denote the block to which X_{ij} belongs and let k_{ij} be the corresponding position of X_{ij} in X_q. Then the following formulae completely determine $E\left[T_q|X_{obs}, \mu, \Sigma\right]$:

$$E\left[X_{ij}|X_{obs}\right] = a_{0,k_{ij}}^q + \sum_{k \in I_{q_{ij}}} a_{k_{ij},k}^q X_q(k),$$

for $X_{ij} \notin X_{obs}$, and

$$E\left[X_{ij}X_{ls}|X_{obs}\right] = a_{k_{ij},k_{ls}}^{q_{ij}} + E\left[X_{ij}|X_{obs}\right] E\left[X_{ls}|X_{obs}\right],$$

if $X_{ij}, X_{ls} \notin X_{obs}$ and $q_{ij} = q_{lk}$. Applying these results to the process increments leads to the trivial calculation of the conditional expected value of T. We have

$$E\left[Y_{ij}|X_{obs}\right] = E\left[X_{ij} - X_{i-1,j}|X_{obs}\right]$$
$$= \begin{cases} X_{ij} - X_{i-1,j}, \text{ if } X_{ij}, X_{i-1,j} \in X_{obs} \\ X_{ij} - E\left[X_{i-1,j}|X_{obs}\right], \text{ if } X_{ij} \in X_{obs} \text{ and } X_{i-1,j} \notin X_{obs} \\ E\left[X_{ij}|X_{obs}\right] - X_{i-1,j}, \text{ if } X_{ij} \notin X_{obs} \text{ and } X_{i-1,j} \in X_{obs} \\ \delta\mu_j, \text{ if } X_{ij}, X_{i-1,j} \notin X_{obs} \end{cases}$$

and

$$E\left[Y_{ij}Y_{ik}|X_{obs}\right] = E\left[\left(X_{ij} - X_{i-1,j}\right)\left(X_{ik} - X_{i-1,k}\right)|X_{obs}\right]$$

$$
= \begin{cases}
(X_{ij} - X_{i-1,j})(X_{ik} - X_{i-1,k}) & \text{if } A_1 \\
(X_{ik} - X_{i-1,k}) E[X_{ij}|X_{obs}] - X_{i-1,j}(X_{ik} - X_{i-1,k}) & \text{if } A_2 \\
(X_{ij} - X_{i-1,j}) E[X_{ik}|X_{obs}] - X_{i-1,k}(X_{ij} - X_{i-1,j}) & \text{if } A_3 \\
X_{ij}(X_{ik} - X_{i-1,k}) - (X_{ik} - X_{i-1,k}) E[X_{i-1,j}|X_{obs}] & \text{if } A_4 \\
(X_{ij} - X_{i-1,j}) X_{ik} - (X_{ij} - X_{i-1,j}) E[X_{i-1,k}|X_{obs}] & \text{if } A_5 \\
E[X_{ij}X_{ik}|X_{obs}] - X_{i-1,k}E[X_{ij}|X_{obs}] & \\
-X_{i-1,j}E[X_{ik}|X_{obs}] + X_{i-1,j}, X_{i-1,k} & \text{if } A_6 \\
X_{ij}X_{ik} - X_{ij}E[X_{i-1,k}|X_{obs}] & \\
-X_{ik}E[X_{i-1,j}|X_{obs}] + E[X_{i-1,j}X_{i-1,k}|X_{obs}] & \text{if } A_7 \\
X_{ij}E[X_{ik}|X_{obs}] - X_{ij}X_{i-1,k} & \\
-E[X_{i-1,j}X_{ik}|X_{obs}] + X_{i-1,k}E[X_{i-1,j}|X_{obs}] & \text{if } A_8 \\
X_{ik} - E[X_{ij}X_{i-1,k}|X_{obs}] - X_{i-1,j}X_{ik} + X_{i-1,j}E[X_{i-1,k}|X_{obs}] & \text{if } A_9 \\
(X_{ij} - X_{i-1,j}) E[X_{ik}|X_{obs}] - (X_{ij} - X_{i-1,j}) E[X_{i-1,k}|X_{obs}] & \text{if } A_{10} \\
(X_{ik} - X_{i-1,k}) E[X_{ij}|X_{obs}] - (X_{ik} - X_{i-1,k}) E[X_{i-1,j}|X_{obs}] & \text{if } A_{11} \\
X_{ij}(E[X_{ik}|X_{obs}] - E[X_{i-1,k}|X_{obs}]) & \\
-E[X_{i-1,j}X_{ik}|X_{obs}] + E[X_{i-1,j}X_{i-1,k}|X_{obs}] & \text{if } A_{12} \\
X_{ik}(E[X_{ij}|X_{obs}] - E[X_{i-1,j}|X_{obs}]) & \\
-E[X_{ij}X_{i-1,k}|X_{obs}] + E[X_{i-1,j}X_{i-1,k}|X_{obs}] & \text{if } A_{13} \\
E[X_{ij}X_{ik}|X_{obs}] - E[X_{ij}X_{i-1,k}|X_{obs}] & \\
-X_{i-1,j}(E[X_{ik}|X_{obs}] - E[X_{i-1,k}|X_{obs}]) & \text{if } A_{14} \\
E[X_{ij}X_{ik}|X_{obs}] - E[X_{i-1,j}X_{ik}|X_{obs}] & \\
-X_{i-1,k}(E[X_{ij}|X_{obs}] - E[X_{i-1j}|X_{obs}]) & if\ A_{15} \\
\delta\Sigma_{jk} + \delta^2\mu_j\mu_k & if\ A_{16}
\end{cases}
$$

where the conditions are as follows:

$$
\begin{aligned}
A_1 \quad & X_{ij}, X_{ik}, X_{i-1,j}, X_{i-1,k} \in X_{obs} \\
A_2 \quad & X_{ik}, X_{i-1,j}, X_{i-1,k} \in X_{obs} \text{ and } X_{ij} \notin X_{obs} \\
A_3 \quad & X_{ij}, X_{i-1,j}, X_{i-1,k} \in X_{obs} \text{ and } X_{ik} \notin X_{obs} \\
A_4 \quad & X_{ij}, X_{ik}, X_{i-1,k} \in X_{obs} \text{ and } X_{i-1,j} \notin X_{obs} \\
A_5 \quad & X_{ij}, X_{ik}, X_{i-1,j} \in X_{obs} \text{ and } X_{i-1,k} \notin X_{obs} \\
A_6 \quad & X_{i-1,j}, X_{i-1,k} \in X_{obs} \text{ and } X_{ik}, X_{ij} \notin X_{obs} \\
A_7 \quad & X_{ik}, X_{ij} \in X_{obs} \text{ and } X_{i-1,j}, X_{i-1,k} \notin X_{obs} \\
A_8 \quad & X_{i,j}, X_{i-1,k} \in X_{obs} \text{ and } X_{ik}, X_{i-1,j} \notin X_{obs} \\
A_9 \quad & X_{i-1,j}, X_{ik} \in X_{obs} \text{ and } X_{i-1,k}, X_{ij} \notin X_{obs} \\
A_{10} \quad & X_{ij}, X_{i-1,j} \in X_{obs} \text{ and } X_{ik}, X_{i-1,k} \notin X_{obs} \\
A_{11} \quad & X_{ik}, X_{i-1,k} \in X_{obs} \text{ and } X_{ij}, X_{i-1,j} \notin X_{obs} \\
A_{12} \quad & X_{ik}, X_{i-1,j}, X_{i-1,k} \notin X_{obs} \text{ and } X_{ij} \in X_{obs} \\
A_{13} \quad & X_{ij}, X_{i-1,j}, X_{i-1,k} \notin X_{obs} \text{ and } X_{ik} \in X_{obs} \\
A_{14} \quad & X_{ij}, X_{ik}, X_{i-1,k} \notin X_{obs} \text{ and } X_{i-1,j} \in X_{obs} \\
A_{15} \quad & X_{ij}, X_{ik}, X_{i-1,j} \notin X_{obs} \text{ and } X_{i-1,k} \in X_{obs} \\
A_{16} \quad & X_{ik}, X_{i-1,k}, X_{ij}, X_{i-1,j} \notin X_{obs}
\end{aligned}
$$

3. One iteration of the EM algorithm is given by: $\theta_{t+1} = (n\delta)^{-1} SWP[0] T^*$, where

$$
T^* = \begin{bmatrix} -1 & E[T_\mu | X_{obs}, \theta_t]' \\ E[T_\mu | X_{obs}, \theta_t] & E[T_\Sigma | X_{obs}, \theta_t] \end{bmatrix}
$$

and $T = (T_\mu, T_\Sigma)$ (the components of T corresponding to estimation of the mean and covariance).

This algorithm will be inefficient for continuous data which often has little or

no completely observed rows, as in the case of high frequency tick financial data, and so the distribution of missing components can depend on all observed variables. This setting involves sweeping an $np \times np$ sized matrix on all the observed variables in the data set, which is can computationally infeasible task in practice rendering the EM algorithm an impractical option.

2.3 Frequentist Imputation

The implementation of the EM algorithm is only feasible in some cases and recommended for smaller datasets. For large asynchronous datasets with no completely observed rows the covariance matrix of the data cannot be stored conveniently. For example, attempting to store the covariance matrix of a 5000 time point bivariate asynchronous dataset in MATLAB running on a Pentium 4 machine with a 3.00GHz processor and 500 MB of RAM leads to an out of memory error. Of course one could resort to storing such data in a file for access but this is often inconvenient and slow. As an alternative we introduce a stochastic algorithm for estimating a Brownian motion in the presence of missing data which yields the same solution as the EM algorithm. It involves producing multiple imputations of the missing component of the data. The ability to impute datasets is not only convenient for estimation purposes but is a powerful tool for error estimation. Imputation based methods can be used to supplement deterministic methods such as the EM algorithm. For example, calculating standard errors of estimators is often a tedious task in the context of the EM algorithm. This is often thought of as a drawback to the EM algorithm in complex problems. For such problems multiple

imputation provides a remedy. One could calculate point estimates using the EM algorithm and subsequently use imputation techniques and complete data variance estimators to get reliable estimates of standard errors in the presence of missing data. Throughout the remainder of this section we present a stochastic version of the EM algorithm for Brownian motion and discuss how one can use imputation to facilitate error analysis for general patterns of missing data and arbitrarily large datasets.

2.3.1 Impute-Solve Algorithm

Although the sweep operator provides an attractive procedure for the calculation of conditional multivariate normal distributions, it does not solve the high dimensionality problem discussed in the previous section. For time series data with asynchronous observations calculating the parameters of $X_{mis}|X_{obs}$ via the sweep operator requires sweeping an $(np+1) \times (np+1)$ matrix on every position corresponding to an observed value, which is far too computationally demanding. Again, one may use the nearest neighbour approach to approximate the EM solution, essentially regressing missing observations on their nearest neighbours only, although we find this to be an unstable alternative with little theoretical motivation. Instead, we suggest a convenient imputation-based solution. Appealing to the rich statistical theory of Markov Chain Monte Carlo (MCMC) methods, and the Gibbs sampler, we provide a solution to the high dimensionality problem. We can break up our imputation problem into several smaller ones, each of which requires sampling from a low dimensional multivariate normal distribution whose parameters are easily obtained through the use of the sweep operator. Similar ideas have been

explored by Eraker (2001) and Elerian et. al. (2001) under the Bayesian paradigm.

The plan is to impute the missing values X_{mis} yielding a completed data set from which we compute complete data maximum likelihood estimates. The updated parameter estimates are then used again to impute the missing data. Iterating in this fashion, yields a sequence of parameter estimates which has the desirable property that it shares the fixed points of the EM algorithm. We refer to this method as the *Impute Solve* (IS) algorithm, which is essentially a stochastic version of the EM approach. Many similar algorithms exists in the literature. The idea was first suggested by Celeux and Diebolt (1985) with applications to estimating mixture models. Other methods have appeared in the literature. Wei and Tanner (1990) use Monte Carlo integration to estimate the EM auxiliary function (Monte Carlo EM or MCEM), requiring a higher number of imputations at each iteration that do not contribute to the estimator in future steps of the algorithm. The asymptotic normal theory for a variety of such procedures has been characterized, by, for example, Robins and Wang (1998) and Nielsen (2000).

In this missing data problem we wish to maximize the observed data likelihood function under the assumption that the missingness mechanism is ignorable. The likelihood function of interest is the joint density of the observed data $f_o\left(X_{obs}|\theta\right)$, which is often a high dimensional object. The problem is finding a root of the observed data score function, which can be written in terms of the complete data score function as

$$S_o(\theta; X_{obs}) = E_\theta \left[S(\theta; X)|X_{obs}\right],$$

where

$$S(\theta; X) = \frac{\partial}{\partial\theta} \ln f(X|\theta)$$

(Fisher 1925) is the complete data score function and f is the likelihood in the complete data model. Due to the high dimensionality of the problem and the complications introduced by the presence of missing data, the function $S(\theta; X_{obs})$ is not available in closed form, making the above representation vital for implementation purposes. We propose a Robbins-Monro type stochastic estimation procedure for the root finding. A similar idea has been applied to a stochastic version of the EM algorithm known as SAEM (stochastic approximation EM, see Delyon, et al. (1999)) in the context of iteratively updating the EM auxiliary function, rather than estimating it outright by Monte Carlo at each iteration. Such methods are efficient because they do not waste any imputations, each contributes to the estimator with the earlier imputations having less weight as the algorithm proceeds. The method can be summarized as follows:

Impute - Solve (IS) Algorithm: Choose a sequence γ_k such that $\sum \gamma_k = \infty$ and $\sum \gamma_k^2 < \infty$ (for example $\alpha_k = k^{-1}$or $k^{-2/3}$). For an initial parameter guess of θ_0 the algorithm iterates according to the following procedure,

Step 1: Simulate the missing component of the data: $X_{mis}^{(k)} \sim X_{mis}|X_{obs}, \theta_k$.

Step 2: Calculate the Maximum Likelihood estimate α_k (e.g. $(\widehat{\mu}, \widehat{\Sigma})$) based on the completed dataset: $X = \left(X_{obs}, X_{mis}^{(k)}\right)$.

Step 3: Update parameter estimate: $\theta_{k+1} = \theta_k + \gamma_k (\alpha_k - \theta_k)$, and then return to step 2 or stop if convergence is determined.

This procedure is basically estimating the EM auxiliary function at each iteration using only one imputation, namely substituting the missing data score with the

completed data score and carrying forward with estimation. These types of stochastic EM algorithms tend to bounce around the "correct" parameter region rather quickly, more dispersion in the earlier iterations also helps keep the algorithm from getting stuck. For this fact alone, many authors advise beginning the deterministic version of the EM algorithm with its stochastic counterpart. We are aware that many suggestions have been made towards increasing the number of imputations at each iteration as the algorithm proceeds, such methods should be considered if convergence is a problem. For example, one could create $m_k \geq 1$ imputations at step 2 and obtain m_k parameter estimates which are then averaged to yield α_k. Accelerations methods which apply to MCEM can be employed in more generality here to speed up convergence. For models with tractable complete data estimators we recommend avoiding estimating the EM auxiliary function with more than one imputation at each iteration since this can often complicate the subsequent maximization step, paralyzing any computational advantage offered by explicit estimators. Delyon et. al. (1999) provide a detailed discussion of Robbins-Monro stochastic approximation procedures in the context of missing data estimation, also proving several convergence theorems. These methods can be used to prove the following result which establishes convergence of the IS algorithm to a root of $S(\theta; X_{obs})$.

THEOREM 1. *Under regularity conditions, if the distribution of the complete data X is in the exponential family, the IS algorithm with initial guess $\theta^{(0)}$ produces an almost surely convergent sequence $\left\{\theta^{(i)}\right\}$ with limit θ_*, where θ_* is a solution of $S(\theta; X_{obs}) = 0$.*

Proof: It is clear that we can write the IS algorithm in Robbins-Monro form

$$\theta_{k+1} = \theta_k + \gamma_k h(\theta_k) + \gamma_k \varepsilon_k$$

where $h(\theta_k) = E[\alpha_k - \theta_k | X_{obs}, \theta^{(k)}]$ and ε_k is random error obtained from the simulation with the property that $E[\varepsilon_k | X_{obs}, \theta_k] = 0$. It follows from standard results on convergence of the Robbins-Monro procedure such as Theorem 5 of Delyon et al. that θ_k converges almost surely to a root of $h(\theta) = 0$ or

$$E[\alpha_k - \theta | X_{obs}, \theta] = 0 \tag{2.3}$$

Within the linear exponential family, it is easy to see that $\alpha_k - \theta$ is a linear function of $S(\theta; X)$, the complete data score function and since the observed data score function $S(\theta; X_{obs}) = E_\theta [S(\theta; X) | X_{obs}]$, it follows from (2.3) that the possible limit points are the roots of $S(\theta; X_{obs}) = 0$. ■

2.3.2 Information and Asymptotic Variances

As suggested by Schafer (1997), imputation could also be used to obtain unbiased estimates of the information matrices required to summarize standard errors of maximum likelihood estimates. For completeness, we summarize such methods here and explicitly give the appropriate formulae in the context of multidimensional Brownian motion.

Let us return to a simple (finite dimensional) multivariate problem for the present discussion and let X be a column vector of the complete data. Suppose that X_{obs} is a vector of the observed data. As we have seen, typically X_{obs} is a function

of X and some variable Δ which determines the missingness, $X_{obs} = X_{obs}(X, \Delta)$, where X_{obs} may correspond to X with certain randomly determined components removed. Usually X_{obs} does not contain any additional information about the parameter over and above that already included in X, but we will not assume this at the outset. Suppose that we wish to estimate a parameter θ associated with the distribution of X. Then the joint probability density of the complete data can be expressed as

$$f(X_{mis}, X_{obs}, \Delta|\theta) = f_m(X_{mis}|X_{obs}, \theta) f_o(X_{obs}|\theta) f_\Delta(\Delta|X_{mis}, X_{obs}),$$

where f_m and f_o are the densities of the missing and observed components of the data respectively and f_Δ is the density of the random variable Δ. Note that we assume here that the distribution of Δ given X_{mis}, X_{obs} is not dependent on the parameter θ. From this, we can obtain the score functions

$$\begin{aligned} S(\theta; X) = S(\theta; X_{mis}, X_{obs}) &= \frac{\partial}{\partial\theta} \ln(f(X_{mis}, X_{obs}|\theta)) \\ S_m(\theta; X|X_{obs}) &= \frac{\partial}{\partial\theta} \ln(f_m(X|X_{obs}, \theta)) \\ S_o(\theta; X_{obs}) &= \frac{\partial}{\partial\theta} \ln(f_o(X_{obs}|\theta)) \end{aligned}$$

which satisfy

$$S(\theta; X) = S_m(\theta; X|X_{obs}) + S_o(\theta; X_{obs}) \tag{2.4}$$

and this is an orthogonal decomposition, i.e.

$$E[S_m(\theta; X|X_{obs}) S_o(\theta; X_{obs})|\theta] = 0$$

Here $S_o(\theta; X_{obs}) = E[S(\theta; X)|X_{obs}]$ (Fisher 1925) is the score function for the marginal distribution of X_{obs} the data we actually observe and $S_m(\theta; X|X_{obs})$ is

the score function obtained from conditional distribution of $X|X_{obs}$. Then from the decomposition above,

$$S_m(\theta; X|X_{obs}) = S(\theta; X_{mis}, X_{obs}) - E[S(\theta; X_{mis}, X_{obs})|X_{obs}]$$

Moreover taking another derivative to obtain the observed information, we have,

$$\begin{aligned} I(\theta; X) = I(\theta; X_{mis}, X_{obs}) &= -\frac{\partial^2}{\partial\theta^2} \ln(f(X_{mis}, X_{obs}|\theta)) \\ I_m(\theta; X|X_{obs}) &= -\frac{\partial^2}{\partial\theta^2} \ln(f_m(X|X_{obs}, \theta)) \\ I_o(\theta; X_{obs}) &= -\frac{\partial^2}{\partial\theta^2} \ln(f_o(X_{obs}|\theta)) \end{aligned}$$

with the decomposition

$$I(\theta; X) = I_m(\theta; X|X_{obs}) + I_o(\theta; X_{obs}).$$

The information in the joint data is the sum of the observed information in X_{obs} and the information in the conditional model $X|X_{obs}$. If the parameter θ is multivariate, we interpret the first derivatives $\frac{\partial}{\partial\theta}$ in the usual way as a gradient derivatives and the second derivatives $\frac{\partial^2}{\partial\theta^2}$ as the matrix of partial derivatives. This additivity of information holds both for the observed and for the Fisher information. For example see Little and Rubin (2002), 8.14 page 172. If we now take conditional expectations given X_{obs}, we obtain

$$\mathcal{I}_{com}(\theta; X_{obs}) = \mathcal{I}_{mis}(\theta; X|X_{obs}) + I_o(\theta; X_{obs})$$

where

$$\begin{aligned} \mathcal{I}_{com}(\theta; X_{obs}) &= E[-\frac{\partial}{\partial\theta} S(\theta; X)|X_{obs}, \theta] \\ \mathcal{I}_{mis}(\theta; X_{obs}) &= E[-\frac{\partial}{\partial\theta} S_m(\theta; X|X_{obs})|X_{obs}, \theta] \\ I_o(\theta; X_{obs}) &= -\frac{\partial}{\partial\theta} S_o(\theta; X_{obs}) \end{aligned}$$

Normally these information functions are evaluated at the maximum likelihood estimator of θ based on the observed data X_{obs}. This means that if we can find an expression for the complete data observed information, it is relatively easy to get an expression for the incomplete data observed information.

$$I_o(\theta; X_{obs}) = \mathcal{I}_{com}(\theta; X_{obs}) - \mathcal{I}_{mis}(\theta; X_{obs}) \tag{2.5}$$

We wish to use (2.5) to estimate $I_o(\theta; X_{obs})$. To this end, suppose for the given value of the observed data X_{obs} we repeatedly impute the missing observations to obtain replications $X_{mis}^{(j)}, j = 1, 2, ..., N$ sampled from the distribution of $X_{mis}|X_{obs}$. Then since

$$\mathcal{I}_{com}(\theta; X_{obs}) = E_\theta[-\frac{\partial}{\partial\theta} S(\theta; X_{mis}, X_{obs})|X_{obs}]$$

it can be estimated by a sample mean

$$-\frac{1}{N}\sum_{j=1}^{N} \frac{\partial}{\partial\theta} S(\theta; X_{mis}^{(j)}, X_{obs})$$

Similarly, the second term is estimated by using the conditional variance of the score function. Since

$$\begin{aligned}\mathcal{I}_{mis}(\theta; X_{obs}) &= E[-\frac{\partial}{\partial\theta} S_m(\theta; X|X_{obs})|X_{obs}, \theta] \\ &= var[S_m(\theta; X|X_{obs})|X_{obs}] = var[S(\theta; X_{mis}, X_{obs})|X_{obs}]\end{aligned}$$

we may estimate this unbiasedly using the sample covariance matrix of the completed data score functions,

$$\frac{1}{N-1}\left\{ \begin{array}{c} \sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs}) S'(\theta; X_{mis}^{(j)}, X_{obs}) - \\ \frac{1}{N}\left[\sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs})\right]\left[\sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs})\right]' \end{array} \right\}$$

Thus we estimate

$$I_o(\theta; X_{obs}) \simeq -\frac{1}{N}\sum_{j=1}^{N}\frac{\partial}{\partial\theta}S(\theta; X_{mis}^{(j)}, X_{obs}) \tag{2.6}$$
$$-\frac{1}{N-1}\left\{\begin{array}{c}\sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs})S'(\theta; X_{mis}^{(j)}, X_{obs}) - \\ \frac{1}{N}\left[\sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs})\right]\left[\sum_{j=1}^{N} S(\theta; X_{mis}^{(j)}, X_{obs})\right]'\end{array}\right\}$$

Of course there is no guarantee that the estimator on the right hand side of (2.6) is positive definite, but since it is a consistent estimator of $I_o(\theta; X_{obs})$ as $N \to \infty$, it should be positive definite for large enough N. See Little and Rubin, Chapter 9.

In the case of a multidimensional parameter, using the sample covariance matrix of the completed data scores $S(\theta; X_{mis}^{(j)}, X_{obs})$ may be a poor estimator of the actual covariance matrix. Indeed if there are k unknown parameters, we require at least $m = k(k+1)/2$ imputations so that this covariance matrix is non-singular. The estimators mentioned here can be iteratively updated during the run of the IS algorithm so none of the imputations are wasted. The could be done, for example, via a stochastic approximation procedure (see Delyon et al. 1999).

For multivariate normal increments z which have mean $\mu\Delta$ and covariance matrix $\Sigma\Delta$ (as in the case of Brownian motion where Δ is a scalar representing time),

$$\ln(f(z)) = -\frac{1}{2\Delta}v'\Sigma^{-1}v - \frac{1}{2}\ln(|\Delta\Sigma|) + c$$

and the complete data score function is

$$\frac{\partial}{\partial\mu}\ln(f(z)) = \Sigma^{-1}v$$
$$\frac{\partial}{\partial\Sigma_{ij}}\ln(f(z)) = \frac{1}{1+\delta_{ij}}\left[-\Sigma^{-1} + \frac{1}{\Delta}\Sigma^{-1}vv'\Sigma^{-1}\right]_{ij}$$

The observed information matrix is obtained by differentiating both of these again. It is a block matrix obtained using results for the matrix derivative of the inverse and the product rule for matrix differentiation, we have

$$
\begin{aligned}
-\frac{\partial^2}{\partial\mu'\mu}\ln(f(z)) &= \Delta\Sigma^{-1} \\
-\frac{\partial^2}{\partial\Sigma_{lk}\partial\Sigma_{ij}}\ln(f(z)) &= \frac{1}{1+\delta_{ij}}\left[\partial_{lk}\Sigma^{-1} - \frac{1}{\Delta}\left(\partial_{lk}\Sigma^{-1}vv'\Sigma^{-1} + \Sigma^{-1}vv'\partial_{lk}\Sigma^{-1}\right)\right]_{ij} \\
-\frac{\partial^2}{\partial\Sigma_{lk}\partial\mu}\ln(f(z)) &= -\partial_{lk}\Sigma^{-1}v
\end{aligned}
$$

where

$$\partial_{lk} = \frac{\partial}{\partial\Sigma_{lk}},\ v = z - \mu\Delta$$

$$\frac{\partial\Sigma^{-1}}{\partial\Sigma_{lk}} = \partial_{lk}\Sigma^{-1} = \begin{cases} -\Sigma^{-1}\Lambda_{lk}\Sigma^{-1} & \text{if} \quad l = k \\ -\Sigma^{-1}\left(\Lambda_{lk} + \Lambda_{kl}\right)\Sigma^{-1} & \text{if} \quad l \neq k \end{cases}$$

and Λ_{lk} is a matrix the same size as Σ with a 1 in position (l,k) and zeros everywhere else.

2.4 Bayesian Alternatives

In this section we examine some Bayesian approaches to the estimation problem. We consider an almost identical algorithm to the IS algorithm under the Bayesian philosophy that the parameter itself is a random vector. This modification will involve iteratively imputing the missing data, and conditional on the completed

dataset simulating an updated parameter value. The method which is generally known as "data augmentation" was formalized by Tanner and Wong (1987) and the iterative algorithm is known as the IP (Impute Posterior) algorithm. Under some regularity conditions such a method produces a Markov chain whose stationary distribution is the joint distribution of the parameter and the missing data. The sequence of parameter values obtained after an initial "burn in" period of the chain are dependent samples from the marginal posterior distribution of the parameter given the observed data. These samples can be used to construct empirical distributions of the parameter (or functions of the parameter) for the purposes of obtaining point estimates and Bayesian confidence intervals, as well as a variety of other statistics which may be of interest.

It is also important to note that the IP algorithm automatically produces what are known as Bayesianly proper multiple imputations. These are independent imputations of the missing data from the predictive distribution of the missing data. The predictive distribution is by definition the distribution of the missing data given the observed data which takes into account uncertainty associated with the parameter. Estimation schemes based on Bayesianly proper multiple imputation were introduced by Rubin in the context of survey sampling. They are intended to provide an approximation to the posterior mean and variance based on a small number draws from the missing data. Since independent samples of the missing data are required a subsequence of the missing data portion of the chain is sampled in this case for example every kth iterate where k is chosen large enough to ensure any dependence between samples is essentially zero. If point estimates are of interest it is argued that only a few imputations of the missing data is required

(Schafer 1997, p.106). In general the larger the fraction of missing data for an estimator the more imputations are required. Rubin (1987) gives crude estimates of the efficiency lost (in terms of variance) by using a finite number of imputations. Schafer (1997, p.106) points outs that the loss in efficiency is surprisingly small for modest amounts of missing data (20%-50%) when only 3-5 imputations are used.

2.4.1 Prior and Posterior Distributions

Multivariate Brownian motion data has independent multivariate normal increments and so the complete data likelihood and posterior density involves a product of normal density functions. In choosing a prior distribution most authors recommend appealing to simplicity by taking a conjugate prior. In the case of normal data this corresponds to the normal inverted-Wishart distribution. For this analysis we will instead opt for a non informative prior. For i.i.d. normal data choosing a conjugate prior or a non informative prior results in the same family of posterior distributions, and so it should not be much more difficult to incorporate some prior information (see Schafer 1997 Chapter 5). Since incorporating prior information can influence estimates substantially, choosing a non informative prior here will facilitate a comparison between the IP algorithm and its frequentist counterpart the IS algorithm. In the case of normal data the non informative improper prior can actually be obtained as a limiting case of the family of conjugate priors. It is given by

$$\pi(\theta) = |\Sigma|^{-\left(\frac{p+1}{2}\right)}.$$

In the case of equally spaced MVBM data with a time step of δ the complete data can be summarized by x_{ij}, $i = 0, ..., N$ and $j = 1, ..., p$ where $x_{ij} = X_{i\delta,j}$. Letting

$y_{ij} = x_{ij} - x_{i-1,j}$, for $i = 1, ..., N$ we have that $y_i = (y_{i1}, ..., y_{ip})$, $i = 1, ..., N$, are i.i.d. multivariate normal with mean $\delta\mu$ and covariance $\delta\Sigma$. Denote by $Y = (y_{ij})$ the matrix of complete data. Following the description in Schafer (1997 p.154), (under the non informative prior) conditional on the complete data the posterior distribution of the parameters can be sampled by noting that

$$\Sigma|Y \sim \frac{1}{\delta} W^{-1}\left(N-1, (NS)^{-1}\right),$$
$$\mu|\Sigma, Y \sim \frac{1}{\delta} MVN\left(\overline{y}, N^{-1}\delta\Sigma\right),$$

where $\overline{y}$ and S are the sample mean and covariance matrix (normalized by N) of the y_i respectively. MVN denotes the multivariate normal distribution and $W^{-1}(m, \Psi)$ denotes the inverse Wishart distribution with m degrees of freedom and scale matrix Ψ. Thus, a sample from the posterior is achieved in two steps. The first samples the diffusion matrix from the inverse Wishart distribution. The second step conditions on the first sample drawing the mean from a multivariate normal distribution. In the case of non-equally spaced data the situation becomes more complicated. Suppose the data y_i correspond to a time spacing of δ_i, then we have that $y_1, ..., y_N$ are independent random vectors with $y_i \sim MVN(\delta_i\mu, \delta_i\Sigma)$, $i = 1, ..., N$. In this case there is no simple way to sample from the complete data posterior, but it may be possible to implement a Gibbs sampling technique to iteratively sample the parameters here. In the case of a known mean μ we can sample from the posterior by noting that

$$\Sigma|Y, \mu \sim W^{-1}\left(N-1, (NS^*)^{-1}\right),$$

where S^* is the sample covariance matrix of $y_i^* = \frac{1}{\sqrt{\delta_i}}(y_i - \delta_i\mu)$, $i = 1, ..., N$.

2.4.2 The IP Algorithm and Estimation

Sampling from the complete data posterior was described above. Suppose our Brownian motion data is partitioned according to its missing and observed values as $X = (X_{obs}, X_{mis})$. Our goal is to produce draws from the marginal posterior given the observed data, namely $\pi(\theta|X_{obs})$. Suppose $\left\{\theta^{(k)}, X_{mis}^{(k)}\right\}_{k=1}^{K}$ is a sequence of draws from the joint distribution of the parameter and the missing data $\pi(\theta, X_{mis})$. The principle of data augmentation tells us that $\theta^{(k)}$, $k = 1, .., K$ are then implicitly draws from $\pi(\theta|X_{obs})$. Sampling directly from $\pi(\theta, X_{mis})$ is difficult (if not impossible) due to its high dimensionality. The IP algorithm solves this problem iteratively by blocking the parameter vector and missing data in a Gibbs sampler. The algorithm is described as follows. Initialize a parameter guess $\theta^{(0)}$ and iterate the following two steps

1. Sample $X_{mis}^{(k)} \sim X_{mis}|X_{obs}, \theta^{(k-1)}$. (I - Impute step)
2. Sample $\theta^{(k)} \sim \theta|\left(X_{obs}, X_{mis}^{(k)}\right)$. (P - Posterior step)

This algorithm is very closely related to its frequentist counterpart the IS algorithm. The difference is that where the IS algorithm calculates a maximum likelihood estimate the IP algorithm simulates a new parameter value. Step 1 is performed by noting that $X_{mis}|X_{obs}, \theta$ has a multivariate normal distribution. This distribution may be high dimensional and it may be convenient to break this problem up into smaller blocks. Step 2 is simply sampling from the complete data posterior which was described in the previous section. After an initial "burn in"

period the sequence of parameters forms a dependent sample from the posterior distribution. If an independent sample is required one may take every jth iterate, where j is chosen large enough to ensure that any dependencies have vanished. For the purposes of sampling posterior means, variances, quantiles, empirical distribution functions and higher moments dependent samples can be used in the same way as independent samples. For example, a point estimate for θ is the mean of the posterior distribution which can be estimated using successive samples of θ after an initial burn in period of the IP algorithm. Schafer (1997, Chapter 4) discusses for what quantities dependent samples are appropriate.

2.4.3 Estimation via Bayesianly Proper Multiple Imputation

Proper multiple imputation borrows from the IP algorithm provided in the previous section. It involves simulating the missing data from its predictive distribution which can be obtain by integrating over the posterior distribution of the unknown parameter θ

$$\pi\left(X_{mis}|X_{obs}\right) = \int \pi\left(X_{mis}|X_{obs},\theta\right)\pi\left(\theta|X_{obs}\right)d\theta.$$

Drawing directly from this distribution is often not an easy exercise and MCMC methods provide the machinery for accomplishing such a task. Multiple imputation works under the premise that the statistician is equipped with inference methods for the complete data case. After producing several imputations the complete data estimators are calculated for each completed data set and then aggregated to arrive at overall estimates. Suppose $X_{mis}^{(k)}$, $k = 1, ..., m$ are independent proper imputations and $\widehat{Q}(X)$ and $\widehat{U}(X)$ are complete data estimators of the quantity

$g(\theta)$ and its variance $VAR(g(\theta))$, respectively. For each completed dataset we get a point estimate along with its variance, denote these by $\widehat{Q}_{(k)} = \widehat{Q}(X_{obs}, X_{mis}^{(k)})$ and $\widehat{U}_{(k)} = \widehat{U}(X_{obs}, X_{mis}^{(k)})$, respectively. As one might expect, the combining rules dictate that the multiple imputation point estimate is given as the mean of the individual points estimates:

$$\overline{Q} = \frac{1}{m} \sum_{k=1}^{m} \widehat{Q}_{(k)}$$

The variance is broken down into two components. The first is the within imputation variance which is the mean of the point estimates of variance for each complete dataset. The second is the between imputation variance which is the variance of the point estimates themselves. The combined estimator of variance is given as

$$\widehat{VAR(\overline{Q})} = \frac{1}{m} \sum_{k=1}^{m} \widehat{U}_{(k)} + \left(1 + \frac{1}{m}\right) \frac{1}{m-1} \sum_{k=1}^{m} \left(\widehat{Q}_{(k)} - \overline{Q}\right)^2.$$

Schafer (1999) provides approximations for confidence intervals and other inferential quantities of interest. The only difficult part of multiple imputation is generating independent Bayesianly proper imputations. The IP algorithm of the previous section serves the purpose here. It draws a sequence $\left\{\theta^{(k)}, X_{mis}^{(k)}\right\}_{k=1}^{K}$ from the joint distribution of the parameter and the missing data $\pi\left(\theta, X_{mis}\right)$. $X_{mis}^{(k)}$ are implicitly draws from the predictive distribution, however successive draws will be dependent. To achieve independent draws we need to subsample the chain at far enough points from each other in the sequence to ensure that dependence has vanished. This can be achieved by examining the autocorrelation of (or functions of) the parameter estimates. This does not guarantee independence but provides a good practical guide which is condoned by most authors.

2.5 Gibbs Sampling and Markov Processes

Imputing large multivariate datasets according to a Markov model with a general pattern of missingness is a computationally challenging task. Such imputation is required for both the IP and IS algorithm. Similar problems have been encountered in the literature, particularly in the context of data augmentation in diffusion models, see for example Elerian, Chib and Shepard (2001) and Eraker (2001). It is often the case that imputation of any one value will depend on all of the observed data. Even if the conditional distribution of the missing data given the observed can be easily computed, one may be required to sample from high dimensional multivariate normal distribution which is computationally infeasible. Instead, the problem of imputing the entire dataset can be broken down into several smaller imputation problems, each of which is a low dimensional problem. What we discuss here can be applied to any function of multivariate Brownian motion. Denote the $i'th$ row of X as x_{i*}, then we can break up our multivariate dataset into its rows $x_{0*}, ..., x_{n*}$, each of which is a set of concurrent values containing at most p missing values. Using the Gibbs sampler we can impute the data X by iteratively imputing each row of X conditionally on the others. Such an algorithm eventually produces a draw from $X_{mis}|X_{obs}$. Due to the Markov property of Brownian motion, the conditional distribution of any row given the others is the same as the conditional distribution of that row given its two nearest neighbouring rows. One iteration of the Gibb's sampling algorithm in this Markovian framework is given by

0. simulate $x_{0*}^{(k+1)}$ from $x_{0*}|x_{1*}^{(k)}$
1. simulate $x_{1*}^{(k+1)}$ from $x_{1*}|x_{0*}^{(k+1)}, x_{2*}^{(k)}$

2. simulate $x_{2*}^{(k+1)}$ from $x_{2*}|x_{0*}^{(k+1)}, x_{1*}^{(k+1)}$

....

p. simulate $x_{p*}^{(k+1)}$ from $x_{p*}|x_{p-1*}^{(k+1)}$.

Of course an initial value for X_{mis} must be chosen. As $k \longrightarrow \infty$, $X^{(k)} = \left(x_{0*}^{(k)}, ..., x_{p*}^{(k)}\right)$ converges to a draw from $X|X_{obs}$. The data do not need to be broken up into blocks according its rows, one may choose any partition for the sampling procedure. However to exploit the Markovian nature of the process partitions should consist of one or more consecutive time points. Also, sampling of the blocks can be done in any fixed order or by randomly permuting blocks, and this may speed up convergence in some circumstances.

2.6 Estimation Examples

In the results that follow for both the IS and IP algorithms the data were imputed by blocking the data according to time points and using a random block Gibbs sampler. Two iterations of the Gibbs sampler was run. This may seem like too few a number, however in all simulated data examples convergence to the correct parameter estimates was obtained under this choice. One such example is provided below. For the IS algorithm we found that it is convenient to "burn-in" the sequence of estimates before employing the Robins Monro averaging procedure. By "burn in" here we simply mean that the next imputation is generated conditionally on only the previous completed dataset maximum likelihood estimate. There is no general best rule to use in deciding when the averaging should begin so we have designed

a very simple rule which seems to work well. It is based on the fact that every non averaged output parameter sequence eventually enters a period of stochastic oscillation. Of course without some form of averaging this sequence will never converge to a value. For a given number of D iterations the sequence decides to start averaging when approximately 50% of the last D and $D/2$ iterations produced "up moves". A simpler rule looks at only the previous D iterations and this rarely causes premature averaging. The chosen averaging rule is quite conservative in that averaging may often be employed far before the one dictated without losing any accuracy. By resetting the random number seed we plot both the averaged and unaveraged sequences to examine the rule choice.

2.6.1 Simulated Data Example

To illustrate the effectiveness of the different methods we apply them to simulated data in hopes to recover reliable parameter estimates. At the moment we will examine the effectiveness for a moderately sized problem. Here we consider a bivariate Brownian motion X with drift μ and covariance matrix Σ. Let us suppose that X represents the logarithm of a stock price as is postulated under the Black-Scholes options pricing model. Suppose that the first component of X is a heavily traded asset and that none of its observations are missing. We will assume that the second component of X is not traded at every time instant giving rise to some missing data. There are several ways to achieve this. For example, we could draw i.i.d. indicator random variables which do not depend on the data or $\theta = (\mu, \Sigma)$ to censor the second component's observations. Such a mechanism clearly leads to missing data that are MCAR (missing completely at random). For this test we opt

for a more complicated missingness mechanism which depends on some observed data. The second component will be censored according to the following rule. If the percentage change in the value of component one is larger than 1% than component two is deemed missing. This rule is mainly chosen out of convenience to produce a missingness mechanism that leads to data which are MAR (missing at random). The data are MAR since the distribution of the missingness indicator depends only on the observed data and does not depend on θ directly. As was discussed, such a missingness mechanism is ignorable for estimation purposes. Under the above mentioned setting $N = 3500$ equidistant data points were simulated with a time step of $dt = 1/252$. The underlying parameter of this two dimensional Brownian motion are chosen to be

$$\mu = (0.1, 0.05)$$

and

$$\Sigma = \begin{bmatrix} 0.0625 & 0.07 \\ 0.07 & 0.16 \end{bmatrix}.$$

This corresponds to stock volatilities of 25% and 40% for each respective component and a correlation coefficient of 0.7. Under this setting the missingness indicator M of the data is a 3500×2 matrix which has zeros along the first column (corresponding to complete observation of the first stock) and a ones in the second column corresponding to component one returns in excess of 1%. This lead to 1,865 missing values for the second component which is about 53% of component two's observations. It was mentioned that Kofman and Sharpe (2003) provided a survey of papers in financial journals which explicitly recognized the presence of missing data. They also reported that the listwise deletion method was the most popular treatment of such data. In our example, listwise deletion corresponds

to deleting every observation of the bivariate Brownian motion in which at least one component was missing. This would require that we remove the 1,865 of the data points of which we only observe the first component of the bivariate Brownian motion. This may do little damage to the efficiency in estimating the drift which is known to be a noisy estimator in any case. Discarding such data has greater consequences in estimating the terms within the diffusion coefficient. It is well known that a continuous observation of a diffusion leads to perfect estimation of its diffusion term. The more refined our observation in a given time interval the greater the precision we will achieve, and thus it clearly inefficient to throw away half of the observed data points. We could also consider a method of estimation which estimates the volatility and covariance components of each stock using the observed cases and then combining these estimates to get an estimate of correlation. It is documented in the statistical literature that such a method can produce correlation values outside of the interval $[-1, 1]$ which is clearly undesirable.

We run a fixed number of 25 iterations of the EM algorithm for this example which was found to be sufficient to obtain convergence. The IS and IP algorithms were both run for 1000 iterations. Every 100th imputation of the IP algorithm was used as an imputation from the predictive distribution. Since we have explicit maximum likelihood estimators and information estimates (and hence variance estimates) for the complete data case, maximum likelihood estimation will be the adopted complete data rule for analyzing Rubin's multiple imputation combining rules. The complete data maximum likelihood estimates based on the uncensored dataset can be easily calculated and are displayed in Table 2.1 (standard errors are given in brackets).

	μ_1	μ_2	Σ_{11}	Σ_{22}	Σ_{12}
Estimate	0.0480	-0.0213	0.0623	0.1634	0.0720
	(0.0670)	(0.1085)	(0.0015)	(0.0039)	(0.0021)

Table 2.1: Uncensored Results

The diffusion coefficient is estimated quite well. This is due to the fact that every increment of the process gives additional information of the variance parameters. The finer our observation of the process the closer to the true value we can expect our estimate to be. In the limit, observing a continuous path of any time length would lead to perfect estimation of the diffusion coefficient which is a consequence of properties of the quadratic variation of the process. The same is not true for the drift. As expected, the estimates of the drift parameters are very poor with relatively large standard errors because our estimate of drift is based uses only one observed sample path of the process. In fact it depends only on the first and last observed values of this path, and these can be quite variable. In the presence of missing data we should expect similar results. In particular, for the first component which has no missing values we expect our corresponding estimates of drift and variance to be unchanged. Table 2.2 displays the results of the incomplete data methods applied to the simulated censored data.. Again, the diffusion coefficient is estimated with reasonable accuracy. For the drift and variance of component one all methods (with the exception of the IP algorithm) give the same result. The negligible discrepancy in the IP method is observed due to the random nature of the algorithm and is ignorable. All methods show results which are close to the true parameter values. It is interesting to note that all methods also give very similar standard errors making it difficult to differentiate between them. Also,

	μ_1	μ_2	Σ_{11}	Σ_{22}	Σ_{12}	ρ
EM	0.0480	-0.0218	0.0623	0.1636	0.0720	0.7126
IS	0.0480 (0.0670)	-0.0218 (0.1086)	0.0623 (0.0015)	0.1638 (0.0058)	0.0721 (0.0026)	0.7135
IP	0.0508 (0.0661)	-0.0190 (0.1121)	0.0624 (0.0015)	0.1640 (0.0063)	0.0720 (0.0028)	0.7113
MI	0.0480 (0.0675)	-0.0214 (0.1095)	0.0623 (0.0015)	0.1662 (0.0061)	0.0730 (0.0025)	0.7170

Table 2.2: Comparison of Estimation Results

the running times of both the IS and IP algorithm were very similar although for this simulated data example we are mainly interested in accuracy of the methods. To ensure convergence of the stochastic algorithms it often suffices to examine the output sequences of the various algorithms. By resetting the random number seed and re-running the IS algorithm without averaging it is possible to assess the averaging rule for the algorithm. Figure 2.2 plots the IS algorithm output sequence for the variance of the second component and the correlation coefficient with and without Robins-Monro averaging. Notice that once the averaging starts the sequence quickly curves into a final state and shows little more movement for the remainder of the run of the algorithm. For non averaged stochastic EM algorithms most authors recommend taking the mean of the tail of the sequence as a point estimate of parameters. Here the non-averaged sequence seems to oscillate around the averaged sequence indicating heuristically that Robins-Monro averaging results in convergence to this mean. Figure 2.3 shows the IP algorithm output sequences. There are no unusual patterns present in the sequence providing some evidence of

convergence of the algorithm. The empirical posterior distributions are also plotted and can be used to provide most relevant summaries for inferential purposes. For Rubin's multiple imputation method proper variance estimates depend critically on independent imputations. The approach in the literature for ensuring this is to examine autocorrelation plots of the output parameter sequences. Figures 2.4 and 2.6 show such plots for the ρ and Σ_{22} parameters, respectively. As expected there is some correlation in the initial lags but both plots show that the correlation in later lags in not significant. Every 100th imputation in the IP algorithm was used for the MI procedure. The autocorrelation seems to die down by lag 20 for both plots providing evidence that 100 lags is more than sufficient to ensure independence. The autocorrelation function for the Σ_{11} is also plotted in Figure 2.5. Since the data are fully observed on this parameter the only randomness in the sequence is independent noise as is confirmed in the plot. The EM algorithm output is plotted in Figure 2.7. The sequences seem to flatten out by 20 iterations indicating that 25 iterations is indeed sufficient. It is also important to compare running times of the respective algorithms. The IS algorithm running time was roughly 59 minutes while that of the IP algorithm was about 55 minutes. One would expect the IP algorithm to take longer since it requires more simulation per iteration. However, this time is offset by calculation of the information matrix at each iteration of the IS algorithm. The difference in these running times is negligible and not a cause for choosing one method over the other. Figure 2.8 shows the IS and EM output sequences versus running time. Recall that both algorithms start at the same initial value. Notice that the IS algorithm enters the correct parameter region much more quickly than the EM algorithm. It is important to note that both sequences converge to the same value which is to be expected and is supported by Theorem 1.

In this example the IS algorithm seems to outperform the EM algorithm in terms of time until convergence. The IP algorithm shows similar comparisons to the EM. Not only are the IP and IS algorithms fast relative to the EM algorithm, but they also automatically output an estimate of the covariance matrix of the parameter vector. Error estimation is often a tedious task under the EM algorithm and it is much more conveniently accommodated by the stochastic algorithms.

The results show that all methods do indeed give similar results in comparable time. In the case of MVBM there is at least one advantage to using the IS algorithm over the IP algorithm. The complete data maximum likelihood estimators are known explicitly for Brownian motion data. Once an imputation is performed, estimates of the drift and diffusion coefficients follow immediately with little computational effort. Contrast this to the IP algorithm. Once an imputation is performed the parameters must be drawn from the posterior distribution. For non equally spaced data with an unknown drift coefficient the posterior is not readily available. There is some remedy available if one is willing to sample these parameters using a Gibbs sampler, but this introduces another layer of approximation into the algorithm which can only negatively affect convergence times. An advantage of the IP algorithm is that the parameter sequence itself provides the means for calculating Bayesian confidence intervals. The IS algorithm devotes additional computation to the analogous goal of generating standard errors of the estimates. Also, the IP algorithm allows one to introduce prior information, at the risk of leading to an intractable posterior. Provided that one could derive a method for sampling from the posterior distribution in the general case it seems that each algorithm only carries slight advantages over the other. The computational time

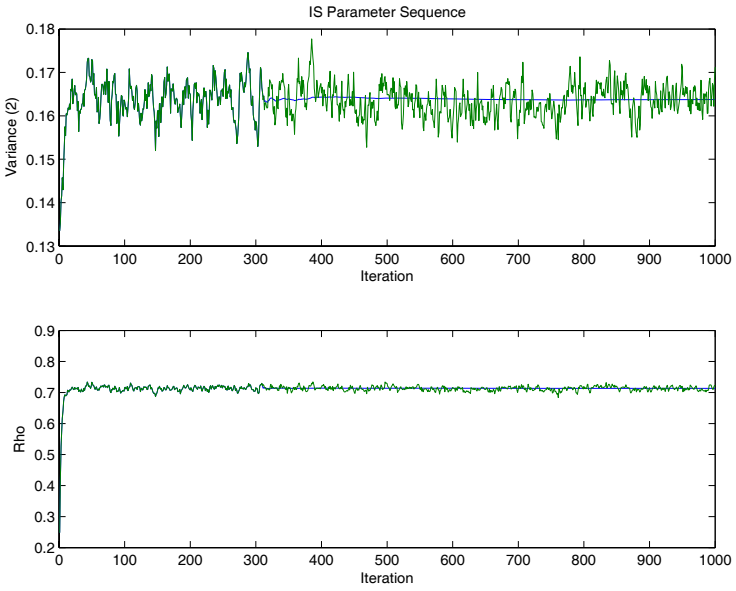

Figure 2.2: IS algorithm output sequence. Both the Robins-Monro averaged and unaveraged sequences.

in calculating standard errors for the IS algorithm would likely roughly offset that used to sample from a more complex posterior. In the case of Multivariate Brownian Motion data, the choice of methodology ultimately depends on ones philosophy of statistical inference.

2.6.2 Estimating the Parameters for Intraday Equity Data

One Minute Data

Fitting a multivariate continuous time model under asynchronous trading is a challenging problem, in part due to the large number of essentially missing observations.

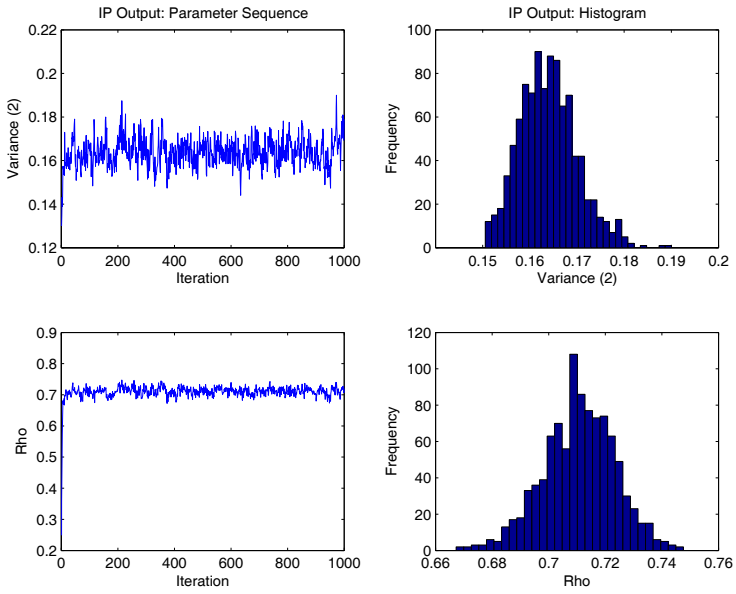

Figure 2.3: IP algorithm output sequences and empirical posterior distributions for the variance of the second component and the correlation coefficient.

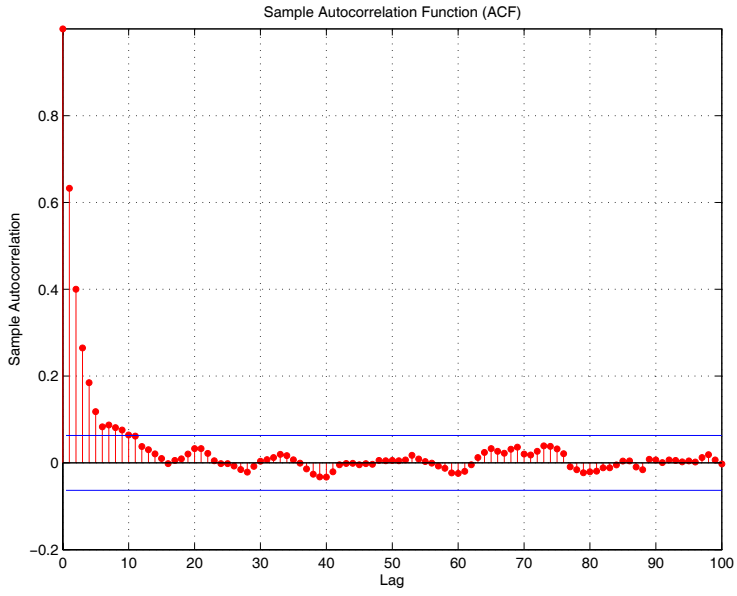

Figure 2.4: Autocorrelation plot of the IP output sequence of ρ.

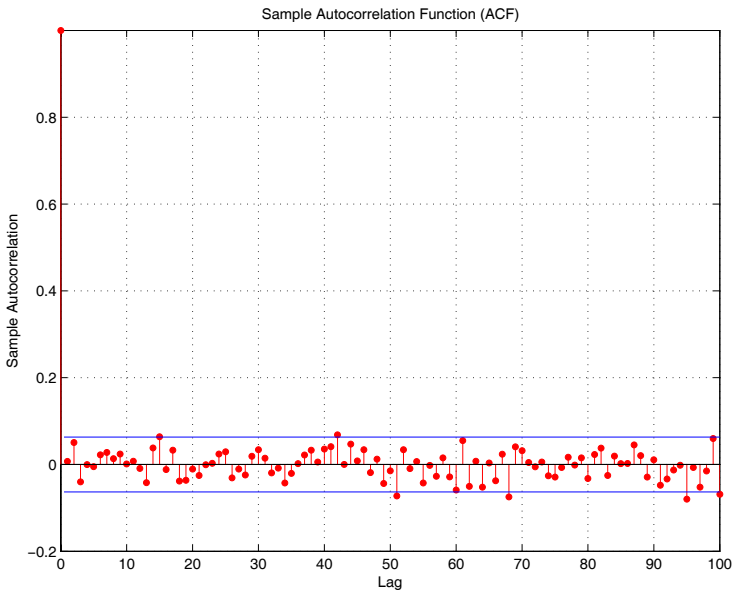

Figure 2.5: Autocorrelation plot of the IP output sequence for the Σ_{11} parameter.

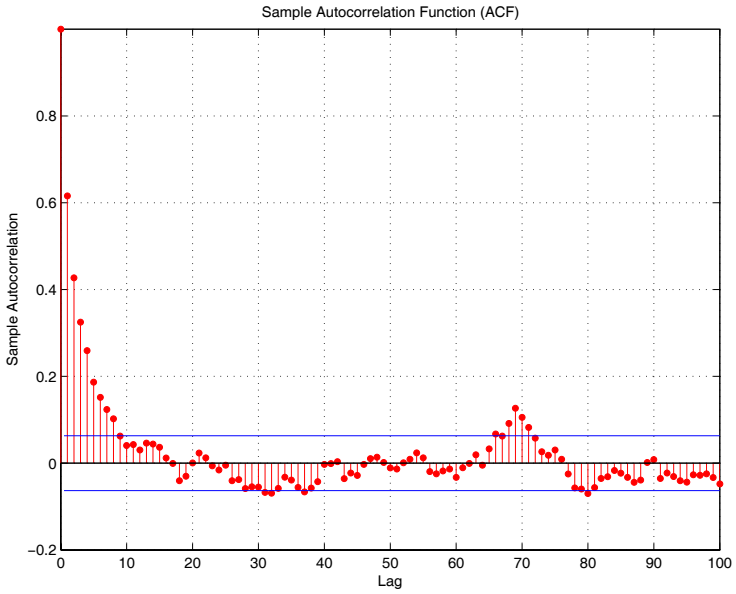

Figure 2.6: Autocorrelation plot of the IP output sequence for the Σ_{22} parameter.

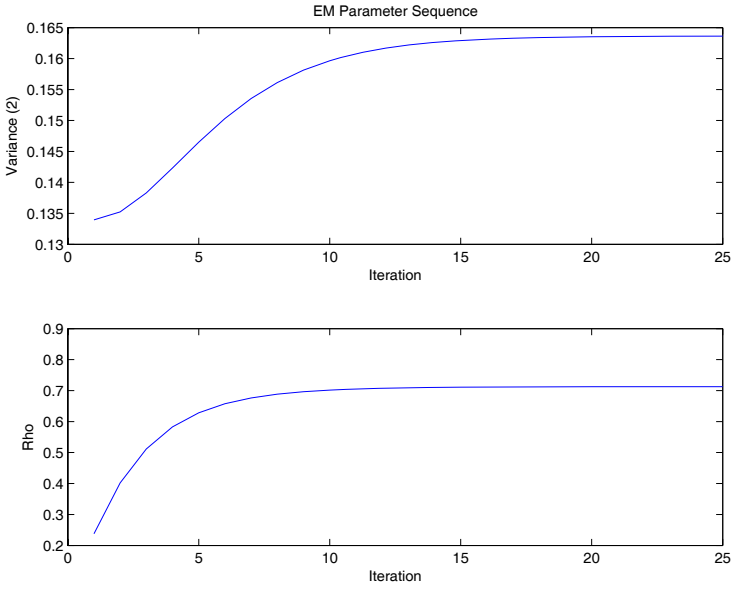

Figure 2.7: The EM algorithm output sequence.

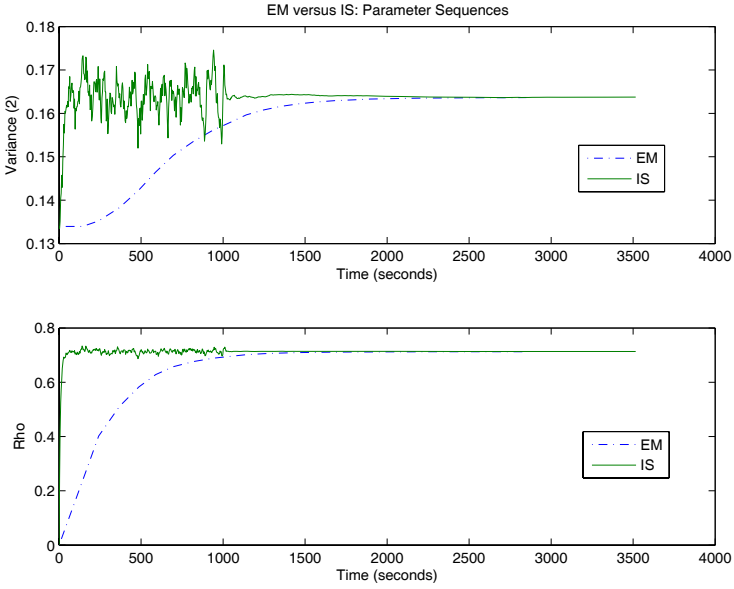

Figure 2.8: EM and IS algorithm output sequences versus time.

Although geometric Brownian motion is known to be a poor model for financial tick data, to demonstrate feasibility, we use the methods of the previous sections to obtain parameter and variance estimates under this model. The goal is to verify that covariate information is useful for estimation and that imputation is a practical method for extracting this information. The data set consists of one minute discretized tick data from the Toronto Stock Exchange (TSE) dated on February 2, 2005 on three bank stocks: the Bank of Nova Scotia (BNS), Royal Bank (RY) and Bank of Montreal (BMO). The data consist of 392 time points with 12, 51 and 45 missing values for BNS, RY and BMO, respectively. A stock price at a particular time point is reported missing if no sales were made in the last minute of trading on that stock.

If we are interested in estimating the parameters of a geometric Brownian motion for BNS this can be done several ways: examining BNS in isolation, or considering the joint data sets of BNS and one or more other stocks. We compare estimation results for BNS in isolation, with results obtained by adding covariate information. In the results that follow the Impute-Solve algorithm was run for 1000 iterations, which was found to be more than enough to ensure convergence in this example. This required under 3 minutes of CPU time for the 2 dimensional data set and about 8 minutes for the 3 dimensional data set. Table 2.3 displays the results of all three analyses, standard errors are given in brackets below each estimate. For BNS in isolation there is no missing data and thus standard errors can be computed by inverting the observed information directly. Notice that the estimates of the BNS parameters with covariate information are consistent with the analogous one dimensional estimates, as one would expect since all the estimators used in each case

	Number of Symbols in Analysis		
	1	2	3
μ_{BNS}	-0.7341 (2.7198)	-0.7389 (2.7177)	-0.7584 (2.7177)
μ_{RY}	-	1.7655 (3.2975)	1.7655 (3.2975)
μ_{BMO}	-	-	-1.6748 (2.3060)
σ^2_{BNS}	0.0258 (0.0019)	0.0258 (0.0019)	0.0258 (0.0019)
σ^2_{RY}	-	0.0389 (0.0030)	0.0389 (0.0030)
σ^2_{BMO}	-	-	0.0189 (0.0014)
$\sigma_{BNS,RY}$	-	0.0024 (0.0018)	0.0023 (0.0018)
$\sigma_{BNS,BMO}$	-	-	0.0010 (0.0012)
$\sigma_{RY,BMO}$	-	-	0.0009 (0.0016)

Table 2.3: One Minute Data Estimation Results

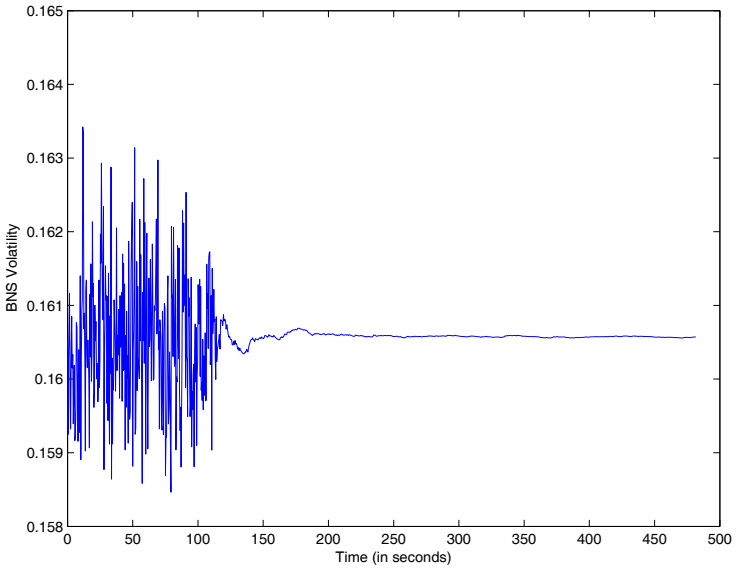

Figure 2.9: The convergence of the IS algorithm parameter sequence for the BNS volatility parameter.

are unbiased. The BNS drift estimate μ_{BNS} is sensitive to the covariate information, showing both a reduction in variance and slight shifts in the estimate with the inclusion of each covariate. Though it is not evident in the first four significant digits shown in Table 2.3, all estimators showed non-increasing variances with the inclusion of covariate information.

Figure 2.9 shows the parameter sequence for the BNS volatility parameter generated by the Impute Solve algorithm. The sequence seems to begin averaging after about 100 iterations and stabilizes soon after 200 iterations. Plots for the other volatility (as well as the mean) parameters are similar, convergence is achieved after about 250 iterations. The covariance parameters also show relatively rapid convergence. Figure 2.10 shows the parameter sequence for the BNS - RY corre-

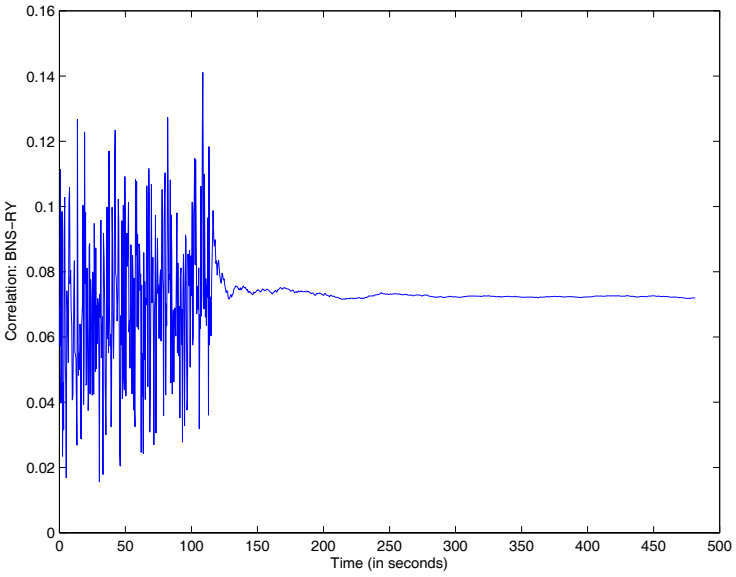

Figure 2.10: The convergence of the IS algorithm parameter sequence for the BNS-RY correlation parameter.

lation parameter. Convergence times will depend on the amount of missing data present in the estimators. For example, for a completely asynchronous dataset the mean and variance parameters would experience faster convergence than the covariance parameters. The latter converge more slowly due to the fact that less information is available to estimate them since the asset prices are never observed simultaneously. If all observations are present for a given estimator then the algorithm will converge after one iteration, as was the case for the RY drift parameter. In general, missingness patterns which cause slow convergence for the EM algorithm will also lead to slow convergence in the IS algorithm. The advantage of the IS algorithm is in its ability to better accommodate large datasets arising from Markov processes using Gibbs sampling. The only way to achieve similar efficiencies for an implementation of the EM algorithm is to approximate conditional expectations by using observations in a fixed time window, an alternative which we found to be unstable.

Continuous Tick Data

The IS algorithm seems to work well for the one minute data attaining convergence quickly and producing standard errors which are consistent with the one dimensional fully observed data case. We can consider the same stocks on the same day of trading used in the previous example, but instead we examine the continuous dataset without rounding observations to the nearest minute. The joint dataset of the BNS and RY stock in continuous time has a total 2183 time points. 822 and 1285 observed values for BNS and RY respectively, with 76 of these observed values occurring synchronously. If we include the BMO stock the dataset expands

	μ_{BNS}	μ_{RY}	Σ_{BNS}	Σ_{RY}	$\Sigma_{BNS,RY}$
Run #1: Initial $\rho = 0.5$	-0.2857	1.6963	0.2197	0.2400	0.1780
Run #2: Initial $\rho = -0.5$	-0.2929	1.6963	0.2200	0.2424	-0.1790

Table 2.4: Continuous Tick Data Estimation Results

to 3092 time points. 1000 iterations of the IS algorithm was performed on the two dimensional dataset. The initial guess of the algorithm was chosen by examining the one dimensional series for each of the two stocks. Since this dataset is close to asynchronous we examine the performance of the IS algorithm for two initial covariance parameters, one positive and one negative. Doing this leads to some interesting results. Table 2.4 shows parameter estimates for two runs of the IS algorithm, the first (Run #1) being initialized with a correlation coefficient of 0.5 and the second (Run #2) with a correlation of -0.5. Notice that these estimated parameters are very different from those obtained from one minute data. This is due to a combination of model misspecification and additional noise in the data coming from the bid-ask spread. This is not central to this analysis as we are accessing the feasibility of the IS algorithm and not the fit of Brownian motion to tick data. With the exception of the covariance, the parameter estimates are consistent across runs. The IS algorithm is only guaranteed to converge to a root of the missing data score function, and so one may be required to examine additional criteria to determine whether or not the achieve limit is indeed a maximum. of the likelihood. Unfortunately, for most interesting high dimensional problems the likelihood may not be available in a convenient form. For this problem we can examine the Hessian of the likelihood evaluated at the estimated solution. If all eigenvalues are negative then the point of interest is indeed a (local) maximum.

For this example we find that the solution attained in Run #1 is a local maximum while the Hessian at the solution in Run #2 has eigenvalues which are both positive and negative. Hence, we can conclude that of the two roots the solution obtained in Run #1 is the only candidate to be the maximum likelihood estimate. This can be considered a cautionary note to the use of this or any other estimation algorithm. The algorithm should be run with several initial guesses to ensure the solution attained is a maximum. Also, note that examining the Hessian matrix is a means to decide whether a solution is a local extreme, but it cannot distinguish which of several extremes is the global maximum. In general, measuring the likelihood at each solution is one criteria for distinguishing between multiple local solutions.

Another interesting point is that the covariance estimates from the two runs are almost symmetric about zero. This phenomena was observed for other trading days as well. This would seem to indicate some type of symmetry in the likelihood in ρ. To examine this further data were simulated according to the time points and missingness indicator of the real data. The parameters used were those estimated from the continuous data. Again, two runs of the IS algorithm were performed testing the sensitivity of the solution to the initial conditions. However, for simulated data the IS algorithm found the correct solution regardless of starting value. In particular, it had no apparent difficulties in converging to the correct covariance parameter in each of the runs. Apparently the IS algorithm is able to accommodate the missingness pattern and high dimensionality of continuous tick data, but some feature in the real stock data leads to a nearly symmetric local roots of the likelihood equation. This could be due to the fact that the model is misspecified or that the MAR assumption does not hold. As a final note it is worth mentioning

that the above experiments were tried in the three dimensional continuous time setting by adding the BMO series to the dataset. Even with the introduction of a third variable the IS algorithm had no difficulty converging. It was once again observed that the root may depend on the starting value of the algorithm, and it is recommended to test several initial covariance estimates.

Chapter 3

Imputation for Diffusion Processes

3.1 One Dimensional Diffusions

Most simulation techniques for diffusions utilize standard Euler-based approximations over a finite time grid which are known to compound and propagate error along the imputed path. Similarly if a path segment or an observation is missing, the approximations are normally based on a local Brownian Bridge approximation. In this section we present an imputation method for a general univariate diffusion which is based on acceptance-rejection sampling. The method is quite powerful, yet simple, and relies on the form of the Girsanov density. Imputation for univariate processes can be useful for a number of applications including estimation and variance reduction in Monte Carlo simulation. For estimation, imputation can be used to augment a data set with unequally spaced times to one with equally spaced times, thereby simplifying the form of the likelihood function and the resulting estimators. For pricing path dependent derivative securities by Monte Carlo

simulation, stratifying the sample paths of the process is a well-known and commonly used variance reduction technique. This requires bridging the initial value of the process to the terminal value at one or more intermediate times, a problem well-suited for imputation.

Consider a scalar Ito process of the form

$$dX_t = \mu(X_t)dt + \sigma(X_t)dW_t$$

where W_t is an ordinary Wiener measure and where the drift term μ and the diffusion σ may depend on some parameters. We assume that the estimation of these parameters is reasonably straightforward if the process X_t is observed with infinite precision in continuous time, but of course, this ideal situation is never realized. Suppose we wish to generate an observation from X_s given the value $X_0 = x_0$ and $X_t = x_t$, $0 < s < t$. A simple well-known transformation reduces this to a problem with constant diffusion term. Let

$$s(x) = \int_0^x \frac{1}{\sigma(z)}dz$$

(assuming this integral is well-defined) and let g be the inverse function of s. Then by Ito's formula, the process $Y_t = s(X_t)$ satisfies a diffusion equation of the form

$$\begin{aligned} dY_t &= s'(X_t)\{\mu(X_t)dt + \sigma(X_t)dW_t\} + \frac{1}{2}s''(X_t)\sigma^2(X_t)dt \\ &= \{s'(g(Y_t))\mu(g(Y_t)) + \frac{1}{2}s''(g(Y_t))\sigma^2(g(Y_t))\}dt + s'(g(Y_t))\sigma(g(Y_t))dW_t \\ &= a(Y_t)dt + dW_t, \text{ say} \end{aligned}$$

where

$$a(y) = \frac{\mu(g(y))}{\sigma(g(y))} - \frac{1}{2}\sigma'(g(y)).$$

It is therefore sufficient to generate a value of the process Y_t in the special case of a unit diffusion coefficient conditional on the value of the process Y at the endpoints $Y_0 = y_0 = s(x_0)$ and $Y_t = y_t = s(x_t)$. The problem is approached from two different perspectives. The first considers the stochastic process at finite time points and the second utilizes the fact that a stochastic process can be viewed as a random element of the space of continuous functions. Both approaches rely on the principle of acceptance-rejection sampling which is described formally in the appendix. Acceptance-rejection sampling is a simulation technique that allows one to sample from a density (usually referred to as the target density) using simulations from another density (the proposal density). This is useful when one knows the form of the target density but is unable to use a traditional simulation scheme to generate random variates from it. The method only requires knowledge of the target density up to a constant of proportionality, and the ability to generate events with probability proportional to the ratio of the densities of interest.

3.1.1 Transition Density Approach

In this section we examine the imputation problem on the basis of the finite dimensional distributions of the given diffusion process. This approach depends on the expansion of the transition density of a diffusion given by Dacunha-Castelle and Florens-Zmirou (1986). For the diffusion process Y and times $0 < s < T$ denote by $p_T^Y(x, y)$ the probability density function of $Y_T|Y_0 = x$, and define $p_s^Y(y; x, z, T)$ to be the density of $Y_s|Y_0 = x, Y_T = z$. The latter will be referred to as the bridge density of the process Y. Let $p_T^W(x, y)$ and $p_s^W(y; x, z, T)$ be defined similarly for the Brownian motion process W. Now, Dacunha-Castelle and Florens-Zmirou

(1986) show that the transition probability density function of Y can be written as

$$p_T^Y(x,y) = n_T(x,y)U_T(x,y)$$

where $U_T(x,y) = \exp\{A(y) - A(x)\}H_T(x,y)$, $n_T(x,y)$ is the normal probability density with mean x and variance T, $A(x) = \int^x a(z)dz$, and

$$H_T(x,y) = E\left[\exp\left\{-\frac{1}{2}\int_0^T (a^2 + a')(W_z^{(x,y)})dz\right\}\right],$$
$$U_T(x,y) = E\left[\exp\left\{\int_x^y a(z)dz - \frac{1}{2}\int_0^T (a^2 + a')(W_z^{(x,y)})dz\right\}\right],$$

where $\left\{W_u^{(x,y)} : u \geq 0\right\}$ is the Brownian Motion process conditional on $W_0 = x$ and $W_T = y$. We are interested in imputing the value Y_s given the values $Y_0 = y_0$ and $Y_T = y_T$ for $0 < s < T$. Note that if W denotes a standard Brownian motion process then the conditional probability density of $Y_s|Y_0 = y_0, Y_T = y_T$ is given by

$$\begin{aligned} p_s^Y(y_s; y_0, y_T, T) &= \frac{p_s^Y(y_0, y_s)\, p_{T-s}^Y(y_s, y_T)}{p_T^Y(y_0, y_T)} \\ &= \frac{H_s(y_0,y_s)H_{T-s}(y_s,y_T)}{H_T(y_0,y_T)} \times \frac{n_s(y_0,y_s)\, n_{T-s}(y_s,y_T)}{n_T(y_0,y_T)} \\ &= \frac{H_s(y_0,y_s)H_{T-s}(y_s,y_T)}{H_T(y_0,y_T)} \times p_s^W(y_s; y_0, y_T, T) \end{aligned}$$

Thus we are able to write the bridge density of Y in terms of the bridge density of a Brownian motion, namely the density of a Brownian bridge at a given time. Based on this decomposition it seems natural to use a Brownian bridge density as the proposal density in an acceptance-rejection sampling scheme. Under this premise the acceptance ratio for acceptance-rejection sampling is the ratio the two

conditional distributions which is given as,

$$\begin{aligned}\frac{p_s^Y(y_s;y_0,y_T,T)}{p_s^W(y_s;y_0,y_T,T)} &= \frac{H_s(y_0,y_s)H_{T-s}(y_s,y_T)}{H_T(y_0,y_T)} \\ &\propto .H_s(y_0,y_s)H_{T-s}(y_s,y_T) \\ &= E\left[\exp\left\{-\frac{1}{2}\int_0^T (a^2+a')(W_z)dz\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right]\end{aligned} \tag{3.1}$$

For acceptance-rejection sampling to work with Brownian bridge candidates this ratio must be bounded. Note that if a^2+a' is bounded below then by properties of expectations and of the exponential function, (3.1) is bounded. To see this suppose there exists some constant c, which may depend on y_0, y_T but not on y_s, such that $a^2+a'+c \geq 0$. Then $\exp\left\{-\frac{1}{2}\int_0^T (a^2+a'+c)(W_z)dz\right\} \leq 1$ $a.s.$, and hence (3.1) is less than or equal to unity. Thus, if we accept a Brownian imputed value of $W_s = y_s$ with probability proportional to (3.1) the accepted variate is distributed according to the bridge density of Y corresponding to time s. This ultimately requires one to generate independent indicator random variables with probability proportional to (3.1). There are several ways to do this, some involving more computation than others. For example, suppose I generate an exponential random variable Z with mean $1/2$ independent of the Brownian motion. Then we have that

$$\begin{aligned}&P\left[Z > \int_0^T (a^2+a')(W_z)dz + c|W_0=y_0,W_s=y_s,W_T=y_T\right] \\ &= E\left[\exp\left\{-\frac{1}{2}\int_0^T (a^2+a')(W_z)dz - \frac{1}{2}c\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right] \\ &= e^{-c/2}E\left[\exp\left\{-\frac{1}{2}\int_0^T (a^2+a')(W_z)dz\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right] \\ &\propto E\left[\exp\left\{-\frac{1}{2}\int_0^T (a^2+a')(W_z)dz\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right]\end{aligned}$$

In this case, the decision variate in the acceptance sampling algorithm is the indicator of the form: $1\left\{Z > \int_0^T (a^2 + a')(W_z)dz + c\right\}$. Such a method requires that at each simulation one can determine whether or not $Z > \int_0^T (a^2 + a')(W_z)dz + c$, which is feasible since this is a Riemann integral with a continuous integrand (with appropriate conditions present of course). This can only be done analytically in some special cases. Ideally, we would like a method for generating the appropriate decision indicator without appealing to numerical integration. Suppose there exist constants c and d such that

$$d \geq h(w) = \frac{a^2(w) + a^{'}(w)}{2} + c \geq 0,$$

for all values w. Suppose we first simulate $W_s^{(x,y)}$ $(W_s|W_0 = x, W_T = y)$ according to a Brownian bridge, then we generate a non-homogeneous Poisson process $N(T)$ with intensity $h\left(W_u^{(x,y)}\right)$, $u \in [0,T]$. This can be done by generating event times τ_i corresponding to a homogenous Poisson process N^* with intensity d and thinning the process by accepting those events at times τ_i with probability $h\left(W_{\tau_i}^{(x,y)}\right)/d$. Simulation of the $W_{\tau_i}^{(x,y)}$ requires Brownian bridge interpolation since these values must be generated conditionally on the endpoints and the value W_s. Then we have that

$$\begin{aligned} &P\left[N(T) = 0 | \sigma\left(W_t, t \in [0,T]\right)\right] \\ &= \exp\left\{-\int_0^T h(W_s)ds\right\} = e^{-cT}\exp\left\{-\frac{1}{2}\int_0^T (a^2 + a^{'})(W_s)ds\right\}. \end{aligned}$$

From this we can conclude that

$$
\begin{aligned}
&P\left[N(T)=0|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&=E\left[I_{[N(T)=0]}|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&=E\left[E\left[I_{[N(T)=0]}|\sigma\left(W_t,t\geq 0\right)\right]|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&=E\left[P\left[N(T)=0|\sigma\left(W_t,t\geq 0\right)\right]|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&=E\left[e^{-cT}\exp\left\{-\frac{1}{2}\int_0^T(a^2+a^{'})(W_s)ds\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&=e^{-cT}E\left[\exp\left\{-\frac{1}{2}\int_0^T(a^2+a^{'})(W_s)ds\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right] \\
&\propto E\left[\exp\left\{-\frac{1}{2}\int_0^T(a^2+a')(W_z)dz\right\}|W_0=y_0,W_s=y_s,W_T=y_T\right].
\end{aligned}
$$

Recognizing (3.1) as a specific Poisson process probability gives an easy and exact way of constructing a practical decision indicator. This comes at the cost of requiring the appropriate boundedness conditions on a^2+a'. An algorithm for simulating $Y_s|Y_0=y_0,Y_T=y_T$ based on this discussion is given as:

1. Simulate y_s from $W_s^{(x,y)}$ or $W_s|W_0=y_0,W_T=y_T$ (Brownian bridge interpolation).

2. Simulate a path of N^* over the interval $[0,T]$. This results in time points $0\leq\tau_1<\tau_2<...<\tau_n\leq T$ corresponding to the Poisson event arrivals. This can be done in several ways. For example, the time between events are known to be independent exponential random variables with mean $1/d$.

3. Independently of N^* simulate a standard Brownian bridge at times $\tau_1<\tau_2<...<\tau_n$ conditional on y_0, y_s and y_T. These points can be generated in any

order. For example, this can be done sequentially by simulating a standard Brownian motion B at times $\tau_1 < \tau_2 < ... < \tau_n \leq T$ and transforming its path according to the recipe

$$\begin{aligned} W_{\tau_1}^{(x,y)} &= x + (y-x)\frac{\tau_1}{T} + B_{\tau_1} - \frac{\tau_1}{T}B_T, \\ W_{\tau_i}^{(x,y)} &= W_{\tau_{i-1}}^{(x,y)} + (y - W_{\tau_{i-1}}^{(x,y)})\frac{\tau_i - \tau_{i-1}}{T - \tau_{i-1}} \\ &\quad + (B_{\tau_i} - B_{\tau_{i-1}}) - \frac{\tau_i - \tau_{i-1}}{T - \tau_{i-1}}(B_T - B_{\tau_{i-1}}), \end{aligned}$$

for $i = 2, .., n$.

4. Generate independently of all of the above processes n uniform random numbers over the interval $[0, 1]$, label these $u_1, ..., u_n$. If $u_i > h(W_{\tau_i})/d$ for all $i = 1, ..., n$ we can reject the points $\tau_1 < \tau_2 < ... < \tau_n$ as events in the non homogeneous Poisson process determining $N(T) = 0$. In this case we accept the simulated skeleton path $(0, x)$, $(\tau_1, W_{\tau_1}), ..., (\tau_n, W_{\tau_n})$, (T, y) as a path of the process $Y_T|Y_0 = y_0, Y_T = y_T$ over $[0, T]$. On the other hand, if for some i we have that $u_i \leq h(W_{\tau_i})/d$ then τ_i is accepted as a point of the non homogeneous Poisson process and so $N(T) > 0$. In this case we reject the simulated Brownian path and start again by returning to step 1.

We have discussed this simulation technique in the context of imputation or simulation conditional on neighbouring endpoints. This method is quite general and also applies to forward simulation of diffusion processes. Furthermore, we will see that this approach can be stated in more generality. We have discussed imputation at a single time point, however, the method actually accepts an entire bridge path of a diffusion as we will see in the next section.

3.1.2 Pathwise Imputation Approach

The pathwise imputation is a stochastic process analogue of acceptance-rejection sampling. To describe the pathwise method we begin by introducing some notation. Denote by $C[0,T]$ the space of $\Re$ valued continuous functions over the interval $[0,T]$. In other words, for all $g \in C[0,T]$, $g : [0,T] \longrightarrow \Re$ is a continuous function. Denote by $\mathcal{B}$ the Borel sigma algebra corresponding to $C[0,T]$ associated with the "sup" norm described in Billingsley (1968). For a scalar valued process Y over $[0,T]$ let P_Y denote the probability measure on $(C[0,T],\mathcal{B})$ induced by the sample paths of Y. For the diffusion Y over $[0,T]$ we denote by $Y^{(x,y)}$ the conditional process of Y given the endpoints x and y, ie. $Y_t^{(x,y)} = Y_t | Y_0 = x, Y_T = y$. Define $W^{(x,y)}$ similarly for a standard Brownian motion W.

THEOREM 2. *Consider a diffusion Y satisfying the above conditions. Suppose that the drift function a is differentiable and satisfies the Novikov condition (see Steele 2000, p.225). Let $p_T(x,y)$ and $n_T(x,y)$ denote the transition densities of Y and W, respectively ($n_T(x,y)$ is the Normal density with mean x and variance T). The Radon Nikodym derivative $\frac{dP_{Y^{(x,y)}}}{dP_{W^{(x,y)}}} : C[0,T] \longrightarrow \Re$ is given by*

$$\frac{dP_{Y^{(x,y)}}}{dP_{W^{(x,y)}}}(\omega) = \frac{n_T(x,y)}{p_T(x,y)} \exp\left[A(y) - A(x)\right] \exp\left[-\frac{1}{2}\int_0^T \left(a^2 + a'\right)(\omega_u)\, du\right].$$

PROOF.

Elements of this proofs are based on the proof of the Dacunha-Castelle and Florens-Zmirou result given in the previous section whose proof is included in the appendix. Girsanov's theorem gives the density of the measure P_Y with respect to

the Wiener measure P_W:

$$\frac{dP_Y}{dP_W}(\omega) = \exp\left[\int_0^T a(\omega_u)\, d\omega_u - \frac{1}{2}\int_0^T a(\omega_u)^2\, du\right].$$

By Ito's lemma we have that

$$A(\omega_T) = A(\omega_0) + \int_0^T a(\omega_u)\, d\omega_u + \frac{1}{2}\int_0^T a'(\omega_u)\, dt$$

Thus,

$$\int_0^T a(\omega_u)\, d\omega_u = A(\omega_T) - A(\omega_0) - \frac{1}{2}\int_0^T a'(\omega_u)\, du.$$

Putting this into the formula for $\frac{dP_Y}{dP_W}(\omega)$ gives

$$\frac{dP_Y}{dP_W}(\omega) = \exp\left[A(\omega_T) - A(\omega_0) - \frac{1}{2}\int_0^T (a' + a^2)(\omega_u)\, du\right].$$

The density on the sample paths for the conditional process $Y^{(x,y)}$ is given by $dP_{Y^{(x,y)}} = p_T(x,y)^{-1} dP_Y$, where $p_T(x,y)$ is the transition density of Y. The analogous sample path density of $W^{(x,y)}$ is $dP_{W^{(x,y)}} = n_T(x,y)^{-1} dP_W$ where $n_T(x,y)$ is the normal density with mean x and variance T. Thus, we can conclude that

$$\begin{aligned}
\frac{dP_{Y^{(x,y)}}}{dP_{W^{(x,y)}}}(\omega) &= \frac{n_T(x,y)}{p_T(x,y)}\frac{dP_Y}{dP_W}(\omega) \\
&= \frac{n_T(x,y)}{p_T(x,y)} \exp\left[A(y) - A(x) - \frac{1}{2}\int_0^T (a' + a^2)(\omega_u)\, du\right] \\
&= \frac{n_T(x,y)}{p_T(x,y)} \exp\left[A(y) - A(x)\right] \exp\left[-\frac{1}{2}\int_0^T (a' + a^2)(\omega_u)\, du\right]. QED
\end{aligned}$$

In general we will not know the transition density of Y. However we can always write

$$\frac{dP_{Y^{(x,y)}}}{dP_{W^{(x,y)}}}(\omega) \propto \exp\left[-\frac{1}{2}\int_0^T (a^2 + a')(\omega_u)\, du\right].$$

The form of the density given above suggests an acceptance simulation scheme using Brownian paths as proposal samples. Knowing the ratio of the densities up to a constant of proportionality is sufficient to perform acceptance sampling so long as the above ratio is bounded. A sufficient condition for this is that the above integrand is bounded below. Suppose there exists $c > 0$ such that $\frac{a^2+a'}{2} + c > 0$, then we can conclude that

$$\exp\left[-\frac{1}{2}\int_0^T (a^2 + a' + 2c)(\omega_u)\, du\right] = \exp(-cT)\exp\left[-\frac{1}{2}\int_0^T (a^2 + a')(\omega_u)\, du\right] \leq 1.$$

Hence,

$$\frac{dP_{Y^{(x,y)}}}{dP_{W^{(x,y)}}}(\omega) \propto \exp(-cT)\exp\left[-\frac{1}{2}\int_0^T (a^2 + a')(\omega_u)\, du\right] \leq 1. \tag{3.2}$$

An acceptance sampling algorithm would consist of simulating a continuous path according to a Brownian bridge process, $\omega = (\omega_s)_{T \geq s \geq 0} \sim P_{W^{(x,y)}}$, and accepting this path with probability (3.2). Of course it is impossible to sample a path in continuous time. In practice, sample paths are simulated on a finite time grid. As a consequence of this, calculating the integral in the exponent of the Girsanov density in finite calculations is inexact. This makes it difficult to generate events with probability (3.2). Constructing a feasible acceptance-rejection algorithm here requires the ability to generate the appropriate indicator events with finite knowledge of the path ω. If this is possible, accepting a path on the basis of finite information

about it, then any path from a Brownian bridge agreeing with the accepted points can also be accepted. In certain cases this is indeed possible. Suppose further that the integrand $\frac{a^2+a'}{2}$ is also bounded above, so that in total we require constants $c, d > 0$ such that

$$d > \frac{a^2 + a'}{2} + c > 0.$$

Let N be a non homogeneous Poisson process with intensity $h(\omega) = \left(\frac{a^2+a'}{2}\right)(\omega) + c$ where the path ω is drawn from $P_{W^{(x,y)}}$. Notice that

$$P\left[N(T) = 0|\omega\right] = \exp\left\{-\int_0^t h(\omega_s)ds\right\} = \exp\left(-cT\right)\exp\left[-\frac{1}{2}\int_0^T \left(a^2 + a'\right)(\omega_u)\,du\right].$$

As was shown in the previous section for a given path $\omega \sim P_{W^{(x,y)}}$ it is possible to determine that the event $\{N(t) = 0\}$ has occurred with finite knowledge of ω. Doing so was achieved through the method of thinning a homogeneous Poisson process. Let N^* be a homogeneous Poisson process with constant intensity d. We can produce a realization of N over $[0, T]$ by first generating a path of N^* and accepting the points of N^* which occur at times say $0 \leq \tau_1 < \tau_2 < ... < \tau_n \leq T$ with respective probabilities $h(\omega_{\tau_1})/d, ..., h(\omega_{\tau_n})/d$ (see Devroye 1986 for a description of thinning). The points of the path which must be sampled to determine acceptance are completely determined by the homogeneous Poisson process. The algorithm is almost identical to that given in the previous section and is given as follows:

1. Simulate a path of N^* over the interval $[0, T]$. This results in time points $0 \leq \tau_1 < \tau_2 < ... < \tau_n \leq T$ corresponding to the Poisson event arrivals.

2. Independently of N^* simulate a standard Brownian bridge $\omega \sim P_{W^{(x,y)}}$ at times $\tau_1 < \tau_2 < ... < \tau_n$ conditional on $\omega_0 = y_0$ and $\omega_T = y_T$.

3. Thin N^* to obtain N. If $N(T) = 0$ we accept the simulated skeleton path $(0, x), (\tau_1, \omega_{\tau_1}), ..., (\tau_n, \omega_{\tau_n}), (T, y)$ as a path of the process $Y^{(x,y)}$ over $[0, T]$. Otherwise return to step 1.

The details of the algorithm regarding thinning and Brownian bridge interpolation are given in the previous section. Similar to the algorithm given in Beskos and Roberts (2005), this algorithm generates a skeleton path $(0, x)$, $(\tau_1, \omega_{\tau_1}), ...,$ $(\tau_n, \omega_{\tau_n})$, (T, y) of $Y^{(x,y)}$ over $[0, T]$. The skeleton paths generated there are done so in the context of forward simulation and are produced using a different construction. Similar independent results have been subsequently obtained by Beskos et. al. (2006b). In the context of forward simulation they also develop an acceptance sampling algorithm which exploits the use of an auxiliary Poisson process. This path was accepted with respect to the pathwise density and can thus be filled in with a Brownian bridge at any other time points of interest without destroying the simulation. The only points of the Brownian path that contribute to determining acceptance are those determined by the process N^*. For example, if at step 2 in the algorithm we augment the sampled Brownian bridge path with additional points of interest the decision step 3 is not affected in any way. Thus, not only are we able to accept a single imputed value or even a finite set of imputed values but in fact the whole sample path of *any Brownian bridge which agrees with these points is accepted*. These randomly spaced discrete points, upon acceptance, provide the skeleton of an acceptable path for the diffusion whose intermediate values can be filled in by simple Brownian Bridge interpolation! Thus, imputing a value of say

Y_s can be done using Brownian bridge interpolation after accepting the path. This is the only difference between this algorithm and that given in the previous section. From the pathwise perspective we see that imputation at specific time points of interest can be deferred until path acceptance has been determined. Pathwise sampling has implications for pricing path dependent derivatives, such as barrier type options, since the computation of hitting time distributions for quite general diffusions is reduced to well known calculations involving Brownian Bridges and the distribution of its extremes.

3.1.3 Forward Simulation

It is possible to modify the conditional simulation approach to handle forward simulation, ie. simulation of a path $\omega \sim P_Y$ where only the initial value $Y_0 = x$ is known. The modification is based on a change of measure originally introduced by Beskos and Roberts (2005) in the context of a one dimensional biased Brownian motion; a Brownian Motion conditional on the its terminal value which is chosen according to an arbitrary density function.

PROPOSITION. Consider the process $\overline{W}$ which is defined to be a standard Brownian motion conditional on $W_T \sim g$. Then $P_{\overline{W}}$ is equivalent to P_W and furthermore

$$\frac{dP_{\overline{W}}}{dP_W}(\omega) = \frac{g(\omega_T)}{(2\pi T)^{-1/2} \exp\left(-\frac{\omega_T^2}{2T}\right)}.$$

PROOF. See Beskos and Roberts.

With this result in place we can write

$$\frac{dP_Y}{dP_{\overline{W}}}(\omega) = \frac{dP_Y}{dP_W}\frac{dP_W}{dP_{\overline{W}}}(\omega) \propto \exp\left[A(\omega_T) - \frac{\omega_T^2}{2T} + \int_0^T \left(a^2 + a'\right)(\omega_u)\, du\right] / g(\omega_T)$$

A crucial choice of the density g simplifies the above Radon Nikodym derivative leaving us only with the exponential of a Riemann integral, take

$$g(u) \propto \exp\left(A(u) - \frac{u^2}{2T}\right), \tag{3.3}$$

so long as $\int_{\Re} \exp\left(A(u) - \frac{u^2}{2T}\right) du < \infty$. Combining these results we have that

$$\frac{dP_Y}{dP_{\overline{W}}}(\omega) \propto \exp\left[-\int_0^T \left(a^2 + a'\right)(\omega_u)\, du\right].$$

We can use the conditional simulation algorithm of the previous section with the slight modification that the endpoint is initially chosen according to the density g. This can usually be done efficiently via acceptance sampling by finding a dominating density for g. See Beskos and Roberts (2005) for a full description of the forward simulation algorithm.

3.2 Imputation for a Class of Multivariate Diffusions

The multivariate problem is examined via transition densities keeping in mind that the pathwise approach is almost identical and equally powerful. Imputation for

multivariate diffusions is a problem of growing importance in the literature, see for example Eraker (2001) who provides a Bayesian analysis of diffusion models using MCMC techniques and the IP (Impute-Posterior) algorithm. The problem has also been well studied by Elerian et al. (2001). Imputation in both of these papers relies on augmenting one's data with points intermittently chosen between the observations, enough points to justify an Euler approximation, so that a Gibbs sampler (or Metropolis-Hastings algorithm) can be employed to bridge successive values of the process. More points in a given interval obviously increases the computing time, but renders the Euler approximation used for the imputations more accurate. Thus the econometrician must (purely subjectively) strike a compromise between excessive computational burden and the degree of approximation. Moreover, as one increases the number of augmented data points, the convergence of the estimation scheme slows down in the limit (as the number of augmented points approaches infinity). Roberts and Stramer (2001) show that these types of algorithms may produce a reducible Markov chain, making convergence impossible. They offer a solution to this problem by considering an appropriate transformation which alters the dominating probability measure, causing irreducibility of the resulting Markov chain.

Unlike the above mentioned methods, we wish a precise imputation scheme in the multivariate setting without introducing extra time points of missing values. This was achieved in the previous section with Normally distributed data, an easier problem due to the simple form of the conditional distributions. For a diffusion, the imputation method in one dimension is based on the convenient decomposition of the transition density, given as the product of the normal density and a conditional

expectation of a functional of a Brownian bridge process. Since this is the exact form of the transition density, our imputations are from the correct distribution not from an Euler approximation to the conditional distributions, and without the necessity of introducing artificial time points to improve an Euler approximation. In order to extend this method to multivariate diffusions we need a result similar to the univariate result of Dacunha-Castelle and Florens-Zmirou (1986) decomposing the transition density of a multivariate diffusion with identity diffusion matrix. The proof of the one dimensional case relies mainly on the Girsanov theorem and a simple application of Ito's lemma, with some additional assumptions we can obtain a multivariate analogue of this result in much the same way as the original paper. To the best of our knowledge this result is new.

LEMMA *Consider a d-dimensional diffusion process of the form*

$$dY_t = \mu_\theta\left(Y_t\right)dt + I_d dW_t,$$

where $\mu_\theta : \Re^d \to \Re^d$ *is a vector drift function with scalar component functions* $\mu_k : \Re^d \to \Re$, $k = 1, ..., d$, I_d *is the d-dimensional identity diffusion matrix,* W *is a standard d-dimensional Wiener process and* θ *a vector of parameters. Suppose the vector field defined by* μ *is conservative with potential function* $G \in C^2$, *ie.* $\nabla G = \mu$ *and* G *has continuous second order partial derivatives. Let* $x \in \Re^d$. *Let* B *be a standard d-dimensional Brownian bridge. For each* $k = 1, ..., d$ *denote* $b_k = -\frac{1}{2}\left(\mu_k^2 + \frac{\partial \mu_k}{\partial x_k}\right)$. *Suppose* $|b_k(x)| = O\left(|x|^2\right)$ *for* $|x| \to \infty$, $k = 1, ..., d$. *Then*

the density of $Y_T|Y_0 = x$ *is of the form*

$$p_T(x,y) = (2\pi T)^{-\frac{d}{2}} \exp\left(-\frac{1}{2T}(y-x)'(y-x) + G(y) - G(x)\right) \times$$
$$E\left[\exp\left(T\sum_{k=1}^{d}\int_0^1 b_k\left(z_u(x,y) + \sqrt{T}B_u\right)du\right)\right],$$

where $z_u(x,y) = (1-u)x + uy$.

PROOF. The proof of this results parallels the one dimensional case. For a diffusion process Z denote by μ_Z^x the measure induced by Z when $Z_0 = x$. Girsanov's theorem gives the density L_T^x of the measure μ_Y^x with respect to the Wiener measure μ_W^x:

$$L_T^x = \frac{d\mu_Y^x}{d\mu_W^x} = \exp\left(\int_0^T \mu(Y_u)\cdot dY_u - \frac{1}{2}\int_0^T \|\mu(Y_u)\|^2 du\right)$$
$$= \exp\left(\sum_{k=1}^{d}\int_0^T \mu_k(Y_u)\, dY_u^k - \frac{1}{2}\sum_{k=1}^{d}\int_0^T \mu_k(Y_u)^2 du\right).$$

Let $\phi : \Re^d \to \Re$ be a Borel measurable bounded function, then we have

$$E_{\mu_Y^x}[\phi(Y_T)] = E_{\mu_W^x}[\phi(Y_T)L_T^x] = E_{\mu_W^x}[E[\phi(Y_T)L_T^x|Y_T]] = E_{\mu_W^x}\left[\phi(Y_T)E_{\mu_W^x}[L_T^x|Y_T]\right]$$
$$= \int \phi(y)E_{\mu_W^x}[L_T^x|Y_T = y](2\pi T)^{-\frac{d}{2}}\exp\left(-\frac{1}{2T}(y-x)'(y-x)\right)dy.$$

From this equation we see that the transition density of $Y_T|Y_0 = x$ is given by

$$p_T(x,y) = \Lambda_T(x,y)(2\pi T)^{-\frac{d}{2}}\exp\left(-\frac{1}{2T}(y-x)'(y-x)\right).$$

where $\Lambda_T(x,y) = E_{\mu_W^x}[L_T^x|Y_T = y]$. It now remains to find an expression for the

expectation term $\Lambda_T(x,y)$. By Ito's lemma we have that

$$\begin{aligned} G(Y_T) &= G(x) + \sum_{k=1}^{d} \int_0^T \frac{\partial G}{\partial x_k}(Y_u)\, dY_u^k + \frac{1}{2} \sum_{k,j=1}^{d} \int_0^T \frac{\partial^2 G}{\partial x_k \partial x_j}(Y_u)\, dY_u^k dY_u^j \\ &= G(x) + \sum_{k=1}^{d} \int_0^T \frac{\partial G}{\partial x_k}(Y_u)\, dY_u^k + \frac{1}{2} \sum_{k=1}^{d} \int_0^T \frac{\partial^2 G}{\partial x_k^2}(Y_u)\, du \\ &= G(x) + \sum_{k=1}^{d} \int_0^T \mu_k(Y_u)\, dY_u^k + \frac{1}{2} \sum_{k=1}^{d} \int_0^T \frac{\partial \mu_k}{\partial x_k}(Y_u)\, du. \end{aligned}$$

Thus,

$$\sum_{k=1}^{d} \int_0^T \mu_k(Y_u)\, dY_u^k = G(Y_T) - G(x) - \frac{1}{2} \sum_{k=1}^{d} \int_0^T \frac{\partial \mu_k}{\partial x_k}(Y_u)\, du,$$

putting this into the formula for L_T^x yields

$$\begin{aligned} L_T^x &= \exp\left[G(Y_T) - G(x) - \frac{1}{2} \sum_{k=1}^{d} \int_0^T \frac{\partial \mu_k}{\partial x_k}(Y_u)\, du - \frac{1}{2} \sum_{k=1}^{d} \int_0^T \mu_k(Y_u)^2\, du \right] \\ &= \exp\left[G(Y_T) - G(x) - \frac{1}{2} \sum_{k=1}^{d} \int_0^T \left(\frac{\partial \mu_k}{\partial x_k}(Y_u) + \mu_k(Y_u)^2 \right) du \right] \\ &= \exp\left[G(Y_T) - G(x) + \sum_{k=1}^{d} \int_0^T b_k(Y_u)\, du \right]. \end{aligned}$$

This gives us an expression for $\Lambda_T(x,y)$ involving no stochastic integrators, we have

$$\begin{aligned} \Lambda_T(x,y) &= E_{\mu_W^x}\left[\exp\left(G(Y_T) - G(x) + \sum_{k=1}^{d} \int_0^T b_k(Y_u)\, du \right) | Y_T = y \right] \\ &= \exp(G(y) - G(x)) \times E_{\mu_W^x}\left[\exp\left(\sum_{k=1}^{d} \int_0^T b_k(Y_u)\, du \right) | Y_T = y \right]. \end{aligned}$$

As in the one dimensional case define a change of variable by $v = \frac{u}{T}$, let $B_v =$

$Y_v^* - vY_1^*$ where $Y_v^* = \frac{1}{\sqrt{T}}\left(Y_{Tv} - x\right)$ and for each $k = 1, ..., d$ we have that

$$\int_0^T b_k\left(Y_u\right) du = T \int_0^1 b_k\left(z_v\left(x, y\right) + \sqrt{T} B_v\right) dv.$$

B is a standard Brownian bridge independent of Y_T under μ_W^x, so we have that

$$\Lambda_T\left(x, y\right) = \exp\left(G\left(y\right) - G\left(x\right)\right) \times E\left[T \exp\left(\sum_{k=1}^{d} \int_0^1 b_k\left(z_v\left(x, y\right) + \sqrt{T} B_v\right) dv\right)\right]$$

We can use this result to devise an accept-reject scheme for simulating multivariate diffusions (with identity diffusion coefficient) conditional on vector endpoints in much the same way as the one dimensional case which was outlined in the previous section. We wish a similar result for a general multivariate diffusion process with arbitrary diffusion coefficient. As in the univariate case, we need to transform variables to obtain a diffusion process whose diffusion coefficient is the identity matrix. Ait-Sahalia (2004) discusses the possibility of such transformations, and characterizes the diffusions that permit such a transformation. He calls such diffusions 'reducible', and gives a precise definition for such a class of processes:

DEFINITION. *The diffusion X is said to be reducible to unit diffusion if and only if there exists a one-to-one transformation of the diffusion X into a diffusion Y whose diffusion matrix is the identity matrix. That is, there exists an invertible function $F\left(x\right)$ such that $Y_t = F\left(X_t\right)$ satisfies a stochastic differential equation of the form*

$$dY_t = \mu_Y\left(Y_t\right) dt + dW_t,$$

where W is a standard Wiener process the same dimension as X.

Consider an arbitrary d-dimensional diffusion process of the form

$$dX_t = \mu(X_t)\,dt + \sigma(X_t)\,dW_t,$$

where $\mu : \Re^d \to \Re^d$ is a vector drift function, $\sigma : \Re^d \to \Re^d \times \Re^d$ the diffusion matrix, W is a standard d-dimensional Wiener process. Ait-Sahalia gives the necessary and sufficient conditions for the reducibility of X as

$$X \text{ is reducible} \Leftrightarrow \sum_{l=1}^{d} \frac{\partial \sigma_{ik}(x)}{\partial x_l}\sigma_{lj}(x) = \sum_{l=1}^{d} \frac{\partial \sigma_{ij}(x)}{\partial x_l}\sigma_{lk}(x),$$

for each triplet $(i,j,k) = 1,..,d$ such that $k > j$. In the case that σ is a nonsingular matrix, the condition simplifies to

$$\frac{\partial \sigma_{ij}^{-1}(x)}{\partial x_k} = \frac{\partial \sigma_{ik}^{-1}(x)}{\partial x_j},$$

where $\sigma_{ij}^{-1} := [\sigma^{-1}]_{ij}$. For the bivariate case this condition becomes

$$\frac{\partial \sigma_{11}^{-1}(x)}{\partial x_2} - \frac{\partial \sigma_{12}^{-1}(x)}{\partial x_1} = \frac{\partial \sigma_{22}^{-1}(x)}{\partial x_1} - \frac{\partial \sigma_{21}^{-1}(x)}{\partial x_2} = 0.$$

For simplicity, we restrict ourselves to the case of nonsingular σ. If X is reducible, then by Ito's lemma the appropriate transformation $F : \Re^d \to \Re^d$ must satisfy the differential equation

$$\frac{\partial F(x)}{\partial x} \equiv [\nabla F_k]_{k=1}^{d} = \sigma^{-1}(x),$$

where $\left[\frac{\partial F(x)}{\partial x}\right]_{ij} = \frac{\partial F_i(x)}{\partial x_j}$. Let $Y_t = F(X_t)$, we have that

$$dY_t = A(Y_t)\,dt + dW_t,$$

where the component functions of $A(y) = (A_k(y))_{k=1}^{d}$ are given by

$$A_k(y) = \nabla F_k\left(F^{-1}(y)\right)\mu\left(F^{-1}(y)\right) + \frac{1}{2}\sum_i \sum_j \partial_{ij}^2 F_k\left(F^{-1}(y)\right) v_{ij}\left(F^{-1}(y)\right),$$

where $v(x) = \sigma(x)\sigma(x)'$ and $\partial_{ij}^2 = \frac{\partial^2}{\partial y_i \partial y_j}$. This can be seen by examining the infinitesimal properties of the process. For small δ Taylor's theorem gives

$$
\begin{aligned}
&var\left(Y_{t+\delta} - Y_t | Y_t\right) \\
&= var\left(F\left(X_{t+\delta}\right) - F\left(X_t\right) | Y_t\right) \approx var\left(\frac{\partial F\left(X_t\right)}{\partial x}\left(X_{t+\delta} - X_t\right) | Y_t\right) \\
&= \frac{\partial F\left(X_t\right)}{\partial x}\frac{\partial F\left(X_t\right)'}{\partial x} var\left(X_{t+\delta} - X_t | Y_t\right) = \sigma^{-1}\left(X_t\right)\sigma^{-1}\left(X_t\right)' var\left(X_{t+\delta} - X_t | Y_t\right) \\
&= \left[\sigma\left(X_t\right)\sigma\left(X_t\right)'\right]^{-1}\sigma\left(X_t\right)\sigma\left(X_t\right)'\delta + o\left(\delta\right) \\
&= I_{d\times d}\delta + o\left(\delta\right).
\end{aligned}
$$

The instantaneous drift can be decomposed similarly

$$
\begin{aligned}
&E\left(Y_{t+\delta}^k - Y_t^k | Y_t\right) \\
&\approx E\left(\nabla F_k\left(X_t\right)\left(X_{t+\delta} - X_t\right) + \frac{1}{2}\left(X_{t+\delta} - X_t\right)'\partial^2 F_k\left(X_t\right)\left(X_{t+\delta} - X_t\right) | Y_t\right) \\
&= \nabla F_k\left(X_t\right) E\left(\left(X_{t+\delta} - X_t\right) | Y_t\right) + \frac{1}{2} E\left(\left(X_{t+\delta} - X_t\right)'\partial^2 F_k\left(X_t\right)\left(X_{t+\delta} - X_t\right) | Y_t\right) \\
&= \nabla F_k\left(X_t\right)\mu\left(X_t\right)\delta + \frac{1}{2} E\left(\sum_i \sum_j \left(X_{t+\delta}^i - X_t^i\right)\left(X_{t+\delta}^j - X_t^j\right)\partial_{i,j}^2 F_k\left(X_t\right) | Y_t\right) \\
&= \nabla F_k\left(X_t\right)\mu\left(X_t\right)\delta + \frac{1}{2}\sum_i \sum_j \partial_{i,j}^2 F_k\left(X_t\right) E\left(\left(X_{t+\delta}^i - X_t^i\right)\left(X_{t+\delta}^j - X_t^j\right) | Y_t\right) \\
&= \left(\nabla F_k\left(X_t\right)\mu\left(X_t\right) + \frac{1}{2}\sum_i \sum_j \partial_{i,j}^2 F_k\left(X_t\right) v_{ij}\left(X_t\right)\right)\delta + o\left(\delta\right).
\end{aligned}
$$

So long as the transformed drift A is a conservative vector field, the problem of simulating $X_T | X_s = x_s, X_0 = x_0$ can be reduced to simulating $Y_T | Y_s = F\left(x_s\right), Y_0 = F\left(x_0\right)$, which is analogous to the one dimensional case. For example, let

$$H_T(x,y) = E\left[\exp\left\{T\sum_{k=1}^{d}\int_0^1 b_k\left(z_u\left(x,y\right)+\sqrt{T}W_u\right)\right\}du\right],$$

with W_z a standard Brownian motion conditional on $W_0 = x, W_T = y$

We are interested in imputing the value Y_s given the values $Y_0 = y_0$ and $Y_T = y_T$ for $0 < s < T$. Note that if W denotes a Brownian motion process with diffusion coefficient equal to the identity, then the conditional probability density of $Y_s|Y_0 = y_0, Y_T = y_T$ is given by

$$\begin{aligned} p_s^Y\left(y_s; y_0, y_T, T\right) &= \frac{p_s^Y\left(y_0, y_s\right) p_{T-s}^Y\left(y_s, y_T\right)}{p_T^Y\left(y_0, y_T\right)} \\ &= \frac{H_s(y_0,y_s)H_{T-s}(y_s,y_T)}{H_T(y_0,y_T)} \times \frac{n_s\left(y_0, y_s\right) n_{T-s}\left(y_s, y_T\right)}{n_T\left(y_0, y_T\right)} \\ &= \frac{H_s(y_0,y_s)H_{T-s}(y_s,y_T)}{H_T(y_0,y_T)} \times p_s^W\left(y_s; y_0, y_T, T\right) \end{aligned}$$

We obtain a method analogous to the one dimensional case. Impute using a standard d-dimensional Brownian bridge and accept the imputation with probability proportional to

$$\begin{aligned} &H_s(y_0,y_s)H_{T-s}(y_s,y_T) \\ &= E\left[\exp\left\{\int_0^1 T\sum_{k=1}^{d} b_k\left(z_s\left(y_0,y_T\right)+\sqrt{T}W_u\right)du\right\}|W_u = w_u, u = 0, s, T\right] \\ &= E\left[\exp\left\{-\int_0^1 h(W_u)du\right\}|W_u = w_u, u = 0, s, T\right], \end{aligned}$$

where

$$w_u = \frac{y_u - z_u\left(y_0, y_T\right)}{\sqrt{T}}, \text{ and}$$
$$h(w) = -T\sum_{k=1}^{d} b_k\left(z_s\left(y_0, y_T\right) + \sqrt{T}w\right).$$

To generate an event with the appropriate probability we can appeal to the results from the one dimensional case. Adopting h as our intensity function we generate a non-homogeneous Poisson process N over $[0,1]$, accepting the imputed value if $N(1) = 0$. Little additional complexity is introduced in the multivariate case. This gives a method for generating bridge variates for quite general multivariate diffusions with identity diffusion matrix. Suppose we only wish to generate certain vector components conditional on the vector endpoints and neighbouring observations at the same time point. For example, for given $1 \leq k \leq d$ we may wish to sample $Y_s^k|Y_s^{-k}, Y_0, Y_T$ or more generally $Y_s^K|Y_s^{-K}, Y_0, Y_T$ where $K \subseteq \{1, ..., d\}$ and $Y_s^K = \left(Y_s^k\right)_{k\in K}$ and $Y_s^{-K} = \left(Y_s^k\right)_{k\in\{1,\ldots,d\}\setminus K}$. In the case of simulating $Y_s^K|Y_s^{-K}, Y_0, Y_T$ we note that the density of this conditional random variable is given by

$$\begin{aligned}
f\left(y_s^K|y_s^{-K}, y_0, y_T\right) &= \frac{p_s^Y\left(y_s; y_0, y_T, T\right)}{f\left(y_s^{-K}|y_0, y_T\right)} \\
&= \frac{p_s^Y\left(y_0, y_s\right) p_{T-s}^Y\left(y_s, y_T\right)}{p_T^Y\left(y_0, y_T\right)} \frac{1}{f\left(y_s^{-K}|y_0, y_T\right)} \\
&= \frac{H_s(y_0, y_s) H_{T-s}(y_s, y_T)}{H_T(y_0, y_T)} p_s^W\left(y_s; y_0, y_T, T\right) \frac{1}{f\left(y_s^{-K}|y_0, y_T\right)} \\
&= \frac{H_s(y_0, y_s) H_{T-s}(y_s, y_T)}{H_T(y_0, y_T)} f^W\left(y_s^K|y_s^{-K}, y_0, y_T\right) \\
&\quad \times f^W\left(y_s^{-K}|y_0, y_T\right) \frac{1}{f\left(y_s^{-K}|y_0, y_T\right)} \\
&= \frac{H_s(y_0, y_s) H_{T-s}(y_s, y_T)}{H_T(y_0, y_T)} \times f^W\left(y_s^K|y_s^{-K}, y_0, y_T\right)
\end{aligned}$$

Thus, to generate $Y_s^K | Y_s^{-K}, Y_0, Y_T$ we first generate a variate with density

$$f^W\left(y_s^K | y_s^{-K}, y_0, y_T\right)$$

and accept this as our imputation with probability proportional to

$$\frac{f\left(y_s^K | y_s^{-K}, y_0, y_T\right)}{f^W\left(y_s^K | y_s^{-K}, y_0, y_T\right)} = \frac{H_s(y_0, y_s) H_{T-s}(y_s, y_T)}{H_T(y_0, y_T)}$$

The procedure is the same as the one-dimensional case: we impute under the assumption the data follows a standard Brownian motion and make a decision to accept the imputation based on the path of a non-homogeneous Poisson process with the appropriate intensity.

The only restriction of the method in the multivariate case is that we require a conservative vector field for the drift function. Complications can arise in calculating the potential of A. Such considerations are model dependent and must be considered on a case by case basis. This leaves a large collection of feasible processes to which the method can be applied. Using these results along with the Gibbs sampler we have a MCMC method which allows one to impute the unobserved asynchronous observations arising from quite general diffusion processes. These methods have many potential applications. For example, similar MCMC analyses to those performed by Eraker (2001) and Elerian et al. (2001) could be done using the imputation methods described in this section without much of the data augmentation outlined in their papers. In credit risk applications where Monte Carlo simulation is used in analyzing structural models, it is common practice to use variance reduction techniques (such as importance sampling) to produce asset paths which cause a default more frequently. For example, see the documentation for the CreditMetrics model. The methods described here can be used to simulate joint firm value asset paths which terminate in a desired region.

3.3 Example: Square Root Diffusion Process (CIR Model)

This process has been used to model fluctuations in the spot rate of interest, stochastic volatility dynamics, in addition to other economic variables. The differential form of the process is given by

$$dr_t = \kappa(\theta - r_t)dt + \sigma\sqrt{r_t}dW_t \tag{3.4}$$

In this case, $\sigma(x) = \sigma\sqrt{x}$, $\mu(x) = \kappa(\theta - x)$, then $s(x) = \int_0^x \frac{1}{\sigma(z)}dz = \frac{2}{\sigma}\sqrt{x}$ so $g(y) = \frac{\sigma^2}{4}y^2$. Then the process defined by $Y_t = s(r_t) = \frac{2}{\sigma}\sqrt{r_t}$ has a diffusion coefficient of 1 and a drift function given by

$$a(y) = \frac{\mu(g(y))}{\sigma(g(y))} - \frac{1}{2}\sigma'(g(y)) = \frac{\kappa(\theta - g(y))}{\sigma\sqrt{g(y)}} - \frac{\sigma}{4}(\frac{\sigma^2}{4}y^2)^{-1/2} \tag{3.5}$$

$$= \frac{\kappa(\theta - \frac{\sigma^2}{4}y^2)}{\sigma\frac{\sigma}{2}y} - \frac{1}{2}y^{-1} = (\frac{2\kappa}{\sigma^2}\theta - \frac{1}{2})y^{-1} - \frac{\kappa}{2}y \quad \text{assuming } Y_t > 0 \tag{3.6}$$

The original process r is known to have non-central chi-squared marginals, the transition density of the process is given by

$$p_t(x,y) = Ke^{-u-v}\left(\frac{v}{u}\right)^{\frac{q}{2}} I_q\left(2\sqrt{uv}\right)$$

where

$$K = \frac{2\kappa}{\sigma^2\left(1 - e^{-\kappa t}\right)}$$

$$u = Kxe^{-\kappa t}$$

$$v = Ky$$

$$q = \frac{2\kappa\theta}{\sigma^2} - 1$$

and I_q is the modified Bessel function of the first kind of order q.

Recall that for the imputation method to be successful, one must obtain constants c and d such that

$$d \geq h(w) = a^2(w) + a^{'}(w) + c \geq 0 \text{ for all } w.$$

In practice the constants c and d can be found using bounds provided by ordinary calculus for the function $a^2 + a'$. In more complicated cases numerical techniques can be used to find the appropriate bounds. In the cases where this function is unbounded then we may truncate the drift function a by determining an appropriate low probability region for the underlying Brownian Bridge. For any given tolerance this truncated region can be determined explicitly by the known distributional properties of the extreme values of a Brownian Bridge. In general this is no easy task, however, in certain circumstances appropriate constants can be chosen with relative ease. When such bounds exist for all w the method produces perfect imputations. For the CIR model the function $a^2 + a'$ has no upper bound. In this case we propose an approximation by restricting the range of w over which we will bound the function. We estimate c and d by using ordinary calculus to find the minimum and maximum values of the function $(a^2 + a')$ over a region in which the process resides with probability close to 1. For a given small tolerance $\varepsilon > 0$, we choose values $0 < \delta < K < \infty$ so that the probability that Y_s leaves the interval (δ, K) for some $0 < s < t$ is less than ε and then use

$$d = \sup\{a^2(w) + a^{'}(w) + c; w \in (\delta, K)\}.$$

Example: **Generating a bridge path from the CIR process**. Suppose we wish to generate $r_{0.5}|r_0 = 0.05, r_1 = 0.05$ where the parameter values of the

process are $(\kappa, \theta, \sigma) = (0.2, 0.06, 0.1)$. The transformation of variables to Y gives the equivalent problem of simulating $Y_{0.5}|Y_0 = Y_1 \approx 4.47$. The drift function of the transformed diffusion Y is given explicitly as $a^2(w) + a'(w) = 1.344w^{-2} + 0.01w^2 - 0.38$ which is plotted in Figure 3.1. Since the original process r_t stays positive almost surely for the given parameter values (Cox et. al. 1985) and $Y_t = \frac{2}{\sigma}\sqrt{r_t}$ we only need to consider bounding $a^2(w) + a'(w)$ in some suitable region of positive values of w. Since this function is bounded below and is continuously differentiable for $w > 0$ the value c is easily found in this case using Calculus methods. For a tolerance of say $\varepsilon = 10^{-10}$ we can estimate δ and K by using the distributional properties of the extremes of a Brownian motion. This choice is inexact since we should be choosing bounds based on the Brownian motion with drift, however for the plausible values in this model we find this is sufficient. The essential ingredient is choosing δ small and K large in order to truncate the drift on a low probability region. The error introduced by this truncation is negligible and can be examined by studying the outputs of simulations for different choices of δ and K. For diffusions which emit a bounded drift no truncation is necessary and the simulations are exact.

To generate a value of $Y_{0.5}|Y_0 = Y_1 \approx 4.47$ we begin by simulating a standard Poisson process with intensity $d/2$ and subsequently interpolate Y at the time instances of the Poisson arrivals and at the time point of interest $t = 0.5$ using a Brownian bridge conditional on the given endpoints. Suppose the simulated time instances of the Poisson process are given by $\tau_1, ..., \tau_n$. This means a draw from the distribution $Y_{\tau_1}, ..., Y_{\tau_n}, Y_{0.5}|Y_0, Y_1$ is made under the assumption that Y is a Brownian motion. Using these simulated values we thin the Poisson process using

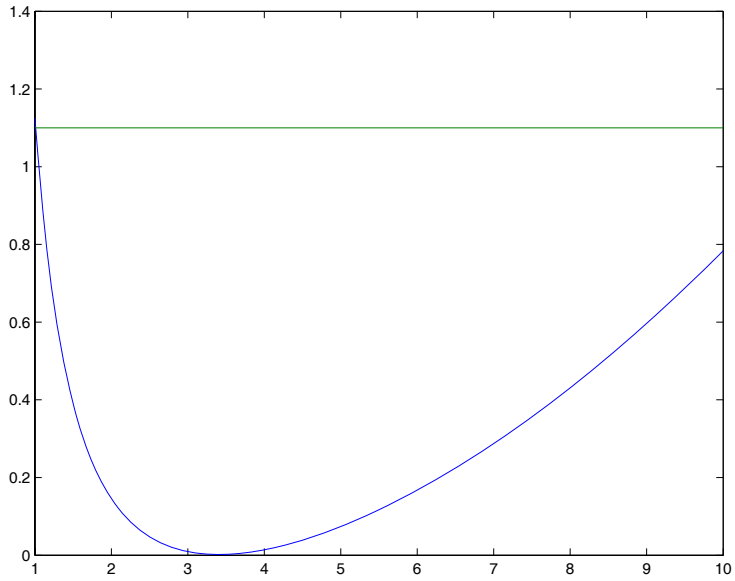

Figure 3.1: (i) Drift of the transformed diffusion shifted by a factor of c (curve) (ii) Estimated bound d over low probability region (flat line)

the intensity function $(a^2 + a' + c)(Y_t)$ by generating uniform random numbers and comparing them to $(a^2+a'+c)/d$. In other words, we simulate $u_1, ..., u_n$ where each $u_i \sim U[0,1]$ independently of each other and all previously generated values. If for some i, $u_i \leq ((a^2 + a')(Y_{\tau_i}) + c)/d$ then the non-homogeneous Poisson process takes value $N(1) > 0$ and we reject the simulated values of Y and begin anew. Otherwise, we can accept the simulated skeleton path of Y: $Y_{\tau_1}, ..., Y_{\tau_n}, Y_{0.5}$.

In a specific instance one arrival occurs in the time interval $[0,1]$ at time $t \approx 0.1$ and the interpolated Brownian bridge at the time points $t = 0.1$ and 0.5 were simulated as 4.67 and 4.31 respectively. A uniform random number was drawn, $u = 0.89$, and compared to: $((a^2 + a')(4.67) + c)/d = 0.03 < 0.89$ indicating that the point is rejected as an arrival for the thinned process. Thus, in this case the realized value of the thinned (non-homogeneous) Poisson process is $N(1) = 0$ and so we can accept the simulated path. This skeleton path transformed back into the r variable is plotted in Figure 3.3. In fact, we can accept any Brownian path agreeing with such points. Figure 3.2 shows the skeleton path augmented with an additional 1000 points equally spaced over $[0,1]$.

Since the true bridge density for the CIR process is known explicitly we can compare this with empirical density constructed by sampling bridge variates using the pathwise method. Figure 3.4 show such a comparison using 100,000 variates. The empirical density matches the true density very closely indicating that the method works quite well.

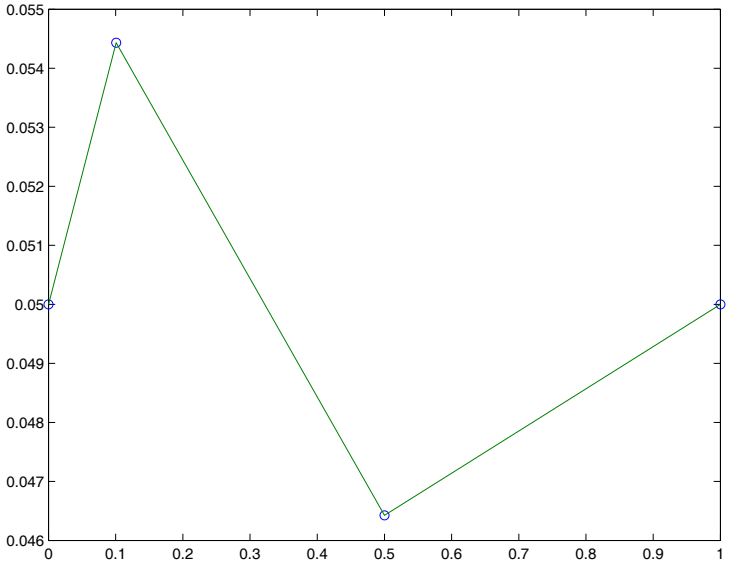

Figure 3.2: Accepted skeleton path.

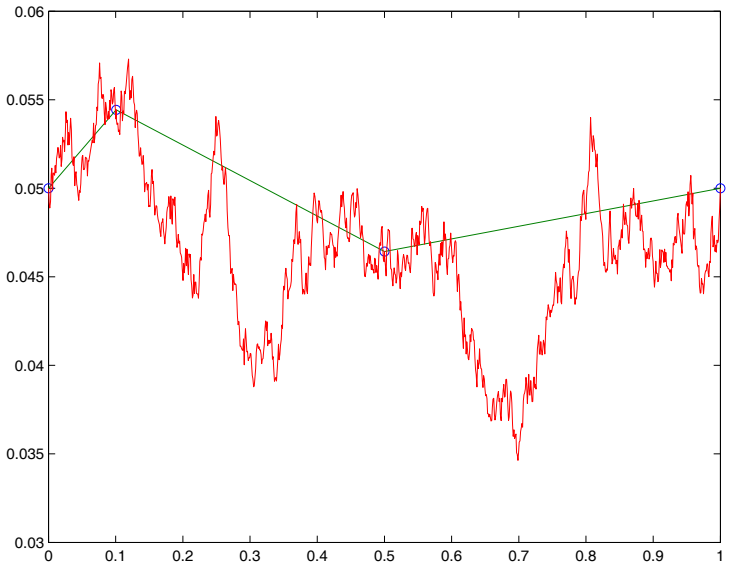

Figure 3.3: Skeleton path augmented with an additional 1000 points using a Brownian bridge.

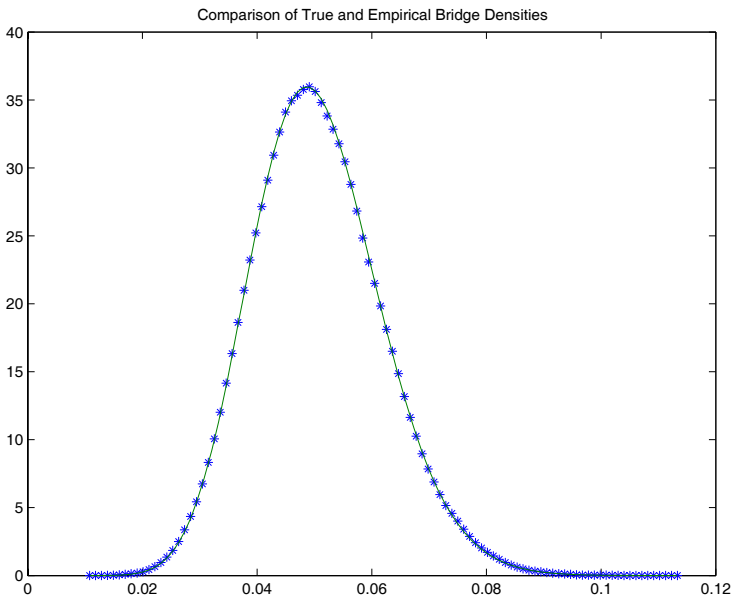

Figure 3.4: Kernal smoothed empirical density (*) superimposed on the true CIR bridge density. 100,000 bridge variates were sampled with the parameter values were $(\kappa, \theta, \sigma) = (0.2, 0.06, 0.1)$, $t = 1$ and $s = 0.5$.

3.3.1 Variance Reduction: Stratifying the Paths of a CIR Process

Assuming that the risk neutral dynamics of the spot interest rate are given as above, one can explicitly derive bond and bond option prices under this model, see Cox, Ingersoll and Ross (1985) for exact pricing formulae. One application of univariate diffusion imputation is variance reduction in Monte Carlo pricing of path dependent derivative securities. This involves stratifying the sample paths of a diffusion so that they pass through pre-specified points at intermediate times of the process, see Glasserman (2000) for a discussion of stratified sampling and variance reduction in finance. This technique is most effective for derivatives whose value depends largely on the stratified values of the process. For example, European style derivatives (such as bond options) are ideal for stratification of the value of the underlying's price at the option's maturity, this is known as terminal stratification. Other securities such as caps and floors have several reset times, in these cases stratifying sampling paths at these dates will also provide significant variance reduction.

Here we show an example of how imputation could be used to implement a stratified sampling procedure for the valuation of bond options in the CIR model. To start, assume we begin at time 0 with spot interest rate r_0 whose risk neutral dynamics are given above. Consider a zero coupon bond maturing at time $S > 0$ and an option to purchase this bond sometime before its maturity, $0 < T < S$, at a strike price of K. To price this bond option using ordinary Monte Carlo we simply simulate forward spot rate paths to time T. If the resultant spot rate produces a time T bond value, $B(r_T, T, S)$, greater than the strike K we record this difference $B(r_T, T, S) - K$, otherwise we record 0. The bond option price is the average

of these recorded values, discounted to time 0 using the sample spot rate paths. Depending on the underlying parameters, this method may produce several paths which contribute nothing to the average because the option expires worthless, ie. $B(r_T, T, S) < K$. These are wasted simulations. The process r_t can be stratified at time T so that we never encounter an unexercised option in our simulations. Note that this will not introduce bias in our pricing estimates as long as we weight our simulations properly (Glasserman 2000).

To implement these ideas, we need to ensure that the observed spot rate at time T is small enough to ensure a time T bond price larger than K. This is achieved by noting the inverse relationship between spot rate and bond prices, and that there exists a critical rate r_c such that if $r_T > r_c$ then the bond price at time T satisfies $B(r_T, T, S) < K$ yielding an unexercised option. Thus, it makes sense to stratify the sample paths of r_t so that they pass through probability equally spaced points in the interval $(0, r_c)$, which is easily done in this case by noting that the distribution of $r_T|r_0$ is noncentral chi square. For a general spot rate model in which the transition function is not explicitly known, one could use the Dacuhna-Castelle and Florens-Zmirou (1986) expansion of the transition density to calculate the time T stratified spot rate values. Our aim is to compare this variance reduced Monte Carlo method to standard simulation and study any efficiency gains. To simplify our calculations at time T we use the known CIR pricing formula to calculate the bond price $B(r_T, T, S)$ in both methods. This practice is internally consistent, and will allow us to isolate the variance reduction achieved by stratifying at time T only. Of course, for a more general spot rate model, we might not have knowledge of such a formula in which case the time T bond price could be calculated by

simulating spot rate paths between T and S. Stratification could be used for this although a bond price depends on the entire path of the spot rate thus reducing the effectiveness of a terminal stratification procedure.

Tables 3.1-3.2 show the results of Monte Carlo tests for bond option pricing in the CIR model. The parameters (κ, θ, σ) are chosen according to those estimated in Chan et al. (1992). The stratified sampling procedure was employed with 10 strata and 50 simulations in each (proportional allocation), producing a total of 500 simulations, the same number used for implementing the ordinary Monte Carlo method. Pathwise simulation in the ordinary Monte Carlo method is done using the known distribution of spot rate increments implied by the CIR model. Glasserman (2000) shows how to generate non central chi-square random variables for this task. For both methods the time interval $(0, T)$ is divided into 50 points. A greater refinement of this interval only increases the accuracy of the pathwise discount factor for each simulation as the distributional properties of the simulated terminal value are not dependent on this choice (unlike Euler based schemes). Tables 3.1-3.2 display simulation results. Each column in the tables contains the estimator, its standard error and the relative efficiencies under each scheme. The CIR parameters used for all tests were taken to be $(\kappa, \theta, \sigma) = (0.2339, 0.0808, 0.0854)$ with an initial spot rate of $r_0 = 0.035$.

Notice that the quality of the ordinary Monte Carlo estimates (measured in terms of both pricing error and variance) degrade as the strike price increases, due to the fact that larger strike prices will generally increase the number of simulated paths leading to an unexercised option. To examine the relative efficiency of two competing Monte Carlo methods (with the same number of simulated paths), one

may examine the ratio of the variances of the respective estimators. If the methods have very different CPU times, or a different number of simulations, this measure should be adjusted to reflect these differences. For example, to account for the CPU times one can multiply the ratio of the variances by the ratio of the running times. In our example, the range of efficiency gains is approximately 132 to 106,091, if we account for CPU running times this range changes to a more modest 11 to 9,324. All of the options tested showed an efficiency gain under the stratified sampling method, especially the more out of the money (higher strike) cases.

One should also consider the observed pricing error, which is not reflected in efficiency measures. Both the crude Monte Carlo estimator and the Stratified estimator are unbiased estimators, however, the stratified estimator is consistently closer to the true price for each option regardless of strike. This consistency is observed across many replications of this experiment. In fact, for several repetitions of this experiment the crude estimator shows relative pricing errors of up to 60% whereas the stratified estimator is consistent across replications. In this example, the relative pricing error for the stratified sampler is less than 1% for all options and expiries with an average relative pricing error of about 0.17% across all options. The ordinary Monte Carlo method shows an average relative pricing error of about 2.31%. In particular, the shorter expiry out of the money option shows a pricing error in excess of 6%.

All of the above measures of performance tell the same story. The stratified estimator of prices is superior to the usual Monte Carlo estimator. Even in cases where both methods produce reasonable prices, the efficiency gains achieved by stratification are evident. These results demonstrate that the imputation method is feasible

Prices×100	Strike - K				
$(T, S) = (0.5,1)$	0.94	0.95	0.96	0.97	0.98
Ordinary	3.831 (0.023)	2.867 (0.023)	1.867 (0.021)	0.920 (0.021)	0.162 (0.0005)
Stratified	3.834 (0.002)	2.853 (0.001)	1.874 (0.001)	0.905 (0.0007)	0.155 (0.0004)
Efficiency: regular time adjusted	199 16	132 11	376 31	1356 115	1674 150
True Price	3.835	2.85	1.873	0.904	0.155

Table 3.1: Bond Option Pricing Results: Shorter Maturity

and effective for the CIR model. It is important to note that the implementation of the stratified sampler was rather crude and could be improved in practice with some effort. For example, using optimal allocation instead of proportional allocation could only increase the observed efficiencies.

3.4 Jump Diffusions

3.4.1 Pathwise Imputation

The imputation scheme for diffusions depends crucially on the form of the Radon-Nikodym derivative of the probability measure induced by the diffusion in question with respect to the Wiener measure. This derivative is given by the Girsanov theorem. The basic ingredients to the simulation algorithm are based on the ability

Prices×100	Strike - K				
(T, S) =(1,2)	0.94	0.95	0.96	0.97	0.98
Ordinary	1.426 (0.045)	0.657 (0.034)	0.207 (0.018)	0.021 (0.005)	0.162 ≈≥(0.0003)
Stratified	1.3679 (0.001)	0.656 (0.001)	0.204 (0.0005)	0.028 (0.0006)	6.0e-004 (0.000002)
Efficiency: regular time adjusted	800 110	1099 177	3848 499	5899 640	≈≥106091 ≈≥9324
True Price	1.369	0.656	0.204	0.028	6.0e-004

Table 3.2: Bond Option Pricing Results: Longer Maturity

to dominate the measure of the diffusion with another measure in which we can simulate from. The Girsanov theorem has many extensions to both infinite activity and finite activity Levy processes. Due to issues of tractability we concentrate on the latter in this section. We begin by stating a Girsanov type result for jump diffusions which we quote from Tankov and Cont (2004).

PROPOSITION: *Let* $(\Omega, \mathcal{F})$ *be a measurable space equipped with filtration* $\{\mathcal{F}_t\}$. *Let* (X_t, P) *and* (X_t, P') *be two Levy processes on* $\Re$ *with characteristic triplets* $(\sigma^2, \upsilon, \gamma)$ *and* $(\sigma'^2, \upsilon', \gamma')$. *Then* $P|_{\mathcal{F}_t}$ *and* $P'|_{\mathcal{F}_t}$ *are equivalent for all* t *(or equivalently for one* $t > 0$*) if and only if the three following conditions are satisfied:*

1. $\sigma' = \sigma$.

2. *The Levy measures are equivalent with* $\int_{-\infty}^{\infty} \left(e^{\phi(x)/2} - 1\right)^2 v(dx) < \infty$, *where*

$\phi(x) = \ln\left(\frac{d\upsilon'}{d\upsilon}\right).$

3. *If* $\sigma = 0$ *then we must in addition have* $\gamma' - \gamma = \int_{-1}^{1} x\,(\upsilon' - \upsilon)\,(dx)$.

When P *and* P' *are equivalent, the Radon-Nikodym derivative is*

$$\frac{dP'|_{\mathcal{F}_t}}{dP|_{\mathcal{F}_t}} = e^{U_t}$$

with

$$U_t = \eta X_t^c - \frac{\eta^2\sigma^2 t}{2} - \eta\gamma t + \lim_{\varepsilon \downarrow 0}\left(\sum_{s\le t, |\Delta X_s|>\varepsilon} \phi\left(\Delta X_s\right) - t\int_{|x|>\varepsilon}\left(e^{\phi(x)} - 1\right)\upsilon(dx)\right), \tag{3.7}$$

where (X_t^c) *is the continuous part of* (X_t) *and* η *is such that*

$$\gamma' - \gamma - \int_{-1}^{1} x\,(\upsilon' - \upsilon)\,(dx) = \sigma^2\eta$$

if $\sigma > 0$ *and zero if* $\sigma = 0$.

In the case of more general jump diffusions (with non-constant drift and diffusion terms) the reader is referred to Jacod and Shiryaev (1980). Note that if $\upsilon = \upsilon' = 0$ then we are in the no jump case and (3.7) reduces to the familiar Girsanov density used in the previous sections. This proposition tells us something important about the class of dominating measures we may use in constructing an acceptance sampling scheme. For the underlying jump diffusion measures to be equivalent we must have equivalence in Levy measures. Hence, the candidate sampling process in any acceptance sampling scheme must itself be a jump process. We cannot produce jump diffusion paths from continuous paths. Furthermore, as Tankov and Cont

(2004) point out we must choose a dominating process whose Levy measure will admit the same type of jumps. For example, we can only dominate a jump diffusion process with a constant jump size with another jump diffusion process admitting the same jump size. In the case of random jump sizes the integrability condition (2) given in the proposition determines the appropriate class of dominating Levy measures.

Consider a jump diffusion whose stochastic differential is given by

$$dY_t = a(Y_{t-})dt + dW_t + c\left(Y_{t-}, D_{N_{t-}+1}\right) dN_t, \tag{3.8}$$

where $\{N_t : t \geq 0\}$ is a Poisson process with intensity λ, $\{D_1, D_2, ...\}$ are i.i.d. random variables with density f_D and W_t is a standard Brownian motion. Assume further that N, W and $D_1, D_2, ...$ are independent of each other. Here we assume a unit diffusion coefficient, however we may transform variables as in the previous section to obtain results for a more general specification. Under this model the process Y behaves like a diffusion with drift function a and identity diffusion coefficient between the jumps. At the time of the kth occurrence of the Poisson process the function c along with the random variable D_k determines the size of the jump in the process Y. Specifically, suppose the kth jump occurs at time τ_k then the jump in Y at this time is given by

$$Y_{\tau_k} - Y_{\tau_k-} = c\left(Y_{\tau_k-}, D_k\right).$$

Consider the problem of simulating the path: $Y_t|Y_0 = x, Y_T = y$ for $t \in [0, T]$. In light of the above result, to remove the dependency on the jump component process in the Girsanov density we should choose a candidate measure with the same jump component as Y. Also, the appealing aspect of the imputation scheme for regular

diffusions is the use of the Brownian bridge and its ease of simulation. To exploit its use here we should restrict the candidate process to one with no drift. This leads to dominating the process Y with a jump diffusion process X of the form:

$$dX_t = dW_t + c\left(X_{t-}, D_{N_{t-}+1}\right) dN_t. \tag{3.9}$$

With such a choice the regular diffusion imputation scheme can be used piecewise over the continuous parts of the proposed path. An imputation algorithm in this context is given as follows.

1. Simulate a path of $X_t|X_0 = x, X_T = y$. This results in a set of jump locations $0 < \tau_1 < ... < \tau_{N_T} \leq T$, i.i.d. jump size random variables $D_1, ..., D_{N_T}$ corresponding to the jump times and the values of the process X_t at the jump times: $X_{\tau_i}, i = 1, ..., N_T$.

2. The process over the $n+1$ time intervals $[0, \tau_1)$, $[\tau_1, \tau_2)$, ..., $[\tau_{n-1}, \tau_n)$, $[\tau_n, T)$ is purely continuous. For each one of these intervals we can use the imputation method for regular diffusions. If all of the segments are accepted we accept the path of X as a path of Y. If not return to step 1.

Once an imputed path is accepted the continuous segments of the path can be filled it at arbitrary time points using Brownian bridge interpolation.

Generating Proposal Paths

The above scheme requires the conditional simulation of X governed by (3.9). In the case that the function c does not depend on the current state of the process the

problem simplifies considerably. For $c(x,d) = c(d)$ we can think of the process Y as being the sum of two independent pieces: one continuous and the second a step function. Thus we can write

$$X_t = W_t + \sum_{j=1}^{N_t} Y_j,$$

where $(Y_j)_{j=1}^{\infty}$ is a sequence of i.i.d. random variables which replace the $c(D_j)$'s. Now, the simulation of $X_t|X_0 = x, X_T = y$ is like simulating two independent random variables conditional on their sum. By independence this can be done using acceptance sampling by first simulating one piece say $\sum_{j=1}^{N_T} Y_j$ and accepting $W_T = X_T - \sum_{j=1}^{N_T} Y_j$ with probability proportional to $\phi(W_t; 0, T)$, where ϕ is the marginal density of W_t (ie. the normal density with mean 0 and variance $\sigma^2 T$). See the appendix for justification. Sampling a path $X_t|X_0 = x, X_T = y$ over the time interval $[0, T]$ can be achieved as follows.

1. Generate $N_T \sim Poi(\lambda T)$. Draw $Y_1, ..., Y_{N_T}$ using the law of Y_1. Put $Y = Y_1 + ... + Y_{N_T}$.

2. Accept W_T, a Brownian motion ending at $X_T - Y = y - Y$, with probability $\phi(y - Y; 0, T)/K$, $K = \frac{1}{\sqrt{2\pi T}}$. If not accepted return to Step 1.

3. Simulate the jump times by sampling $\tau_1, ..., \tau_{N_T} \sim U[0, T]$ i.i.d.

4. Simulate W_{τ_i} $i = 1, .., N_T$ conditional on $W_0 = x$ and $W_T = y - Y$ (Brownian bridge interpolation).

For a more general form of the function c the conditional simulation of X is similar. The jump instances are simulated along with the i.i.d. jump sizes corresponding to those times. The path is then built sequentially, ignoring the final

endpoint, until the value of the path just after the last jump is obtained. The final increment of the path is completely governed by the continuous part of the process, namely the Brownian motion. So the post final jump value of the process and the endpoint of the path can be used to decide acceptance of the path by noting the final increment is Normally distributed. The algorithm is formally based on a result given in the appendix and is outlined as follows.

1. Generate $N_T \sim Poi(\lambda T)$. Simulate the jump times by sampling $\tau_1, ..., \tau_{N_T} \sim U[0,T]$ i.i.d. Draw $D_1, ..., D_{N_T}$ independently using the law of D_1.

2. Draw $\varepsilon_1, ..., \varepsilon_{N_T}$ independent standard Normal random variables. Build the path sequentially up until time τ_{N_T} : set $X_{\tau_1-} = x + \sqrt{\tau_1}\varepsilon_1$, $X_{\tau_1} = X_{\tau_1-} + c(X_{\tau_1-}, D_1)$, $X_{\tau_2-} = X_{\tau_1} + \sqrt{\tau_2 - \tau_1}\varepsilon_2$, $X_{\tau_2} = X_{\tau_2-} + c(X_{\tau_2-}, D_2)$, ..., $X_{\tau_{N_T}-} = X_{\tau_{N_T}-} + \sqrt{\tau_{N_T} - \tau_{N_T-1}}\varepsilon_{N_T}$, $X_{\tau_{N_T}} = X_{\tau_{N_T}-} + c(X_{\tau_{N_T}-}, D_{N_T})$.

3. Accept the final increment of the process: $y - X_{\tau_{N_T}}$ with probability $\phi(y - X_{\tau_{N_T}}; 0, \tau_{N_T} - \tau_{N_T-1})/K$, $K = \frac{1}{\sqrt{2\pi\sigma^2\left(\tau_{N_T} - \tau_{N_T-1}\right)}}$, otherwise return to Step 1.

Notice that these schemes for generating proposal paths can easily accommodate a constant drift as well as a deterministic jump arrival intensity. As with any example of acceptance sampling, choice of the proposal density is not unique and should be considered on the basis of ease of simulation and resemblance to the target random variable or process. In the following example the above method is used to perform conditional simulation of the CIR model with multiplicative jumps.

Example: CIR Process with Jumps

Consider the CIR diffusion process given in equation (3.4) used to model the short rate of interest. Suppose that at times determined by a homogeneous Poisson process the spot rate process exhibits multiplicative jumps. Such an extended CIR process can be represented by the following s.d.e.

$$dr_t = \kappa(\theta - r_{t-})dt + \sigma\sqrt{r_{t-}}dW_t + r_{t-}dJ_t, \tag{3.10}$$

where $J_t = \sum_{i=1}^{N_T}(\phi - 1)$ is a compound Poisson process, N_T is a homogeneous Poisson process with intensity λ and $\phi_1, \phi_2, \phi_3...$ are i.i.d. strictly positive random variables. When a jump occurs at time τ the value of the process at the next instant is: $r_\tau = r_{\tau-} + r_{\tau-}(\phi - 1) = \phi r_{\tau-}$, for some random independent random realization of ϕ. Recall that the variance stabilizing transformation derived in the previous section is given by $s(r) = \frac{2}{\sigma}\sqrt{r}$. Letting $Y_t = s(r_t) = \frac{2}{\sigma}\sqrt{r_t}$ and applying Ito's lemma for jump diffusions the s.d.e. for Y_t can be written as

$$dY_t = a(Y_t)dt + dW_t + Y_t dJ_t^*,$$

where the drift function a is given in equation (3.5) and J_t^* is a compound Poisson process of the form $J_t^* = \sum_{i=1}^{N_T}(\Phi_i - 1)$ where $\Phi_1, \Phi_2, \Phi_3, ...$ are i.i.d. random variables such that $\Phi_i =^d \sqrt{\phi_i + 1}$. Thus the same homogeneous Poisson process determines the jump times of Y and when a jump occurs it is multiplicative by an independent random factor of Φ. Consider the example in the previous section where we simulate a conditional path of r_t over the interval $[0, 1]$ where it is known that $r_0 = r_1 = 0.05$ and $(\kappa, \theta, \sigma) = (0.2, 0.06, 0.1)$. Suppose the jump size random variables for the original process are lognormally distributed with log variance given by δ^2 and log mean $-\delta^2/2$ (with this choice the expected multiplicative jump is 1), so that $\log\phi_i = N(-\delta^2/2, \delta^2)$ $i = 1, 2, 3....$ independently. Note that a lognormal

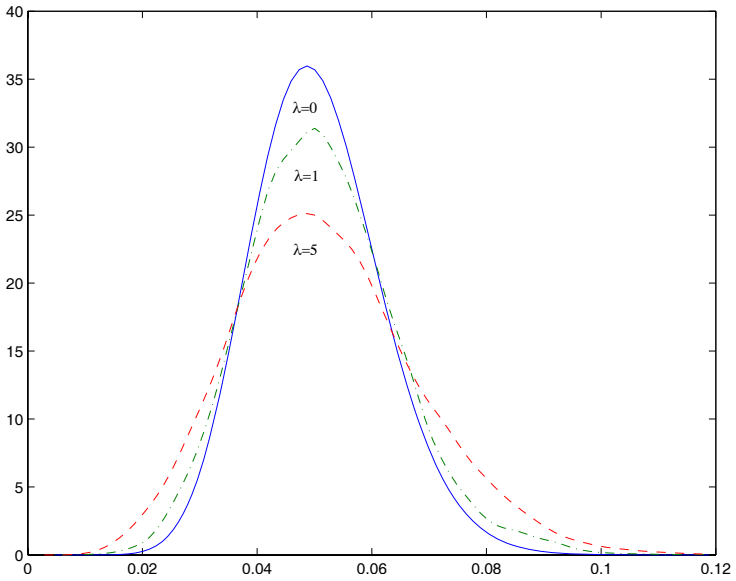

Figure 3.5: Density of $r_{0.5}|r_0 = r_1 = 0.05$ for different λ values. For $\lambda = 0$ the true density is plotted. The plots for $\lambda > 0$ are smoothed empirical densities.

distribution assumption is not crucial here and choosing another distribution does not complicate matters. Figure 3.5 shows empirical density plots of the random variable $r_{0.5}|r_0 = r_1 = 0.05$ when $\delta = 0.1$ for different λ values. The empirical densities were generated using random variates drawn according to the above algorithm smoothed with a Gaussian kernel density estimate. The case $\lambda = 0$ corresponds to the no-jump regular CIR process which is also shown in Figure 3.4. As the intensity of jumps λ gets larger the tails of the bridge density of the process get thicker. This is consistent with the fact that ceteris paribus larger values of λ lead to more variability in the process, and hence more uncertainty in the conditional distribution of $r_{0.5}$.

3.4.2 Path Augmentation via Bridge Sampling

The pure diffusion part of (3.8) satisfies an s.d.e. of the form: $dY_t^* = a(Y_t^*)dt + dW_t$. If we can simulate the transition variates $Y_t^*|Y_0^* = y$ then we can construct a fairly simple algorithm for generating paths for the jump diffusion Y_t. Conceptually the algorithm for generating a discrete realization of Y_t over the interval $[0, T]$ is as follows:

1. Simulate a path of N_t $t \in [0, T]$. Denote the jump times as: $0 = \tau_0 < \tau_1 < ... < \tau_{N_T} \leq T$.

2. Simulate independently the random variables $D_1, ..., D_{N_T}$ which will help determine the jump sizes at each of the jump times.

3. For each jump $k = 1, ..., N_T$ simulate $Y_{\tau_k -}|Y_{\tau_{k-1}}$ according to the distribution of $Y^*_{\tau_k - \tau_{k-1}}|Y_0^*$. Set $Y_{\tau_k} = Y_{\tau_k -} + c(Y_{\tau_k -}, D_k)$.

If the variates $Y_t^*|Y_0^*$ in the above algorithm are sampled using the pathwise method then i) they will be accompanied by a skeleton path which must be stored, and ii) we can fill in the path using a Brownian bridge as before. If not (for example in the case of the CIR we may opt to use the exact non central chi square transitions) the pathwise imputation algorithm can be used to fill in the path between the jump times. This will augment the simulated path with additional points. The time grid defined by the jump locations and any additional points from pathwise imputation form a skeleton path of the process. Between any two jump instances the path of the jump diffusion is continuous, and upon pathwise imputation any Brownian bridge agreeing with the imputed points can used to "fill-in" the path

without destroying the properties of the original simulation. In this way, the finite collection of points of the process simulated at the jump instances and during pathwise imputation provide complete information of the entire continuous path! Just as a piecewise linear function can be defined solely by the location of its knots and the rule of linear interpolation, so can a jump diffusion path be characterized by a skeleton path and Brownian bridge interpolation. This allows one to represent the infinite dimensional object of a jump diffusion path by a finite array of points. The advantage of simulating a path in this fashion is that the segments of the path may be analyzed as if they were a Brownian bridge. This has implications for simulating various path functionals for general jump diffusions, an application which will be explored in Chapter 4.

Chapter 4

Applications of Pathwise Imputation

4.1 Simulating Functionals of a Diffusion's Path

The novelty of the imputation algorithm is that once the skeleton path has been simulated it can subsequently be treated as though it were a Brownian path. Not only is this useful for filling in values of the path at time points of interest, but it is also serves as a means for calculating functionals of the path. For example, the distribution of the maximum of the diffusion Y over $[0, T]$ might be complicated, however it is well known in the case of a Brownian bridge. One can use this to simulate the maximum of Y by simulating a skeleton path of Y and subsequently generating the maximum of a Brownian bridge over each subinterval of the skeleton path. Since each intervening Brownian bridge is independent of the others the maximum of Y can be calculated as the largest maximum of the simulated

bridges. The same is true for simulating the first hitting time of Y to some barrier. Conditional on a Brownian bridge hitting some barrier over an interval the hitting time distribution is Inverse Gaussian. Combining this with the known probability of a Brownian bridge hitting a barrier simulating hitting times for the diffusion Y is straight forward. Some applications require the probability that a path has never crossed a barrier rather then the hitting time or the maximum. This is treated in much the same way by calculating separately the probabilities over each sub interval of the skeleton path. Due to independence of the subintervals, the product of these probabilities is the probability that the path never reached the barrier over the full time interval $[0, T]$. Sometimes we will be interested in simulating the above functionals over sub time intervals of the path. This is done by first augmenting the path using Brownian bridge interpolation at the interval endpoints of interest, and subsequently simulating or calculating the required functional over each subinterval. The results are then aggregated easily using the fact that Brownian paths living on disjoint intervals are independent. Of course, we are generally interested in applying these results to the original diffusion X. The transformation of Y back to the original diffusion X is monotone and so the simulations translate back to the variable of interest quite easily.

For the applications to be discussed we will need a variety of simulation methods. Pricing lookbacks will require us to simulate the maximum or the minimum of a diffusion. Efficient simulation of single and double barrier option prices can be carried out by calculating the probability that a skeleton path has not crossed a single or double boundary. Pricing a barrier option with a rebate will require the simulation of the hitting time to a boundary. Pricing extreme spread options

which depend on the maximum value of an asset over two disjoint intervals are easily handled by simulating the maximum over each interval separately. Another type of barrier option known as a hotdog option depends on the boundary crossing of an asset to a piecewise constant barrier. Again, this is fairly simple to handle using a divide and conquer strategy over each interval corresponding to different barrier levels. In the next few sections simulation algorithms are developed for a general diffusion X for tackling these and other types of problems. We adopt the notation from Chapter 3 here and denote $s : \Re \longrightarrow \Re$ to be the variance stabilizing transformation of X with inverse function $g : \Re \longrightarrow \Re$. Note that all algorithms are stated for bridge diffusion processes (conditioned diffusions). They also apply to non-conditioned diffusions as long as the endpoint of the process is sampled according to the density derived by Beskos and Roberts (2005) which is given in equation (3.3). Also, the following discussions extend quite naturally to jump diffusions whose arrival intensity does not depend on the state of the process.

4.1.1 Probability of Barrier Crossing

Suppose W_u is a Brownian bridge over $[t, s]$ $(t < s)$ starting at $W_t = x$ and ending at $W_s = y$. Let $H \in \Re$ be some barrier level. Let $M_{t,s} = \max_{t \le u \le s} W_u$ and $m_{t,s} = \min_{t \le u \le s} W_u$. Then we have the following result which gives the probability that the maximum of a Brownian bridge remains below a given barrier over a finite time interval:

$$P\{M_{t,s} \le H\} = \begin{cases} 1 - \exp\left[-\frac{2(H-x)(H-y)}{s-t}\right], & H > \max(x, y) \\ 0, & \text{otherwise} \end{cases}. \qquad (4.1)$$

If we are interested in the minimum of a Brownian bridge remaining above some level we can use the symmetry of a Brownian path. We simply apply the above

result to the negative of the path and the barrier level. For a double barrier things become more complicated. To analyze this case we begin by making the simplifying assumption that W_u is a Brownian bridge over $[0,1]$ starting at $W_0 = 0$ and ending at $W_1 = y$. . Billingsley (1968) gives the following result:

$$\begin{aligned} P(a,b,u,v) &= P\left\{a < m_{0,1} \leq M_{0,1} < b, u < W_1 < v\right\} \\ &= \sum_{k=-\infty}^{\infty} P\left\{u + 2k(b-a) < Z < v + 2k(b-a)\right\} \\ &\quad - \sum_{k=-\infty}^{\infty} P\left\{2b - v + 2k(b-a) < Z < 2b - u + 2k(b-a)\right\}, \end{aligned}$$

where $Z \sim N(0,1)$. From this we can obtain the probability of a Brownian bridge not hitting a double barrier by the following calculation:

$$\begin{aligned} P_{NH}^{BB}(a,b,u) &= P\left\{a < m_{0,1} \leq M_{0,1} < b | W_1 = u\right\} \\ &= \lim_{\delta \longrightarrow 0} P\left\{a < m_{0,1} \leq M_{0,1} < b | u < W_1 < u + \delta\right\} \\ &= \lim_{\delta \longrightarrow 0} \frac{P\left\{a < m_{0,1} \leq M_{0,1} < b, u < W_1 < u + \delta\right\}}{P\left\{u < W_1 < u + \delta\right\}} \\ &= \lim_{\delta \longrightarrow 0} \frac{P\left\{a < m_{0,1} \leq M_{0,1} < b, u < W_1 < u + \delta\right\}/\delta}{P\left\{u < W_1 < u + \delta\right\}/\delta} \\ &= \frac{\frac{\partial P(a,b,u,v)}{\partial v}|_{v=u}}{\phi(u)} \\ &= \frac{1}{\phi(u)} \left[\sum_{k=-\infty}^{\infty} \phi\left(u + 2k(b-a)\right) - \sum_{k=-\infty}^{\infty} \phi\left(2b - u + 2k(b-a)\right) \right], \end{aligned}$$

where ϕ is the Standard Normal density which is the density of the random variable $W_1 | W_0 = 0$. The terms in this sum decay quite quickly permitting quick and accurate computation of this probability. The problem of a general Brownian bridge over $[t,s]$ starting at $W_t = x$ and ending at $W_s = y$ with maximum $M_{t,s}$ and

minimum $m_{1,s}$ over $[t,s]$ is now easily calculated noting the relationship

$$P\left\{a < m_{t,s} \leq M_{t,s} < b | W_t = x, W_s = y\right\} = \frac{P_{NH}^{BB}\left(\frac{a-x}{\sqrt{s-t}}, \frac{b-x}{\sqrt{s-t}}, y-x\right)}{\sqrt{s-t}}. \qquad (4.2)$$

The algorithm for simulating a path of a diffusion X over $[0,T]$ conditional on $X_0 = x$ and $X_T = y$ along with the probability that X has not crossed a single barrier H (or a double barrier given by $H > L$) is given as follows.

1. Transform the diffusion X, the endpoints the barrier level(s) using the transformation s. Let $Y_t = s(X_t)$ (which is a diffusion with unit diffusion coefficient), $x^{'} = s(x), y^{'} = s(y)$, $H' = s(H)$ and $L' = s(L)$.

2. Simulate a skeleton path of $Y_t | Y_0 = x', Y_T = y'$ over $[0,T]$ to arrive at $\{(0,x'),\ (\tau_1, Y_{\tau_1}),\ ...,\ (\tau_n, Y_{\tau_n}),\ (T,y')\}$.

3. For each of the $n+1$ intervals of the skeleton path calculate the probability the barrier has not been breached using one of the appropriate formulae given above ((4.1) and (4.2) for single and double barrier calculations, respectively). Label these probabilities p_i, $i = 1, ..., n+1$.

4. Return the finite path of $X = g(Y)$ which is given by transforming the skeleton path of Y via the transformation g: $\{(0,x),\ (\tau_1, g(Y_{\tau_1})), ..., (\tau_n, g(Y_{\tau_n})),\ (T,y)\}$. Return $P = \prod_{i=1}^{n+1} p_i$ which is the probability the barrier H (or $H > L$) has not been crossed by X_t over $[0,T]$.

If X_t is required at additional time points they must be simulated between steps 2 and 3 using Brownian bridge interpolation.

4.1.2 Simulating Extreme Values

Suppose W_u is a Brownian bridge over $[t, s]$ starting at $W_t = x$ and ending at $W_s = y$. Let $M_{t,s}$ and $m_{t,s}$ once again denote the maximum and minimum of a Brownian bridge over the time interval $[t, s]$. Beskos and Roberts (2005) show how one can simulate $M_{t,s}$ with relative ease. If $E \sim Exp(1)$ is an Exponential random variable with mean 1 then

$$M_{t,s} =^d \frac{1}{2}\left(\sqrt{2(s-t)E + (y-x)^2} + x + y\right), \tag{4.3}$$

where $=^d$denotes equivalent in distribution. The minimum is generated by symmetry

$$m_{t,s} =^d -\frac{1}{2}\left(\sqrt{2(s-t)E + (y-x)^2} - x - y\right).$$

These formulae enable one to marginally generate the minimum or the maximum of a Brownian bridge by transforming an Exponential random variable. Simultaneous generation of the minimum and the maximum $(m_{t,s}, M_{t,s})$ is not as simple. For this we first simulate the maximum $M_{t,s}$ of the Brownian bridge using the result in equation (4.3). We subsequently use the inverse c.d.f. method to generate $m_{t,s}|M_{t,s}, W_t, W_s$. The c.d.f. for this conditional distribution is derived from the Billingsley (1968) result given in the previous section and also appears in McLeish

(2005). For a Brownian bridge over the unit interval started at 0 we have

$$\begin{aligned} F_{L|H,C}(a|b,u) &= P\left\{m_{0,1} \leq a | M_{0,1} = b, W_1 = u\right\} \\ &\quad 1 + \frac{1}{2\phi'(2b-u)} \frac{\partial^2}{\partial b \partial v} P(a,b,u,v)|_{v=u} \\ &= \frac{-1}{\phi'(2b-u)} \\ &\quad \times \sum_{k=1}^{\infty} \left\{ \begin{array}{l} -k\phi'(u+2k(b-a)) + (1+k)\phi'(2b-u+2k(b-a)) \\ +k\phi'(u-2k(b-a)) + (1-k)\phi'(2b-u-2k(b-a)) \end{array} \right\}. \end{aligned}$$

The minimum cannot exceed the maximum or the endpoint value. This cdf can be inverted efficiently using a hybrid Newton-Raphson and bisection method constrained to an interval. The terms in the above series are Normal density functions which are easy to differentiate making the derivative of this c.d.f. explicitly available for root finding. Also, since these density functions decay very quickly, the above infinite series converges quickly making calculations efficient. The algorithm for jointly generating the minimum and maximum of a Brownian bridge over $[t,s]$ starting at $W_t = x$ and ending at $W_s = y$ is given as follows.

1. Sample $E \sim Exp(1)$. Set $M_{t,s} = \frac{1}{2}\left(\sqrt{2(s-t)E + (y-x)^2} + x + y\right)$.
2. Sample independently $U \sim U[0,1]$. Find a such that $F_{L|H,C}\left(a|\frac{b-x}{\sqrt{s-t}}, \frac{y-x}{\sqrt{s-t}}\right) = U$. Set $m_{t,s} = \sqrt{s-t}a + x$.

This provides us with the means for joint simulation of the maximum and minimum of a Brownian bridge. This is easily extended to conditioned diffusions. The algorithm for simulating the extremes of a general diffusion X_t over $[0,T]$ conditional on the endpoints $X_0 = x$ and $X_T = y$ is described as follows.

1. Transform the diffusion X and its endpoints. Let $Y_t = s(X_t)$, $x' = s(x)$ and $y\prime = s(y)$.

2. Simulate a skeleton path of $Y_t|Y_0 = x', Y_T = y'$ over $[0,T]$ to arrive at $\{(0,x'),\ (\tau_1, Y_{\tau_1}), ...,\ (\tau_n, Y_{\tau_n}),\ (T,y')\}$.

3. For each of the $n+1$ intervals simulate the required extreme of the process Y_t using the appropriate Brownian bridge method above. Label these extremes (m_i, M_i), $i = 1, ..., n+1$. Let $m_Y = \max\{m_1, ..., m_{n+1}\}$ and $M_Y = \max\{M_1, ..., M_{n+1}\}$.

4. Return the finite path of $X = g(Y)$ $(\{(0,x),\ (\tau_1, g(Y_{\tau_1})),\ ...,\ (\tau_n, g(Y_{\tau_n})),\ (T,y)\})$ along with the minimum and maximum of this path which are given by $m = g(m_Y)$ and $M = g(M_Y)$.

Again, if X_t is required at additional time points they must be simulated between steps 2 and 3 using Brownian bridge interpolation before the extremes are simulated.

4.1.3 Simulating Hitting Times

The results regarding simulation of the hitting time of a Brownian bridge to some barrier are given in Beskos and Roberts (2005) as well as many other sources. Suppose W_u is a Brownian bridge over $[0,t]$ started at 0 and terminating at y. Consider some barrier level $\alpha > 0$ (for $\alpha < 0$ use symmetry by considering the negative path and barrier level). Let $\tau_\alpha = \inf\{s \in [0,t] : W_s \geq \alpha\}$. Simulation of the hitting time τ_α to α is computed as follows. Let $\eta = \alpha/\sqrt{t}$ and $\zeta = \frac{\alpha - y}{\sqrt{t}}$. If $\eta\zeta < 0$ then the barrier α is hit with probability 1 and the hitting time can be

simulated as $\tau_\alpha = t\frac{I}{I+1}$, where $I \sim IG\left(-\eta/\zeta, \eta^2\right)$. IG denote the inverse Gaussian distribution. If $\eta\zeta > 0$ then the barrier α is hit with probability $\exp(-2\eta\zeta)$. If it is hit the hitting time can be generated be setting $\tau_\alpha = t\frac{I}{I+1}$, where $I \sim IG\left(\eta/\zeta, \eta^2\right)$. If the barrier is not hit in $[0, t]$ we can conclude that $\tau_\alpha > t$.

For a conditioned diffusion X with endpoints x and y over the time interval $[0, T]$ the algorithm for simulating the hitting time τ_H to a barrier level H is given as follows.

1. Transform the diffusion, it endpoints and the barrier level using the transformation s. Let $Y_t = s(X_t)$, $x' = s(x)$, $y\prime = s(y)$ and $H' = s(H)$.

2. Simulate a skeleton path of $Y_t|Y_0 = x', Y_T = y'$ over $[0, T]$ to arrive at $\{(0, x'), (\tau_1, Y_{\tau_1}), ..., (\tau_n, Y_{\tau_n}), (T, y')\}$.

3. Treating the skeleton path as the points of a Brownian bridge proceed sequentially through each sub-interval simulating a hitting time to H'. If a barrier is hit in a given interval record the time τ_H at which this occurs, stop traversing the intervals and proceed to step 4.

4. Return the finite path of $X = g(Y)$ $(\{(0, x), (\tau_1, g(Y_{\tau_1})), ..., (\tau_n, g(Y_{\tau_n})), (T, y)\})$along with the hitting time τ_H if one has occurred. If the barrier has not been breached then the event $\{\tau_H > T\}$ has occurred.

4.2 Pricing Exotic Options in the CEV Jump Diffusion Model

In this section we extend the CEV process which has been well studied in the literature. One of the apparent deficiencies of the Black-Scholes model is the fact that the instantaneous return volatility of the stock is constant. This has consequences for option pricing and it is known that more reasonable volatility assumptions allow for closer empirical fits to observed data. The addition of jumps to the price process and/or a stochastic volatility component is known to improve pricing performance of most models. The CEV model of stock prices is similar in many respects to the Black-Scholes model. It attempts to resolve some shortcomings in the Black-Scholes framework by relating the instantaneous return volatility to the current stock price. The traditional CEV model is formulated so that when the stock price increases, its returns volatility decreases and vice versa. The risk neutral dynamics of the CEV process are given as

$$\frac{dS_t}{S_t} = rdt + \sigma S_t^{\frac{\alpha}{2}-1} dW_t, \tag{4.4}$$

where r is the continuously compounded riskless rate of interest, $\sigma > 0$ and α are constants which determine the instantaneous volatility of the process. We follow the approach of most authors by employing the restriction $0 \leq \alpha \leq 2$. Many popular models are nested within this one. For $\alpha = 0$ we get the absolute model, $\alpha = 1$ gives the square root process and $\alpha = 2$ gives the Black-Scholes framework. The CEV process was first introduced by Cox (1975) as an alternative to Geometric Brownian motion. For additional insight into the model and its consequences for pricing the reader is referred to Beckers (1980) and Emanuel and MacBeth (1982).

Schroder (1989) develops a method for calculating European call option prices which involves efficient computation of the non central chi squared distribution. More recently the model has been used by Boyle and Tian (1999) to calculate the prices of path dependent options. Specifically, they develop a trinomial tree approximation to 4.4 and use it to price barrier and lookback options. They find that European call prices produced by the model are similar to those implied by the Black-Scholes model. However, for different values of α, the prices of barriers and lookbacks can significantly differ from their Black-Scholes counterparts. Linetsky and Davydov (2001) take this analysis one step further and derive analytical results for the Laplace transforms of barrier and lookback prices. They also examine the model for values of α outside the traditional range. One of their key findings is that the delta hedges for certain barrier and lookback options under different specifications of the CEV model can have different signs then those implied by the Black-Scholes model. In these analyses the authors compared prices from different models by matching instantaneous return variances of the underlying processes. For a given volatility σ_{BS} in the Black-Scholes framework and sub model indexed by α this is achieved by setting $\sigma = \sigma_{BS} S_0^{1-\alpha/2}$. Varying the α parameter provides a rich class of models and a range of prices for these path dependent securities.

Kou (2002) develops a double exponential jump diffusion model and compares it to several models in the literature. He discusses how the CEV model lacks realism due to the fact that it is unable to produce a leptokurtic return distribution, a feature often observed in financial time series. He points out that, as a result of this, the CEV model is unable to reproduce the implied volatility curve present in some options markets. As is summarized in Benzoni et. al. (2002), Jones

(1999) estimates the CEV model using a Bayesian MCMC technique and states that augmenting the Black-Scholes model with the CEV volatility function could provide a reasonable substitute for the addition of a jump component. However, Benzoni et. al. go on to state that "...the full option smirks cannot be rationalized by the estimated model, and the performance of the CEV specification relative to jump-diffusions is unclear." A common message being conveyed by all of these papers is that the CEV model is useful and has a valuable place in the literature, but like the Black-Scholes model it also suffers from certain non-ignorable deficiencies. This provides the motivation to consider an extension of the CEV model by the addition of a jump component to (4.4). We write the extended model as

$$\frac{dS_t}{S_{t-}} = \mu dt + \sigma S_{t-}^{\frac{\alpha}{2}-1} dW_t + dJ_t,$$

where J is the jump component of the process. Pathwise imputation methods will be used to examine the resulting model in a quick and efficient way. As is common in the literature, we assume that J takes on the following form

$$J_t = \sum_{j=1}^{N_t} (Y_j - 1),$$

where N_t is a Poisson process with intensity λ, and $(Y_n)_{n=1}^{\infty}$ are iid random variables. In the jump setting there is no unique martingale measure, but in this case choosing

$$\mu = r - \lambda m,$$

where $m = E[Y_j] - 1$ makes the discounted price process $e^{-rt}S_t$ a martingale enforcing the no arbitrage condition (see Andersen and Andreasen 2000). Thus, the price of a security dependent on S with payoff function h can be represented as the conditional expectation: $e^{-rT}E\left[h(S_T)|S_0\right]$. The form of J admits the interpretation that if τ_j, $j = 1, ..., n$, are the jump times of the process then the jump in S

is multiplicative and takes the form $S_{\tau_j} = S_{\tau_j -} Y_j$. The intensity of the Poisson process could be taken to be deterministic or stochastic, however we will hold it constant for this analysis. The distribution of the multiplicative jump sizes Y_j must be specified. There are several choices for this. For instance, in the context of a Geometric Brownian motion with jumps Kou (2002) specifies the jump size as double exponential claiming that such a distribution is conducive to analytical computation. Instead we will follow the lead of most authors who specify a log-normal jump distribution, see for example Merton (1976) and Metwally and Atiya (2002). We assume that

$$\log Y_1 \sim N\left(\nu, \delta^2\right),$$

for some $-\infty < \nu < \infty$ and $\delta > 0$.

4.2.1 Simulation of the CEV Jump Diffusion Model

In order to simulate the CEV jump diffusion process we will first simulate the jump times according to a Poisson process and subsequently simulate the value of the process the instant before and after each jump. This is done without appealing to any time step schemes. Instead, CEV transition variates are drawn directly from the transition density which is given in the next section. Once the process is simulated at the jump instances we draw bridge paths using the conditional simulation scheme to connect the path across these points. This leaves us with a skeleton of the jump diffusion path whose time points are the union of the jump times and any additional time points generated by the bridging algorithm. The advantage of constructing paths in this fashion is that the segments between time points can be treated as (transformations of) Brownian paths. This will allow for

efficient joint simulation of CEV jump diffusion terminal values and functionals of the path. We begin by considering the diffusion part of (4.4)

$$dS_t = rS_t dt + \sigma S_t^{\frac{\alpha}{2}} dW_t.$$

It is known that for $\alpha \geq 1$ that this process has a unique solution with an absorbing boundary at 0 and ∞ is an unattainable boundary (see Andersen and Andreasen 2000). For $\alpha < 1$ there is no unique solution unless we attach an additional condition to the boundary 0. Similar to the approach taken in Andersen and Andreasen (2000) we will associate the solution to the above s.d.e. as the one with an absorbing boundary at $S = 0$ for all $0 \leq \alpha < 2$. This condition must be enforced to ensure the martingale property of the discounted price process. For $\alpha = 2$ the process is Geometric Brownian motion and is positive almost surely. Thus, in constructing a forward simulation algorithm we should include the possibility that the process hits 0 although we will see this occurs with a negligible probability and can often be ignored without resulting in noticeable bias. The continuous part of the density of $S_T|S_t$ is a transformation of the non-central chi squared distribution and was derived by Cox (1975). It is given by

$$f_{CEV}\left(S_T, T; S_t, t\right) = (2-\alpha)\gamma^{\frac{1}{2-\alpha}} \left(ab^{1-2\alpha}\right)^{\frac{1}{4-2\alpha}} e^{-a-b} I_{\frac{1}{2-\alpha}}\left(2\sqrt{ab}\right), \tag{4.5}$$

where

$$\tau = T - t,$$
$$\gamma = \frac{2r}{\sigma^2 (2-\alpha)\left(e^{r(2-\alpha)\tau} - 1\right)},$$
$$a = \gamma S_t^{2-\alpha} e^{r(2-\alpha)\tau},$$
$$b = \gamma S_T^{2-\alpha},$$

and I_q is the modified Bessel function of the first kind of order q. Note that this density does not integrate to one and must be weighted by$[1 - P(S_T = 0)]^{-1}$. Boyle and Tian (1999) give the strictly positive absorption probability for the process in the case $0 \leq \alpha < 2$ as

$$G\left(\frac{1}{2-\alpha}, \gamma S^{2-\alpha} e^{r(2-\alpha)T}\right) \tag{4.6}$$

where $G(.,.)$ is the complementary Gamma distribution function given by

$$G(u,v) = \frac{1}{\Gamma(u)} \int_v^\infty e^{-x} x^{u-1} dx.$$

Thus, we can sample a variate of the form $S_T|S_t$ by first testing whether the process hits 0. This can be done by drawing the indicator random variable $I\left\{U \leq G(\frac{1}{2-\alpha}, \gamma S^{2-\alpha} e^{r(2-\alpha)T})\right\}$ where $U \sim U[0,1]$ is a uniform random number. Conditional that the process has not hit 0 we can sample $S_T|S_t$ from the above density which is simply a transformation of a non-central chi square density. The appropriate transformation is given in Cox (1975) and Schroder (1989) as $X_t = S_t^{2-\alpha}$ which can be used to get an expression for the cdf of $S_T|S_t$ in terms of the non central chi squared distribution. It is given by

$$P(S_T \leq s|S_t) = 1 - \mathcal{X}^2\left(2a, \frac{2}{2-\alpha}, 2\gamma s^{2-\alpha}\right),$$

where $\mathcal{X}^2(.,v,\lambda)$ is the non-central chi square distribution function with v degrees of freedom and noncentrality parameter λ. This can be subsequently inverted to produce the required random draws. We performed such an inversion using a hybrid Newton-Raphson and bisection method scheme. Though it works well it can be time consuming and there are alternatives in the context of pricing. If the random variates are used to calculate an expectation via Monte Carlo then one could alternatively use an importance sampling scheme. Of course this requires that estimators are weighted with the appropriate likelihood ratios. This method is very fast and efficient. To sample the endpoints of CEV paths we simulate comparable Black-Scholes (Geometric Brownian Motion) transition variates, and subsequently weight our estimators using the ratio of the CEV density to that of the appropriate log normal density. Does this still allow us to use to exploit the pathwise bridge imputation technique to analyze CEV paths as if they were Brownian? Indeed it does. Upon obtaining a log normal transition variate we bridge the initial value to the endpoint using the pathwise simulation techniques described earlier. Calculating or simulating (transformations of) functionals of this path can be performed as if was a Brownian bridge. Since the conditional bridge path is simulated correctly as a CEV path we need only use the likelihood ratio corresponding to the endpoint of the path, and not the ratio corresponding to the joint distribution of the endpoint and the functional of interest. For jump diffusions there is a slight modification in that the value of the process at the instant before each jump is simulated using lognormal variates. This requires that we weight the estimator with a product of likelihood ratios, one for each jump time as well as the terminal value of the path. Formally, the method is justified as follows. Suppose $g(S_u(\omega), 0 \leq u \leq T)$ is some functional of a CEV path, for example this

could be the maximum, minimum or the hitting time of the path to a barrier level. Suppose $h(g, S_T)$ is a payoff function depending on both the endpoint of the path and this functional. Let $f_\alpha = f_{CEV}$ denote the density function of the random variable $S_T^\alpha | S_0$ and denote by E_α the expectation operator under the CEV dynamics indexed by α. Ignoring the discount factor for the moment, the value of a contract with payoff function h is given as

$$E_\alpha [h(g, S_T) | S_0] = E_\alpha [h(g, S_T) | S_0, S_T > 0] P(S_T > 0) + E_\alpha [h(g, 0) | S_0] P(S_T = 0). \tag{4.7}$$

In many instances the second term in (4.7) will be zero, for example if h is the payoff function of a down-and-out call. In any case, evaluation of the second term relies only on calculating the appropriate path functional with the knowledge that absorption at zero has occurred. Evaluating the first term in (4.7) via Monte Carlo simulation involves generating CEV terminal values conditional on absorption at zero not occurring. This requires simulation from the density given in (4.5). We have the machinery to do this, however for computational concerns we opt for an importance sampling alternative which draws from a log normal density instead. Note that we can rewrite the expectation of interest as

$$\begin{aligned}
E_\alpha [h(g, S_T) | S_0, S_T > 0] &= E_\alpha [E_\alpha [h(g, S_T) | S_T, S_0, S_T > 0] | S_0, S_T > 0] \\
&= \int_0^\infty E_\alpha [h(g, S_T) | S_T, S_0, S_T > 0] f_\alpha(S_T) dS_T \\
&= \int_0^\infty E_\alpha [h(g, S_T) | S_T, S_0, S_T > 0] \frac{f_\alpha(S_T)}{f_2(S_T)} f_2(S_T) dS_T \\
&= E_2 \left[E_\alpha [h(g, S_T) | S_T, S_0] \frac{f_\alpha(S_T)}{f_2(S_T)} | S_0, S_T > 0 \right].
\end{aligned}$$

The form of this expectation gives the recipe for simulation. If absorption has not occurred the endpoint of the path is drawn from a log normal distribution. Conditional on the initial value of the process and the simulated endpoint the functional is generated under the original dynamics of the process, and the final payoff is weighted with a likelihood ratio. The steps for completing one simulation are as follows.

1. Simulate $U \sim U(0,1)$. If $U < G(\frac{1}{2-\alpha}, \gamma S^{2-\alpha} e^{r(2-\alpha)T})$, $S_T^\alpha = 0$ so absorption occurred and we proceed to step 3. Otherwise continue to step 2.

2. Simulate $S_T \sim f_2$ (lognormal variate) and record the likelihood ratio $LR = f_\alpha(S_T)/f_2(S_T)$.

3. Calculate the functional of the path g conditional on the initial value and the terminal value determined in step 1 or 2. This is done assuming the original dynamics of S and not under the log normal assumption.

4. Calculate the option payoff: $h(g, S_T)$.

5. The contribution to the final Monte Carlo estimator is of the form: $h(g, S_T)LR$.

In order to simulate one jump diffusion path the above algorithm is applied piecewise over the time intervals determined by the jump times. Over each interval the process is governed by the continuous part of the jump diffusion. Results concerning functionals and likelihood ratios need to be aggregated accordingly across time intervals. For example, the maximum of a jump diffusion path can be simulated by generating independently the maximum of the segments of the path pertaining to sequential jump times. The largest of these local maximums is then

taken as the maximum value over the entire path. As long as we treat the jump diffusion paths piecewise according to the jump times all of the calculations for purely continuous processes extend quite naturally.

Absorption at Zero

Some special considerations need to be made in the case of absorption, ie. in the case that the process S_t was determined to have hit 0. For a barrier call option without rebate an absorption has the effect of a worthless expiry since 0 must be less (or equal to) than any strike price. A Down-and-Out put option behaves similarly since the event of absorption implies the option has been knocked out. However, for example, for a Up-and-Out put option, absorption to zero does not imply that the option expires worthless. If the path of the underlying fails to trigger the upper boundary and hits zero before expiry the option would pay the strike price at maturity. In general, handling this case requires simulation of various path functionals conditional on absorption. Let $\tau_0 = \inf\{t : S_t = 0,\, 0 < t \leq T\}$ be the first hitting time of the CEV process to the level 0. Then the c.d.f. of τ_0 conditional that absorption has occurred is given by:

$$P(\tau_0 \leq t | \tau_0 \leq T) = \frac{P(\tau_0 \leq t)}{P(\tau_0 \leq T)} = \frac{P(S_t = 0)}{P(S_T = 0)}. \tag{4.8}$$

The probabilities on the right hand side are given in equation (4.6). Thus, conditional that the process has hit zero the hitting time can be simulated by inverting (4.8). Once a hit time is generated one is faced with the task of simulating (or calculating) some functional of the path given its initial value and the absorption time τ_0. This is a difficult problem. In the case where the functional of interest is the maximum and the underlying process is a Brownian motion this problem is

equivalent to simulating the maximum of a Bessel bridge. In a 1999 paper, Pitman and Yor discuss this problem showing that the c.d.f. of the maximum can be written as a infinite series involving the zeros of the Bessel function. They also give other representations most of which are terribly inconvenient for computational purposes. The apparent difficulty in the "simple" context of Brownian motion coupled with the (extremely) low probability of absorption justify an approximate route to solving the problem. If the process hits zero the hitting time τ_0 is simulated and a bridge path is constructed from the initial point to some small value $\varepsilon > 0$ at time τ_0. The simulation then proceeds as before treating this bridge as if it were constructed with regard to absorption. In general one should calculate the probability of absorption to determine to what extent this approximation will affect results. For the parameter values tests in Boyle and Tian (1999) and Linestsky and Davydov (2001) the absorption probability is about 10^{-15} and so the event can be ignored without affecting any significant digits of any estimators.

4.2.2 Bridge Path Simulation in the CEV Model

With an endpoint of the path simulated we wish to construct bridge paths for the CEV process so that functionals of the path can later be analyzed by appealing to Brownian bridge calculations. Recall that in the discussion on conditional simulation of diffusions we required an identity diffusion coefficient. In the case of the CEV process we need to transform the process to achieve this. The appropriate transformation is also used in Boyle and Tian (1999) in the construction of a trinomial tree approximating the continuous time process. For $0 < \alpha < 2$ it is given by

$$s(x) = \frac{1}{\sigma}\int^{x} z^{-\frac{\alpha}{2}}dz = \frac{1}{\sigma}\left(1-\frac{\alpha}{2}\right)^{-1} z^{1-\frac{\alpha}{2}}.$$

This has inverse:

$$g(y) = \left[\sigma\left(1-\frac{\alpha}{2}\right)\right]^{\frac{2}{2-\alpha}} y^{\frac{2}{2-\alpha}}.$$

Let $Y_t = s(S_t)$ then the s.d.e. for Y is given by $dY_t = a(Y_t)dt + dW_t$ with

$$\begin{aligned}
a(y) &= \frac{rg(y)}{\sigma g(y)^{\frac{\alpha}{2}}} - \frac{1}{2}\sigma\frac{\alpha}{2}g(y)^{\frac{\alpha}{2}-1} \\
&= \frac{r}{\sigma}g(y)^{\frac{2-\alpha}{2}} - \frac{1}{2}\sigma\frac{\alpha}{2}g(y)^{\frac{\alpha-2}{2}} \\
&= \frac{r}{\sigma}\left(\left[\sigma\left(1-\frac{\alpha}{2}\right)\right]^{\frac{2}{2-\alpha}} y^{\frac{2}{2-\alpha}}\right)^{\frac{2-\alpha}{2}} - \frac{1}{2}\sigma\frac{\alpha}{2}\left(\left[\sigma\left(1-\frac{\alpha}{2}\right)\right]^{\frac{2}{2-\alpha}} y^{\frac{2}{2-\alpha}}\right)^{\frac{\alpha-2}{2}} \\
&= r\left(1-\frac{\alpha}{2}\right)y - \frac{\alpha}{4}\left(1-\frac{\alpha}{2}\right)^{-1} y^{-1}.
\end{aligned}$$

For $\alpha = 0$ and $\alpha = 2$ the respective unit diffusion coefficient transformations are given by $s(x) = \frac{1}{\sigma}x$ and $s(x) = \frac{1}{\sigma}\log x$. Let $h(y) = (a^2 + a')(y)$ be the required intensity function used in generating the non homogeneous Poisson process for conditional simulation. The method required this function to be bounded above and below. For the CEV model with $0 < \alpha < 2$ the intensity is given explicitly as

$$h(y) = r^2(1-\alpha)y^2 + \left[\frac{\alpha^2}{4(1-\alpha)^2} - \frac{\alpha}{2(1-\alpha)}\right] y^{-2} + r\left(1-\frac{3\alpha}{2}\right).$$

This function is bounded below achieving its minimum at $y_* = \left[\frac{2\alpha(1-\alpha)}{4r^2(1-\alpha)^3}\right]^{\frac{1}{4}}$, however it is not bounded above (as y approaches 0 or ∞ it becomes unbounded). We modify the intensity function by truncating it in a low probability region. In other

words we choose δ small and K large and take our new intensity function to be

$$\widetilde{h}(y) = \begin{cases} h(y) & y \in [\delta, K] \\ h(\delta) & y \in [0, \delta] \\ h(K) & y \in [K, \infty) \end{cases} .$$

This function is bounded above by $\max\left(h(\delta), h(K)\right)$. This method is used in the case of sampling bridge variates of the CIR process in Chapter 3 and performs extremely well. It is worth mentioning that Beskos et. al. (2005) discuss the simulation methodology in more detail and provide a general algorithm for handling some non bounded intensity functions. For example, they give a result for intensity functions bounded in one tail. It requires the conditional simulation of a Brownian Bridge given the maximum or the minimum. This is can be performed exactly by noting that the conditioned process is a Bessel bridge (Revue and Yor 1994). This approach is used for the case $\alpha = 0$ for which the intensity function is bounded in the left tail. Details concerning the simulation in this case is found in Beskos et. al. (2005). To implement a similar method for $0 < \alpha < 2$ one is required to simulate a Brownian bridge conditioned on both its maximum and minimum. There is no such method available in the literature for doing so precisely, and it is not clear that approximate methods for doing so outperform the method of truncation described above. In any event, truncation is very simple and does not require any additional simulation techniques. It reduces all computations to the context of a Brownian motion which is a great conceptual and computational advantage in practice.

4.2.3 Pricing Path Dependent Options

In this section the bridge sampling method is applied to the pricing of lookback and barrier options (including those equipped with rebate payments). Barrier options come in many forms but they all share the common trait that if the underlying's price reaches a certain level during the life of the option the payoff at maturity is affected in some way. Thus it is called a path dependent option. For example, a Down-and-Out European call option with strike price X is a type of barrier option similar to a plain vanilla call except that it may becomes "knocked-out" if the underlying hits a certain lower barrier level during the life of the option. Once an option is knocked out it is deemed to be worthless. If the barrier event is not triggered during the option's life then at maturity it pays what a regular vanilla call would pay. Thus, for a lower barrier level LB and an option life over the time interval $[0, T]$ the payoff function has some path dependency taking the form

$$1\left\{\inf_{t\in[0,T]} S_t > LB\right\}\max\left(S_T - X, 0\right).$$

Other examples of knock-out options are the Up-and-Out call (which has an upper barrier level) and the double barrier knock-out call (which has two barrier levels). There are also "knock-in" versions of these options in which the barrier event brings the option to life. Knock-in options can only produce a positive payoff a barrier is reached sometime during the option's life. Also, all of the above mentioned path dependencies are available for put style options. See Hull (2000) for further discussions on barrier options and certain parity relationships between knock-in and knock-out options. A knock-out option with a rebate is a simple modification of the regular knock-out barrier option. It pays some prespecified rebate R at the first instant the barrier level is reached during the life of the option. It gives some

compensation for the fact that the option has been knocked-out. The other class of options we will examine here are the lookback options. They also incorporate some path dependency and have the special property that they always expire in the money. A lookback call pays the difference between the terminal value of the underlying and its minimum over the life of the option. A lookback put pays the difference between the maximum of the underlying over the life of the option and its terminal value. It is worth mentioning here that there are further modifications to these options which are used in practice whose pricing can be easily accommodated by the methods used here. For example, the "hot-dog" option is a double knock out barrier option whose barriers are time dependent. See, for example, Babsiri and Noel (1998) for various forms of these exotic options and their pricing under the Black-Scholes model. The approach taken here can be viewed as an extension of their methods to more general price processes which is made possible by the pathwise imputation method.

This analysis focuses on the knock-out style barrier options with the understanding that the knock-in options are treated in almost the same way. We first test the bridge simulation method in the no jump case comparing it to results from Boyle and Tian (1999) and Linestsky and Davydov (2001) who employ alternative methods for calculating prices. The type of option will dictate the method of simulation. For barrier options without rebates we use the method of conditional simulation which involves replacing our Monte Carlo estimator with a conditional expectation (see Glasserman 2000). For example, in the case of the down-and-out call option the payoff is given as

$$(S_T - X)^+ 1\{m_T > LB\},$$

where $m_T = \inf_{t\in[0,T]} S_t$. Suppose we condition on the times and values of the process corresponding to the skeleton path generated by the bridge sampling algorithm. Label the corresponding sigma algebra $\mathcal{F} = \sigma\{S_o, S_1, ..., S_n = S_T, \tau_1, ..., \tau_n\}$. Let m_i denote the minimum of the process over the ith subinterval. The conditional expectation of our estimator given $\mathcal{F}$ is calculated as

$$\begin{aligned} E\left[(S_T - X)^+ 1\{m_T > LB\}|\mathcal{F}\right] &= (S_T - X)^+ E\left[1\{m_T > LB\}|\mathcal{F}\right] \\ &= (S_T - X)^+ \prod_{i=1}^{n} E\left[1\{m_i > LB\}|S_{i-1}, S_i, \tau_{i-1}, \tau_i\right] \\ &= (S_T - X)^+ \prod_{i=1}^{n} P\{m_i > LB|S_{i-1}, S_i, \tau_{i-1}, \tau_i\} \\ &= (S_T - X)^+ \prod_{i=1}^{n} p_i, \end{aligned}$$

where the p_i are the probabilities that the process does not breach the barrier in the ith subinterval. After a bridge path has been simulated the transformed path can be treated as if it came from a Brownian motion allowing the $p_i's$ to be calculated using results concerning a Brownian bridge. For this example

$$p_i = \begin{cases} 1 - \exp\left[-\frac{2(H-y_i)(H-y_{i-1})}{\tau_i - \tau_{i-1}}\right], & -H > \max(-y_{i-1}, -y_i) \\ 0, & \text{otherwise} \end{cases},$$

where $y_i = \frac{1}{\sigma}\left(1 - \frac{\alpha}{2}\right)^{-1} S_{\tau_i}^{1-\frac{\alpha}{2}}$ and $H = \frac{1}{\sigma}\left(1 - \frac{\alpha}{2}\right)^{-1} LB^{1-\frac{\alpha}{2}}$. Since in the case of $0 \leq \alpha < 2$ we generate the endpoint of the process according to Geometric Brownian Motion the estimator must also be adjusted by the appropriate likelihood weight. The Up-and-Out and double barrier options are priced in much the same way by using the appropriate containment probabilities shown in previous sections. The lookback and rebate options are priced by producing skeleton paths whose functionals (namely the extremes and the hitting time respectively) can be analyzed

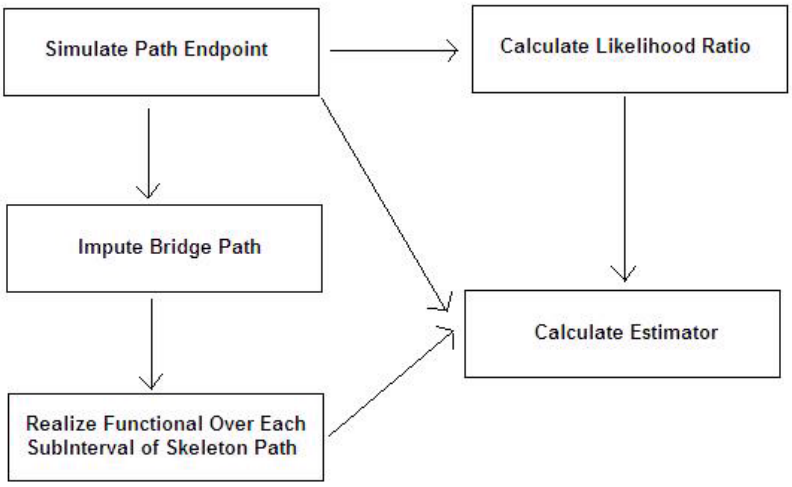

Figure 4.1: Flow chart of Monte Carlo simulations.

using Brownian bridge techniques. Regardless of option type, all paths will be generated under the general scheme outlined in Figure 4.1.

A path endpoint is generated assuming the Black-Scholes model from which a likelihood ratio can be calculated and a bridge path imputed. From the simulated skeleton bridge path the appropriate functional of the path can be generated or calculated. For barrier options this will involve calculating a containment probability. The inclusion of rebates requires the hitting time to the barrier level to be simulated. Pricing lookbacks necessitates generation of the maximum or minimum of the process. The results are then aggregated to produce the final estimator. The form of the Monte Carlo estimators used are given in Table 4.1. The subscript i denotes the ith replication in the Monte Carlo experiment and the variables are as follows:

Option Type	Estimator
Barrier - Call	$\frac{1}{N}e^{-rT}\sum_{i=1}^{N}(S_T^i - X)^+ P_i LR_i$
Barrier - Put	$\frac{1}{N}e^{-rT}\sum_{i=1}^{N}(X - S_T^i)^+ P_i LR_i$
Barrier with Rebate - Call	$\frac{1}{N}\sum_{i=1}^{N}\left[\begin{array}{c} e^{-rT}(S_T^i - X)^+ 1\{\tau_i > T\} \\ +e^{-r\tau_i}R1\{\tau_i \le T\} \end{array}\right] LR_i$
Barrier with Rebate - Call	$\frac{1}{N}\sum_{i=1}^{N}\left[\begin{array}{c} e^{-rT}(X - S_T^i)^+ 1\{\tau_i > T\} \\ +e^{-r\tau_i}R1\{\tau_i \le T\} \end{array}\right] LR_i$
Lookback - Call	$\frac{1}{N}e^{-rT}\sum_{i=1}^{N}(S_T^i - m_T^i)^+ LR_i$
Lookback - Put	$\frac{1}{N}e^{-rT}\sum_{i=1}^{N}(M_T^i - S_T^i)^+ LR_i$

Table 4.1: Monte Carlo Estimators

X - strike price

S_T - terminal stock price

LR - likelihood ratio

P - probability that the barrier level has not been breached on run i

τ - first hitting time to the barrier level

m_T - minimum price level over $[0, T]$

M_T - maximum price level over $[0, T]$.

Note that the above estimators have the same form in the context of jump diffusions. Of course for jump diffusions additional computations need to be made. For example, the likelihood ratio for a jump diffusion path will be the product of

the likelihood ratios pertaining to the simulation of the process just prior to each of the jump instances. This involves a random number of terms depending on the number of jumps which has taken place.

Pricing Results

Tables 4.2-4.7 display pricing results for the various options under several chosen parameter values which were adopted from previous studies to provide a basis for comparison. For all of the options tested the initial stock value was taken to be $S_0 = 100$, the strike price $X = 100$, the risk free rate $r = 0.1$ and the time to maturity $T = 0.5$. For the call (put) style options the lower and upper bounds are given respectively by $LB = 95$ (70) and $UB = 140$ (125). The rebate value is taken to be $R = 1$. 1,000,000 simulated paths were used to price each option. For each option and α value the corresponding cell in the table contains three figures: the price, the standard error in round brackets, and the CPU time in seconds contained in square brackets. The right hand column of each table provides the maximum percentage difference across all values of alpha for each particular option. With reasonable standard errors these prices match those of Boyle and Tian (1999) who arrive at prices using a trinomial tree approach. Observe that the CPU time for the Black-Scholes model ($\alpha = 2$) is only a fraction of the CPU time used for the other α values. This is due to the fact that i) importance sampling is not required for this case, and more importantly ii) the Black-Scholes dynamics can be inverted with respect to the driving Brownian motion so that Brownian bridge techniques can be used directly (by-passing the need for bridge imputation). Also notice that across all options the CPU times for the case $\alpha = 0$ are consistently larger than the

other α ($\neq 2$) values. This is a result of using a perfect imputation scheme based on the Bessel bridge dynamics which is fully described in Beskos and Roberts (2005). As was mentioned, for the other α values we truncate the intensity function used for imputation. The fact that these prices agree so closely with those of Boyle and Tian (1999) is a testament to validity of the truncation procedure. In terms of pricing accuracy truncation seems to work just as well as the "exact" simulation method used for the absolute model ($\alpha = 0$). This suggests that the approximation of the intensity function used for imputation is a rather benign process.

To greater appreciate the speed and accuracy of the pathwise imputation pricing method we consider Monte Carlo pricing under a simulation driven by a traditional Euler scheme. Pricing barrier options using an ordinary Euler discretization is known to produce two sources of bias. The first being the bias introduced by discretizing a continuous process over a finite grid. The second is that of discretely monitoring the barrier. When a process is approximated over a finite grid information about the process between the grid points is lost. For example, the process at two successive time grid points may lie below an upper barrier while the path connecting these points breaches the barrier. This leads to an underestimation of the number times the barrier is breached. As an example consider the Up-and-Out call option in the case of $\alpha = 1$. This option was priced using an Euler scheme with a time step of 1 day and 1,000,000 simulated paths. Boyle and Tian (1999) give the price of this option as 7.0372. This price is confirmed by the pathwise imputation method which gives a price of 7.0314 with a standard error of 0.009 in 217 seconds. The simple Euler scheme applied to this option gives a price of 7.2226 with a standard error of 0.010 in 366 seconds. That translates to a rel-

ative pricing error in excess of 2.7% with a running time about 69% longer than the pathwise imputation Monte Carlo method. There are a variety of methods which can be introduced to reduce the effect of discretely monitoring the barrier. The naive method simply approximates the process with a finer time grid which is computationally demanding and not very rewarding. A better method is to approximate the process between the grid points by a Geometric Brownian motion so that the probability of the barrier being breached can be approximated using (4.1). Modifying the Euler method with this approximation greatly reduces the bias of the method giving a price of 6.9940 with a standard error of 0.009 and a running time of 484 seconds. Even with this modification, the pathwise imputation method is still superior both in terms of accuracy and running time. Similar conclusions hold for the other options considered here. It is important to note that pathwise imputation can be used in conjunction with an Euler scheme to reduce and possibly eliminate the bias involved in a monitoring a continuous barrier discretely. Instead of assuming the process between the Euler time points is Geometric Brownian motion pathwise imputation can be used to examine the process between grid points.

Even in the worst cases, the CPU times for the pure diffusion CEV model are very low giving standard errors that are (for most cases) less than a cent. By running a tenth of the simulations considered in this study, standard errors of 1-3 cents can be achieved in about 20-30 seconds of running time for the options considered here. Under the jump diffusion dynamics only a subset of the options and α values are tested. The same pure diffusion model parameters are used with the addition of the jump component. A certain consistency is observed. For

example, consider the Up-and-Out call option. The price is decreasing in λ which makes sense since higher λ values increase the variability of the process which increase the probability that the option will be knocked-out, hence decreasing its value. The price trends across λ values of the other options can be interpreted similarly. It is also of interest to examine the effect of the jump parameters on prices while holding the overall variance of the process fixed. Such a study will be carried out in the next section. Note that pricing under the similar jump diffusion dynamics requires about 10 times the CPU time. These seem unusually longer than the pure diffusion case since a jump diffusion path with the chosen parameters produces on average only about 2-3 times more calculation per path. This large difference is due to the programming environment. MATLAB 7.1 was used for the implementation. Due to the nature of the jump diffusion paths their construction (including likelihood ratios) must be done using "for" loops which is known to be a great disadvantage in MATLAB. Similar calculations in the pure diffusion model are easy to "vectorize" producing running times closer to what one would observe in a lower level programming language such as C or C++. In general, keeping other parameters fixed, increasing λ will increase the CPU time. Higher λ values lead to more jumps per path on average and hence more CPU calculation.

Knock-Out Options						
	α					
	0	0.5	1	1.5	2	max. diff.
DOC	5.6593 (0.008) [456]	5.6459 (0.008) [220]	5.6527 (0.009) [217]	5.66 (0.009) [285]	5.6623 (0.010) [32]	0.29%
UOC	7.7043 (0.010) [393]	7.3839 (0.010) [219]	7.0314 (0.009) [217]	6.6875 (0.009) [285]	6.3222 (0.008) [32]	21.86%
DKOC	4.1298 (0.006) [500]	3.883 (0.006) [328]	3.6411 (0.005) [327]	3.4022 (0.005) [446]	3.1609 (0.004) [127]	30.65%
DOP	3.1438 (0.005) [388]	3.3359 (0.005) [219]	3.5405 (0.005) [216]	3.7187 (0.006) [283]	3.8975 (0.006) [32]	23.97%
UOP	4.6714 (0.015) [388]	4.6818 (0.012) [218]	4.6845 (0.009) [215]	4.6805 (0.008) [282]	4.6754 (0.008) [32]	1.31%
DKOP	3.1225 (0.005) [525]	3.3283 (0.005) [361]	3.5162 (0.005) [362]	3.6964 (0.006) [518]	3.8815 (0.006) [153]	24.31%

Table 4.2: CEV Knock-Out Option Results

Lookbacks						
	α					
	0	0.5	1	1.5	2	max. diff.
Lookback Call	16.123 (0.012) [438]	15.999 (0.012) [266]	15.902 (0.012) [263]	15.74 (0.012) [343]	15.643 (0.013) [75]	3.07%
Lookback Put	11.278 (0.015) [437]	11.474 (0.015) [266]	11.724 (0.010) [263]	11.989 (0.009) [343]	12.28 (0.009) [75]	8.85%

Table 4.3: CEV Lookback Option Results

Instantaneous Variance Distribution in the CEV JD Model

Thus far option prices have been analyzed by fixing a Black-Scholes volatility and comparing CEV prices with equivalent instantaneous variances for different alpha values. It was found that for some options there are considerable price differences for different values of alpha, confirming the results of Boyle and Tian (1999) and Davydov and Linetsky (2001). As a result of this, pricing and hedging schemes based on the model are alpha dependent. With the model choice one should also carefully consider model risk when pricing particular options. We wish to extend this analysis to the case of CEV jump diffusions to determine the effects and relationships between the various jump parameters and alpha level. For example, in the jump diffusion setup a certain amount of variation in the stock price can be attributed to the jump component of the model. Within the jump component variation is dictated by jump parameters, namely the intensity of jump arrivals

Knock-Out Options with Rebates						
	α					
	0	0.5	1	1.5	2	max. diff.
DOC	6.3480 (0.010) [425]	6.3636 (0.011) [252]	6.3510 (0.011) [250]	6.3776 (0.011) [317]	6.3875 (0.012) [68]	0.62%
UOC	7.7160 (0.011) [394]	7.4342 (0.010) [221]	7.0925 (0.010) [219]	6.7596 (0.010) [289]	6.4187 (0.009) [36]	20.21%
DOP	3.1992 (0.005) [388]	3.3889 (0.006) [218]	3.5696 (0.006) [217]	3.7533 (0.006) [285]	3.9291 (0.006) [34]	22.82%
UOP	4.9398 (0.023) [396]	4.9348 (0.021) [226]	4.9294 (0.009) [224]	4.9374 (0.008) [297]	4.9340 (0.007) [45]	0.21%

Table 4.4: CEV Knock-Out Option with Rebate Results

CEV Jump Diffusion: Knock-Out Options					
		α			
	λ	0	1	2	max. diff.
DOC	1	5.8088 (0.010) [2079]	5.8192 (0.010) [1713]	5.8305 (0.010) [189]	0.37%
	2	5.9811 (0.012) [2663]	5.9502 (0.011) [2196]	5.977 (0.011) [192]	0.52%
UOC	1	7.3391 (0.010) [2077]	6.7123 (0.009) [1709]	6.0478 (0.008) [181]	21.35%
	2	6.9787 (0.010) [2667]	6.3857 (0.009) [2188]	5.7639 (0.008) [192]	21.08%
DKOC	1	3.736 (0.006) [2239]	3.2957 (0.005) [1882]	2.8539 (0.004) [293]	30.91%
	2	3.3887 (0.006) [2874]	2.9813 (0.005) [2397]	2.5837 (0.004) [312]	31.16%

Table 4.5: CEV Jump Diffusion Knock-Out Option Results

CEV Jump Diffusion: Lookbacks					
		α			
	λ	0	1	2	max. diff.
Lookback Call	1	16.851 (0.014) [2130]	16.57 (0.013) [1769]	16.296 (0.019) [249]	3.41%
	2	17.565 (0.017) [2733]	17.278 (0.015) [2270]	16.931 (0.015) [286]	3.74%
Lookback Put	1	12.024 (0.019) [2129]	12.538 (0.012) [1767]	13.106 (0.009) [248]	9%
	2	12.810 (0.037) [2733]	13.301 (0.012) [2268]	13.922 (0.010) [284]	8.68%

Table 4.6: CEV Jump Diffusion Lookback Option Results

CEV Jump Diffusion: Barriers with Rebates					
		α			
	λ	0	1	2	max. diff.
UOC	1	7.656 (0.011) [2124]	7.0164 (0.010) [1763]	6.3292 (0.009) [186]	20.96%
	2	7.6148 (0.012) [2731]	6.9369 (0.011) [2272]	6.2326 (0.010) [199]	22.18%
DOC	1	6.5216 (0.012) [2142]	6.5392 (0.012) [1785]	6.5428 (0.012) [215]	0.33%
	2	6.6696 (0.013) [2733]	6.6757 (0.013) [2272]	6.7256 (0.013) [227]	0.84%

Table 4.7: CEV Jump Diffusion Knock-Out Option with Rebate Results

and the jump size parameters. Are prices dependent on the different combinations of these parameters which lead to the same overall variation? If so, are these dependencies significant? We aim to address these and similar questions here. In order for our comparisons to be meaningful we will fix the equivalent Blacks-Scholes volatility to 25%. In the previous analysis the σ parameter of the CEV model was chosen to achieve the required initial instantaneous volatility through the formula $\sigma = \sigma_{BS} S_o^{1-\frac{\alpha}{2}}$. This equation which is taken from Macbeth and Merville (1980) is simply a comparison of instantaneous return volatilities. Writing the CEV model in terms of instantaneous returns we arrive at a specification of the form

$$\frac{dS_t}{S_t} = rdt + \sigma S_t^{\frac{\alpha}{2}-1} dW_t.$$

An equivalent Black-Scholes volatility of σ_{BS} for a given level of α is achieved by matching the above diffusion term to the required volatility since the diffusion term is indeed the instantaneous volatility of the CEV model. In the presence of jumps an equivalent formulation is given by the stochastic differential equation:

$$\frac{dS_t}{S_{t-}} = \mu dt + \sigma S_{t-}^{\frac{\alpha}{2}-1} dW_t + dJ_t.$$

To match instantaneous variances (or volatilities) under this specification we need to consider the variation of the jump component as well as the variation in the continuous part of the equation. Before continuing we will make the assumption that when a multiplicative jump occurs it has a mean of 1. Since Y_j is lognormal with $\log Y_j \sim N(\upsilon, \delta^2)$ we know that

$$E\left[Y_j\right] = \exp\left(\nu + \delta^2/2\right),$$

and

$$Var\left[Y_j\right] = \exp\left(2\nu + 2\delta^2\right) - \exp\left(2\nu + \delta^2\right).$$

Thus, $E[Y_j] = 1$ can be achieved by setting $v = -\delta^2/2$. This also simplifies the variance term giving $\sigma_Y^2 := Var[Y_j] = \exp(\delta^2) - 1$. This will reduce the number of jump parameters to two, the intensity of arrivals λ and the jump size variance σ_Y^2. In this framework, the variance of the jump component J_t can be derived as follows

$$\begin{aligned} Var[J_t] &= E[Var[J_t|N_t]] + Var[E[J_t|N_t]] \\ &= E\left[VAR\left[\sum_{j=1}^{N_t}(Y_j - 1)|N_t\right]\right] + Var\left[E\left[\sum_{j=1}^{N_t}(Y_j - 1)|N_t\right]\right] \\ &= E\left[\sigma_Y^2 N_t\right] + 0 \\ &= \sigma_Y^2 \lambda t. \end{aligned}$$

Thus, we have the informal relation that $Var[dJ_t] = \sigma_Y^2 \lambda$. In sum, the total instantaneous return variance of the CEV model is then given by

$$Var\left[\frac{dS_t}{S_{t-}}\right] = \sigma^2 S_t^{\alpha-2} + \sigma_Y^2 \lambda.$$

For given values of α and say λ there is no unique way to choose σ^2 and σ_Y^2 to achieve a specified equivalent Black-Scholes volatility of σ_{BS}. Since we wish to analyze the contribution of the variance of the jump component it makes sense to also fix the proportion of variance coming from the both the continuous and jump components. Let $0 \leq \varepsilon < 1$ be the proportion of instantaneous variance coming from the jump component of the model with the convention that $\varepsilon = 0$ corresponds the regular (no-jump) CEV model. Under this framework one can control the componentwise variance of the process. For given levels of α and λ we choose σ^2 and σ_Y^2 according the following relations $\sigma^2 S_t^{\alpha-2} = (1-\varepsilon)\sigma_{BS}^2$ and $\sigma_Y^2 \lambda = \varepsilon \sigma_{BS}^2$.

These can be summarized as

$$\sigma^2 = (1-\varepsilon)\,\sigma_{BS}^2 S_t^{2-\alpha}$$

and

$$\sigma_Y^2 = \sigma_{BS}^2 \frac{\varepsilon}{\lambda}.$$

In this way the total instantaneous return variance is equal to σ_{BS}^2. The second equation can be written in terms of the log jump size variance as $\delta^2 = \log\left(\sigma_{BS}^2 \frac{\varepsilon}{\lambda} + 1\right)$. Notice that the jump size variance is inversely proportional to the arrival intensity of the jumps. As $\lambda \to \infty$ the jumps become infinitely frequent and small with the jump component converging to a continuous process in the limit. As $\lambda \to 0$ the jumps become very infrequent with highly variable magnitudes. For different choices of α, λ and ε we can analyze the effects of the variance distribution on option pricing.

To this end we analyze the effects on the price of an Up-and-Out Call option with a 3 month maturity and the same parameters as given in the previous sections. No papers have previously studied the CEV Jump Diffusion model making it difficult to specify exactly what an appropriate range of ε should be. As a basis for comparison we examine Merton's model (the Black-Scholes model with constant intensity lognormal jumps). We note at the onset that although it has been extensively studied in the literature, several authors point out difficulties in its estimation and calibration. Statistical estimation is difficult since with observations at discrete times distinguishing small jumps from the continuous part of the process is no easy task. This problem has been compared to that of estimating an infinite mixture model which is known to be ill posed for Likelihood estimation. Vetzal et. al. (2005) point out that calibration from option quotes also presents

difficulties since the calibration error surface may contain large flat portions near its minimum. As a result, several distinct combinations of parameters will give sufficiently small pricing error. Benzoni et. al. (2002) estimate Merton's model using historical S&P 500 data. Their results indicate that a negligible amount of variability in the process is due to the jump component. On the other hand, Andersen and Andreasen (2000) calibrate the model to S&P option quotes. Their results indicate that as much as 50% of the variability in the process is due to the jump component of the process. Based on these studies we examine Up-and-Out call option prices for three different ε values: 10%, 25% and 50%.

Figure 4.2 shows Up-and-Out call prices with a 3 month maturity as a function of λ for each ε value and α values of 0, 1 and 2. For each λ, α and ε, σ_Y^2 is chosen to ensure an overall instantaneous return volatility of $\sigma_{BS} = 25\%$ using the procedure described above. In the pure diffusion (no jump) model we find that the maximum relative price difference across all alpha values is just over 5%. Note that this is about one fourth of the maximum price difference observed for the equivalent option with a 6 month expiry indicating a substantial time dependence on the pricing differences. Across all (α, λ) combinations we find that the maximum relative price difference is increasing in ε. For $\varepsilon = 10\%$, 25% and 50% the estimated maximum difference is 5.75%, 8.44% and 17.90%, respectively. For equivalent instantaneous volatilities increasing the proportion of variation in the process due to jumps can have a great impact on the range of prices dictated by the model. Figure 4.2 shows that for fixed ε and α smaller values of λ give smaller Up-and-Out call prices. The intuition behind this is as follows. As $\lambda \longrightarrow 0$ the jump size variance $\sigma_Y^2 \longrightarrow \infty$. For small λ there are usually no jumps. When a jump occurs it is highly likely to

cause a massive increase or decrease in the level of the process. In either case, the Up-and-Out call option expires worthless. So although the instantaneous volatility is constant as $\lambda \longrightarrow 0$ the nature of the jumps coupled with the form of the payoff function produces more sample paths of zero expiry. As a rule of thumb, empirical observation suggests that as $\lambda \longrightarrow 0$ the limiting value of the option is the value of the option priced under the pure continuous process with instantaneous return variance equivalent to $(1-\varepsilon)\sigma_{BS}^2$. As $\lambda \longrightarrow \infty$ the jump size variance $\sigma_Y^2 \longrightarrow 0$. So for large λ the process admits many jumps of small magnitudes. In the limit as $\lambda \longrightarrow \infty$ this the jump component forms a continuous process. This leads to the result that the price of the Up-and-Out call under the jump diffusion converges to the value of the option priced under the continuous part of the process with instantaneous return volatility equivalent to σ_{BS}^2. Thus, for fixed σ_{BS} we have a means of producing bounds on the jump diffusion option price using only prices from the continuous model. This has great intuitive appeal and implications for model risk management and analysis. Similar results hold for other types of barrier options. Those which expire worthless for large spot price deviations, such as the Double-Knock-Out Call and Down-and-Out put options, behave similarly.

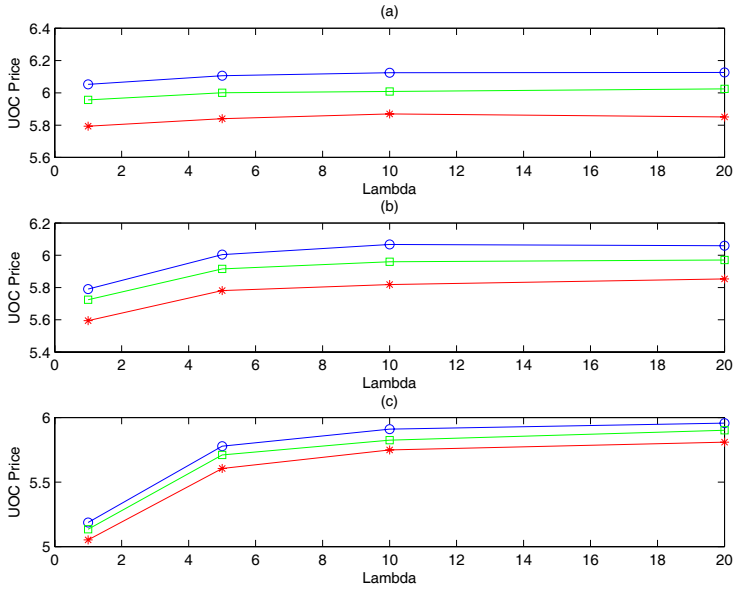

Figure 4.2: Up-and-Out Call Prices:

Parameters: $T = 3$ months, $r = 10\%$, $\sigma_{BS} = 25\%$, $S_0 = 100$, $UB = 140$

'o' - $\alpha = 0$, '$\square$' - $\alpha = 1$, '$*$' - $\alpha = 2$

(a) $\varepsilon = 10\%$

(b) $\varepsilon = 25\%$

(c) $\varepsilon = 50\%$

Chapter 5

Conclusions and Extensions

Imputation of large asynchronous datasets is a computationally demanding problem, especially in the case of diffusions where explicit transition densities and conditional densities are generally unknown. Presently, simulation, imputation and interpolation of diffusion processes relies almost exclusively on finite time-step Euler or Milstein approximations. The conditional distributions are thus only approximate, and in order to improve the approximation, values need to be imputed at a large number of intermediate points. We have demonstrated that imputation from the exact conditional distributions is feasible. For multivariate Brownian motion, the imputing process is just a collection of Brownian bridges which can be easily simulated. For more general diffusions, the bridge process is not tractable and we have introduced an efficient simulation method motivated by the general form of the transition density given by Dacunha-Castelle and Florens-Zmirou (1986). The imputation technique developed has strong connections to pathwise elements of stochastic processes, and allows bridge processes of general diffusions to be ex-

pressed in terms of Brownian bridges. The methods for diffusions were extended to jump diffusions in which the arrival intensity of jumps was independent of the driving Brownian motion. The most intriguing property of pathwise imputation is that it allows a simulated (jump) diffusion path to be completely characterized by a finite skeleton of points along with the rule of Brownian bridge interpolation. As a consequence, additional properties and functionals of the path can easily be deduced by appealing to the analogous Brownian bridge results. These properties make pathwise imputation an ideal approach for simulating the prices of path dependent options under quite general diffusion models. In particular, the methods discussed were applied to path stratification in the CIR interest rate model and path dependent option pricing in the CEV jump diffusion stock model with great success. Unbiased CEV barrier and lookback option prices were achieved in good time without appealing to Euler or other approximate time step methods.

It should be noted the that imputation methodology for diffusions introduced in Chapter 3 can be presented in more generality. The proposal process was taken to be a Brownian motion, but this is not a necessary restriction. This choice is based purely on convenience. For example, for positive diffusions one may consider a Geometric Brownian motion as the proposal process. This requires one to find a scalar transformation s of the diffusion $dX_t = \mu(X_t)dt + \sigma(X_t)dW_t$ such that $Y_t = s(X_t)$ is a diffusion process with diffusion term given by $Y_t dW_t$. Finding the appropriate transformation is a straight forward application of Ito's lemma and the inverse function theorem. Girsanov's theorem can then be used in the same way as before to obtain the density of Y_t with respect to Geometric Brownian motion. Geometric Brownian motion is also a convenient proposal process because a log

transformation of it yields Brownian motion, and so most properties of Brownian motion and Brownian bridges can be applied directly. In general, the proposal process can be any diffusion process which we can simulate from. We need only take an appropriate transformation of the target diffusion so that the diffusion coefficients of the proposal process and the transformed target process match. This requirement is due to the fact that diffusion processes with different diffusion coefficients induce mutually singular probability measures on $(C[0,T], \mathcal{B})$.

Another important extension of the simulation methodology is to diffusion processes with time dependent coefficients which are very popular in financial modelling. These types of processes are handled in almost exactly the same way except that an additional term appears in the resulting acceptance ratio. Consider a diffusion of the form $dY_t = a(Y_t, t)dt + dW_t$. In this case the Girsanov density is given by

$$\frac{dP_Y}{dP_W}(\omega) = \exp\left[\int_0^T a\left(\omega_u, u\right) d\omega_u - \frac{1}{2}\int_0^T a\left(\omega_u, u\right)^2 du\right].$$

Let $A(y,t) = \int^y a(z,t)dz$. Then by Ito's lemma we have that

$$A(Y_T, T) = A(Y_0, 0) + \int_0^T \frac{\partial A(Y_u, t)}{\partial y} dY_u + \frac{1}{2}\int_0^T \frac{\partial^2 A(Y_u, u)}{\partial y \partial y} du + \int_0^T \frac{\partial A(Y_u, u)}{\partial t} du$$

or

$$\int_0^T a\left(Y_u, u\right) dY_u = A(Y_T, T) - A(Y_0, 0) - \frac{1}{2}\int_0^T \frac{\partial a\left(Y_u, u\right)}{\partial y} du - \int_0^T \frac{\partial A(Y_u, u)}{\partial t} du.$$

Hence,

$$\frac{dP_Y}{dP_W}(Y) \propto \exp\left[-\frac{1}{2}\int_0^T \left(a\left(Y_u, u\right) + a\left(Y_u, u\right)^2 + 2\frac{\partial A(Y_u, u)}{\partial t}\right) du\right]$$

The Girsanov density is almost the same as in the homogeneous coefficients case, the only difference being the addition of time derivative term in the exponent. Assuming the necessary boundedness conditions hold with the introduction of this term the method works in the exact same way as previously described.

Bibliography

[1] Y. Ait-Sahalia and P. Mykland (2004). Estimators of Diffusions with Randomly Spaced Discrete Observations: A General Theory. Annals of Statistics, Vol. 32, No. 5, 2186–2222.

[2] L. Andersen and J. Andreasen (2000). Volatility skews and extensions of the Libor market model. Applied Mathematical Finance, 7, 1-32.

[3] L. Andersen and J. Andreasen (2001). Jump-Diffusion Processes: Volatility Smile Fitting and Numerical Methods for Option Pricing. Review of Derivatives Research, 4, 231-262.

[4] T. Andersen, L. Benzoni and J. Lund (2002). An Empirical Investigation of Continuous-Time Equity Return Models. The Journal of Finance, Vol.LVII, No.3.

[5] M. Babsiri and G. Noel (1998). Simulating Path-Dependent Options: A New Approach. Journal of Derivatives, Winter 1998;6,2,p.65.

[6] J. Bertoin, J. Pitman and J. Ruiz de Chavez (1999). Constructions of a Brownian Path with a Given Minimum. Elect. Comm. in Probab. 4 31-37.

[7] Beckers. S. (1980). The Constant Elasticity of Variance Model and Its Implications For Option Pricing. The Journal of Finance, Vol.35, No.3, 661-673.

[8] Beskos & Roberts (2005). Exact simulation of diffusions. The Journal of Applied Probability Vol. 15, No. 4, November.

[9] Beskos, Fearnhead, Papaspiliopoulos & Roberts (2006a). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes. J.R. Statist. Soc. B 68, part 2, pp. 1-29.

[10] Beskos, Papaspiliopoulos & Roberts (2006b). Retrospective exact simulation of diffusion paths with applications. Working paper.

[11] P. Billingsley (1968). Convergence of Probability Measures. Wiley, New York.

[12] M. Broadie, P. Glasserman and S. Kou (1997). A Continuity Correction for Discrete Barrier Options. Vol.7, No.4,325-348.

[13] M. Broadie and O. Kaya (2004). Exact Simulation of Stochastic Volatility and other Jump Diffusion Processes. Working paper (Sept. 29, 2004.).

[14] P. Boyle (1977). Options: A Monte Carlo Approach. The Journal of Financial Economics. 4, 323-338.

[15] P. Boyle and Y. Tian (1999). Pricing lookback and barrier options under the CEV process. The Journal of Financial and Quantitative Analysis, Vol.34, No.2, 241-264.

[16] P.Carmona, F. Petit, J.Pitman and M. Yor (1998). On the Laws of Homogeneous Functionals of the Brownian Bridge. Working paper.

[17] G. Celeux & J. Diebolt (1985) The SEM algorithm : A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. C.S.Q. Vol 2, 73-82.

[18] K. Chan, A. Karolyi, F. Longstaff & A. Sanders (1992). An Empirical Comparison of Alternative Models of the Short-Term Interest Rate. The Journal of Finance, Vol. 47, Issue 3 p. 1209.

[19] J. Cox, J. Ingersoll & S. Ross (1985) A Theory of the Term Structure of Interest Rates. Econometrica, 53, 2, 385-408.

[20] D. Dacunha-Castelle & D. Florens-Zmirou (1986). Estimation of the Coefficients of a diffusion from discrete observations Stochastics 19, 263-284.

[21] D. Davydov and V. Linetsky (2001). Pricing and Hedging Path-Dependent Option Under the CEV Process. Management Science, Vol.47, No.7, 949-965.

[22] B. Delyon (1996). General Results on the Convergence of Stochastic Algorithms. IEEE Transaction on Automatic Control, Vol. 41, No.9.

[23] B. Delyon, M. Lavielle & E. Moulines (1999) Convergence of a Stochastic Approximation Version of the EM Algorithm Ann. Statist. 27. 94-128

[24] A. P. Dempster, N.M. Laird, and D.B. Rubin (1977). Maximum Likelihood from incomplete data via the EM algorithm. J. of the Royal Statistical Society, Series B, 34:1-38.

[25] D. Emanuel and J. MacBeth (1982). Further Results on the Constant Elasticity of Variance Call Option Pricing Model. The Journal of Quantitative Financial Analysis, Vol.17, No.4, 533-554.

[26] O. Elerian, S. Chib & N. Shepard (2001). Likelihood inference for discretely observed non-linear diffusions. Econometrica, 69, 959-993.

[27] B. Eraker (2001). MCMC Analysis of Diffusion Models with Applications to Finance Journal of Business and Economic Statistics, 19, 2, 177-191

[28] D. Florens-Zmirou (1989). Approximate Discrete-Time Schemes for Statistics of Diffusion Processes Statistics 20, 547-557

[29] P. Glasserman (2000). Monte Carlo Methods in Financial Engineering. Springer-Verlag, New York.

[30] P. Glasserman and M. Broadie (1996). Estimating Security Price Derivatives Using Simulation. Management Science, Vol.42, No.2.

[31] P. Glasserman, P. Heidelberger and P. Shahabuddin (1999). Asymptotically Optimal Importance Sampling and Stratification for Pricing Path-Dependent Options. Mathematical Finance, Vol.9, No.2, 117-152.

[32] J. Hull (2000). Option, Futures, and Other Derivatives (4th Edition). Prentice Hall, NJ.

[33] J. Jacod and A. Shiryaev (1980). Limit Theorems for Stochastic Processes. Springer-Verlag.

[34] M. Jeanblanc, J. Pitman and M.Yor (1996). The Feyman-Kac formula nd decomposition of Brownian paths. Technical Report No.471, Department of Statistics, University of Calcifornia.

[35] M. Johannes & N. Polson (2003). MCMC Methods for Financial Econometrics. To appear: Handbook of Financial Econometrics.

[36] Ioannis Karatzas and Steven Shreve (1991). Brownian Motion and Stochastic Calculus. Springer-Verlag, New York.

[37] P. Kofman & I. Sharpe (2003). Using Multiple Imputation in the Analysis of Incomplete Observations in Finance. J. Financial Econometrics 1: 216-249.

[38] S. Kou (2002). A Jump-Diffusion Model for Option Pricing. Management Science, Vol.48, No.8, 1086-1101.

[39] V. Linetsky (2004). Computing hitting tim densities for CIR and OU diffusions: applications to mean reverting models. Journal of Computational Finance, 7, 1-22.

[40] R. Little & D. Rubin (2002). Statistical Analysis with Missing Data, 2nd Edition. Wiley, New York.

[41] N. K. Malhotra (1987), Analyzing Market Research Data with Incomplete Information on the Dependent Variable, J. of Marketing Research 24, 74-84.

[42] J. Macbeth and L. Merville (1979). Testing of the Black-Scholes and Cox Call Option Valuation Models. The Journal of Finance, Vol.35, No.2, 285-301.

[43] McLeish, D (2005). Monte Carlo Simulation and Finance. Wiley, New Jersey.

[44] R. Merton (1976). Option Pricing when Underlying Stock Returns are Discontinuous. Journal of Financial Economics, 3, 125-144.

[45] S. Metwally and A. Atiya (2002). Using Brownian Bridge for Fast SImulation of Jump-Diffusion Processes and Barrier Options. Journal of Derivatives, Fall 2002, 43-54.

[46] Nevel'son and Has'minskii (1976). Stochastic Approximation and Recursive Estimation. American Mathematical Society, Providence Rhode Island.

[47] J. Nicolau (2002). A new technique for simulating the likelihood of stochastic differential equations Econometrics Journal 5, 91-103

[48] S. Nielsen (2000). On simulated EM algorithms. J. of Econometrics, 96, 267-292.

[49] A. Pedersen (1995). A new approach to maximum likelihood estimation for stochastic differential equations based on discrete observations. Scandinavian Journal of Statistics 22, 55-71.

[50] J. Pitman and M. Yor (1999). The Law of the Maximum of a Bessel Bridge. Electronic Journal of Probability, Vol.4 no. 15.

[51] D. Revuz and M. Yor (1994). Continuous Martingales and Brownian Motion. Springer-Verlag.

[52] G. Roberts & O. Stramer (2001). On inference for nonlinear diffusion models using the Hastings-Metropolis algorithms. Biometrika 88, 603–621.

[53] J. Schafer (1997). Analysis of Incomplete Multivariate Data. Chapman&Hall, New York.

[54] M. Schroder (1989). Computing the Constant Elasticity of Variance Option Pricing Formula. The Journal of Finance, VolXLIV, No.1.

[55] M. Steele (2000). Stochastic Calculus and Financial Applications. Springer-Verlag, New York.

[56] Peter Tankov and Rama Cont (2004). Financial Modelling with Jump Diffusions. Chapman and Hall.

[57] N. Wang & J. Robins (1998). Large-sample theory for parametric multiple imputation procedures. Biometrika 85, 935-948

[58] G. Wei & M.Tanner (1990). A Monte-Carlo implementation of the EM algorithm and the Poor Man's data augmentation algorithm. J. Amer. Statist. Assoc. 85, 699-704.

Chapter 6

Appendix

6.1 Acceptance-Rejection Sampling

There are many ways to describe acceptance-rejection sampling. The following description is taken from Beskos and Roberts (2005). Let $(S, \mathcal{S})$ be a sufficiently regular measurable space and ν,μ probability measures on it such that μ is absolutely continuous with respect to υ. Assume that there exists $\varepsilon > 0$ such that $f := \epsilon \frac{d\mu}{d\upsilon} \leq 1 \ \upsilon - a.s.$ and that it is easy to sample from υ. Then the following proposition can be used to return draws from μ.

PROPOSITION. Let $(Y_n, I_n)_{n \geq 1}$ be a sequence of i.i.d. random elements taking values in $S \times \{0, 1\}$ such that $Y_1 \sim \upsilon$ and $P\{I_1 = 1 | Y_1 = y\} = f(y)$ for all $y \in S$. Define $\tau = \min\{i \geq 1 : I_i = 1\}$. Then $P\{Y_\tau \in dy\} = \mu(dy)$.

6.2 Transition Densities of Diffusion Processes

The following lemmas appear in D. Dacunha-Castelle and D. Florens-Zmirou (1986) but contain errors. We include correct versions here.

Lemma 1. (D. Dacunha-Castelle and D. Florens-Zmirou) Consider a one dimensional diffusion model of the form

$$dY_t = b_\theta(Y_t)\,dt + \sigma dW_t$$

where $Y_0 = x \in \Re, b_\theta : \Re \to \Re$ is a drift function with unknown parameter θ, $\sigma > 0$ is a constant diffusion coefficient and W is a standard Brownian motion. Let $G(x) = \int_0^x b_\theta(u)\,du$. Let B be a Brownian bridge, $E[B_t^2] = t(1-t), t \in [0,1]$. Denote

$$g = -\frac{1}{2}\left(\frac{b_\theta^2}{\sigma^4} + \frac{b_\theta'}{\sigma^2}\right).$$

Let p_τ be the transition density of the diffusion defined above. Suppose $|g_\alpha(x)| = O\left(|x|^2\right)$ for $|x| \to \infty$. Then:

$$p_\tau(x,y) = \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left(-\frac{(x-y)^2}{2\sigma^2\tau} + \frac{G(y) - G(x)}{\sigma^2}\right)$$
$$\times E\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_u(x,y) + \sqrt{\sigma^2\tau}B_u\right)du\right)\right]$$

where $z_u(x,y) = (1-u)x + uy$.

Proof. For a diffusion process Z denote by μ_Z^x the measure induced by Z when $Z_0 = x$. Girsanov's theorem gives the density L_τ^x of the measure μ_Y^x with respect to the Wiener measure μ_W^x:

$$L_\tau^x = \frac{d\mu_Y^x}{d\mu_W^x} = \exp\left(\int_0^\tau \frac{b(Y_u)}{\sigma^2} dY_u - \frac{1}{2}\int_0^\tau \frac{b^2(Y_u)}{\sigma^2} du\right).$$

Let $\phi : \Re \to \Re$ be a Borelian bounded function, then we have

$$E_{\mu_Y^x}\left[\phi\left(Y_\tau\right)\right] = E_{\mu_W^x}\left[\phi\left(Y_\tau\right) L_\tau^x\right] = E_{\mu_W^x}\left[E\left[\phi\left(Y_\tau\right) L_\tau^x | Y_\tau\right]\right] = E_{\mu_W^x}\left[\phi\left(Y_\tau\right) E_{\mu_W^x}\left[L_\tau^x | Y_\tau\right]\right]$$
$$= \int \phi\left(y\right) E_{\mu_W^x}\left[L_\tau^x | Y_\tau = y\right] \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left[-\frac{\left(x-y\right)^2}{2\sigma^2\tau}\right] dy.$$

From this equation we see that the transition density of $Y_\tau | Y_0 = x$ is given by

$$p_\tau(x,y) = \Lambda_\tau\left(x,y\right) \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left[-\frac{\left(x-y\right)^2}{2\sigma^2\tau}\right].$$

where $\Lambda_\tau\left(x,y\right) = E_{\mu_W^x}\left[L_\tau^x | Y_\tau = y\right]$. It now remains to find an expression for the expectation term $\Lambda_\tau\left(x,y\right)$. By Ito's lemma we have that

$$dG\left(Y_\tau\right) = G'\left(Y_\tau\right) dY_\tau + \frac{\sigma^2}{2} dG''\left(Y_\tau\right) d\tau$$

, and so

$$G\left(Y_\tau\right) = G\left(x\right) + \int_0^\tau b\left(Y_u\right) dY_u + \frac{\sigma^2}{2} \int_0^\tau b'\left(Y_u\right) du,$$

hence

$$\int_0^\tau b\left(Y_u\right) dY_u = G\left(Y_\tau\right) - G\left(x\right) - \frac{\sigma^2}{2} \int_0^\tau b'\left(Y_u\right) du.$$

This integral also appears in the formula for L_τ^x, substituting for this term yields an expression for L_τ^x involving no stochastic integrators:

$$L_\tau^x = \exp\left(\frac{1}{\sigma^2}\left(G\left(Y_\tau\right) - G\left(x\right) - \frac{\sigma^2}{2}\int_0^\tau b'\left(Y_u\right) du\right) - \frac{1}{2}\int_0^\tau \frac{b^2\left(Y_u\right)}{\sigma^2} du\right)$$
$$= \exp\left(\frac{1}{\sigma^2}\left(G\left(Y_\tau\right) - G\left(x\right)\right) + \sigma^2 \int_0^\tau g\left(Y_u\right) du\right)$$
$$= \exp\left(\frac{1}{\sigma^2}\left(G\left(Y_\tau\right) - G\left(x\right)\right)\right) \times \exp\left(\sigma^2 \int_0^\tau g\left(Y_u\right) du\right)$$

Thus,

$$\Lambda_\tau(x,y) = E_{\mu_W^x}\left[\exp\left(\frac{1}{\sigma^2}\left(G(Y_\tau) - G(x)\right)\right) \times \exp\left(\sigma^2 \int_0^\tau g(Y_u)\,du\right) | Y_\tau = y\right]$$
$$= \exp\left(\frac{1}{\sigma^2}\left(G(y) - G(x)\right)\right) \times E_{\mu_W^x}\left[\exp\left(\sigma^2 \int_0^\tau g(Y_u)\,du\right) | Y_\tau = y\right].$$

In order to simplify the integral in the expectation term we consider a series of transformations beginning with a simple change of variable, let $v = u/\tau$, then $\int_0^\tau g(Y_u)\,du = \tau \int_0^1 g(Y_{\tau v})\,dv$. Now, let $Y_v^* = \frac{Y_{\tau v} - x}{\sqrt{\sigma^2 \tau}}$ and define $B_v = Y_v^* - vY_1^*$. Note that $Y_0^* = 0$ and $Y_1^* = \frac{y-x}{\sqrt{\sigma^2\tau}}$. Substituting this into the integrand gives

$$\int_0^1 g(Y_{\tau v})\,dv = \int_0^1 g\left(z_v(x,y) + \sqrt{\tau\sigma^2} B_v\right) dv.$$

Hence,

$$E_{\mu_W^x}\left[\exp\left[\sigma^2 \int_0^\tau g(Y_u)\,du\right] | Y_\tau = y\right]$$
$$= E_{\mu_W^x}\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_v(x,y) + \sqrt{\tau\sigma^2} B_v\right) dv\right) | Y_\tau = y\right].$$

Note that under the measure μ_W^x Y is a Wiener process starting at $Y_0 = x$, therefore under μ_W^x the process B is a standard Brownian bridge which is independent of Y_1^* and Y_τ. It then follows that

$$E_{\mu_W^x}\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_v(x,y) + \sqrt{\tau\sigma^2} B_v\right) dv\right) | Y_\tau = y\right]$$
$$= E\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_v(x,y) + \sqrt{\tau\sigma^2} B_v\right) dv\right)\right]$$

where B is a standard Brownian bridge. Substituting this into Λ gives

$$\Lambda_\tau(x,y) = \exp\left(\frac{1}{\sigma^2}\left(G(y) - G(x)\right)\right)$$
$$\times E\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_v(x,y) + \sqrt{\tau\sigma^2} B_v\right) dv\right)\right],$$

and so

$$p_\tau(x,y) = \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left(-\frac{(x-y)^2}{2\sigma^2\tau} + \frac{G(y) - G(x)}{\sigma^2}\right) \times E\left[\exp\left(\sigma^2\tau \int_0^1 g\left(z_u(x,y) + \sqrt{\sigma^2\tau} B_u\right) du\right)\right],$$

completing the proof. QED.

Lemma 2. Consider a one dimensional diffusion model of the form

$$dZ_t = b_\theta(Z_t)\, dt + a_\sigma(Z_t)\, dW_t$$

where b_θ and W are defined as above, $Z_0 = x \in \Re$ and $a_\sigma : \Re \to (0,\infty)$ is a positive diffusion coefficient, let

$$s(x) = \int_0^x \frac{du}{a(u)},$$

and denote by v inverse function of s. Let

$$C(x) = \frac{b(v(x))}{a(v(x))} - \frac{1}{2} a'(v(x)), \quad a, b \in C^1$$

$$H(x) = \int_0^{s(x)} C(u)\, du, \quad g = -\frac{1}{2}\left(C^2 + C'\right)$$

Suppose $|g_\alpha(x)| = O\left(|x|^2\right)$ for $|x| \to \infty$. Then:

$$p_\tau(x,y) = \frac{1}{\sqrt{2\pi\tau} a_\sigma(y)} \exp\left(-\frac{(s(x) - s(y))^2}{2\tau} + H(y) - H(x)\right) \times E\left[\exp\left(\tau \int_0^1 g_\alpha\left(z_u(s(x), s(y)) + \sqrt{\tau} B_u\right) du\right)\right]$$

Proof: For simplicity we drop the subscripts on the drift and diffusion functions. Lemma 2 results from Lemma 1, let $U_t = s(Z_t)$ and denote by v the inverse

function s. Note that $s'(x) = a(x)^{-1}$ and $s''(x) = -a'(x)\,a(x)^{-2}$. Then by Ito's lemma we have that

$$\begin{aligned}
dU_t &= s'(Z_t)\,dZ_t + \frac{1}{2}s''(Z_t)\,a^2(Z_t)\,dt \\
&= s'(v(U_t))\,(b(v(U_t))\,dt + a(v(U_t))\,dW_t) + \frac{1}{2}s''(v(U_t))\,a^2(v(U_t))\,dt \\
&= \frac{b(v(U_t))}{a(v(U_t))}dt + dW_t - \frac{1}{2}\frac{a'(v(U_t))}{a^2(v(U_t))}a^2(v(U_t))\,dt \\
&= \left(\frac{b(v(U_t))}{a(v(U_t))} - \frac{1}{2}a'(v(U_t))\right)dt + dW_t. \\
&= C(U_t)\,dt + dW_t
\end{aligned}$$

By lemma 1, for $\tau > 0$ the density of $U_\tau | U_0 = u_0$ is given by

$$\begin{aligned}
p_\tau^U(u_0, u) = \frac{1}{\sqrt{2\pi\tau}}\exp\left(-\frac{(u_0-u)^2}{2\tau} + G(u) - G(u_0)\right) \\
\times E\left[\exp\left(\tau\int_0^1 g\left(z_x(u_0,u) + \sqrt{\tau}B_x\right)dx\right)\right],
\end{aligned}$$

where

$$G(x) = \int_0^x C(u)\,du,$$

$$g(x) = -\frac{1}{2}\left(C(x)^2 + C(x)'\right)$$

and B is a standard Brownian bridge. We wish to obtain the density of $Z_\tau | Z_0 = z_o$, $p_\tau^Z(z_0, z)$. Z_τ is simply a transformation of U_τ, and so by basic results concerning

probability density functions we have that

$$\begin{aligned} p_\tau^Z(z_0, z) &= p_\tau^U(s(z_0), s(z)) |s'(z)| \\ &= \frac{1}{\sqrt{2\pi\tau}} \exp\left(-\frac{(s(z_0) - s(z))^2}{2\tau} + G(s(z)) - G(s(z_0)) \right) \\ &\times E\left[\exp\left(\tau \int_0^1 g\left(z_x(s(z_0), s(z)) + \sqrt{\tau} B_x \right) dx \right) \right] \times \frac{1}{a(z)} \\ &= \frac{1}{\sqrt{2\pi\tau} a(z)} \exp\left(-\frac{(s(z_0) - s(z))^2}{2\tau} + H(z) - H(z_0) \right) \\ &\times E\left[\exp\left(\tau \int_0^1 g\left(z_x(s(z_0), s(z)) + \sqrt{\tau} B_x \right) dx \right) \right] .QED. \end{aligned}$$

6.3 Auxiliary Simulation Results

#1. This result is used for conditional simulation of the process given in (3.9) for which the function c is not dependent on the current state of the process. Consider two independent random variables X and Y. Denote their sum as $Z = X + Y$. Consider the problem of simulating $(X, Y)|Z$. Suppose the p.d.f. of X satisfies $f_X \leq c$. Use accept/reject sampling:

1. Generate $Y \sim f_Y$. Put $X = Z - Y$.

2. Accept X with probability

$$\frac{f_{X|Z}(X|Z)}{f_Y(Z - X)} = \frac{f_X(X) f_Y(Z - X)}{f_Z(Z) f_Y(Z - X)} = \frac{f_X(X)}{f_Z(Z)} \propto \frac{f_X(X)}{c}.$$

#2. This result is used for the conditional simulation of the process given in (3.9). In the following, each X_i can be considered as the random variable determining the jump or the an increment in the process. Let $X_1, ..., X_n$ be independent

random variables with probability density functions

$$f_i(x),\ i = 1, 2, ..., n.$$

We wish to generate a sample conditional on $T = t$ where

$$T = g\left(X_1, ..., X_{n-1}\right) + X_n.$$

Note that the joint probability density function of $(X_1, ..., X_{n-1}, T)$ is given by

$$f_1(x_1)...f_{n-1}(x_{n-1})f_n(t - g(x_1, ..., x_{n-1}))$$

since the Jacobian of the transformation $(X_1, ..., X_n) \rightarrow (X_1, ..., X_{n-1}, T)$ is one. This implies that the conditional distribution of $(X_1, ..., X_{n-1})$ given $T = t$ is proportional to

$$f_1(x_1)...f_{n-1}(x_{n-1})f_n(t - g(x_1, ..., x_{n-1}))$$

and we can generate from this conditional distribution by generating independently (i.e. from the joint p.d.f. $f_1(x_1)...f_{n-1}(x_{n-1})$) and then accepting this vector with probability proportional to $f_n(t - g(x_1, ..., x_{n-1}))$.