

## Fast valuation of financial derivatives

J. G. M. Schoenmakers and A. W. Heemink

A method for pricing European contingent claims (options) based on Monte Carlo simulation with variance reduction is presented. The evolution of the option price can be formulated as a Kolmogorov final value problem and thus be calculated numerically either by solving the deterministic partial differential equation or by simulating a large number of trajectories of the corresponding stochastic differential equation. The authors discuss a Monte Carlo simulation method combined with variance reduction obtained from a Girsanov transformation of the stochastic differential equation by a correction term that is obtained as a rough solution of the partial differential equation computed by a classical numerical method. The trade-off between these methods is investigated and it is shown that the composite method is more efficient than either the standard Monte Carlo or the classical numerical method.

### 1. INTRODUCTION

The subject of this paper is a general and efficient pricing method for nonstandard financial derivatives, e.g. exotic options, interest rate derivatives, etc. The need for such a method arises from the situation in the financial markets, which have been strongly increasing in competition and complexity over the last decade.

From stochastic finance theory, the evaluation of a wide class of derivatives comes down to the computation of the expectation functional

$$Ef(X_T^{t,x}). \quad (1.1)$$

The stochastic process  $X$  describes the price evolution of one or more underlying financial quantities such as stock prices, exchange indices, etc., under a risk-neutral probability measure. The process  $X$  is supposed to have the initial value  $x$  at time  $t$ . The function  $f$  specifies the value of the derivative at the exercise time  $T$ . Depending on the style of the derivative,  $T$  can be a fixed time or a stopping time. With the exception of some special cases, analytic expressions for (1.1) cannot be given, so we have to search for numerical solutions.

Our method for the computation of (1.1) will be based on Monte Carlo simulation applied to numerical discretizations of stochastic differential equations. The Monte Carlo method is ideal for parallel processing. Moreover one can tackle a great diversity of pricing problems with one and the same implementation of this method. However, in view of the rather high variance of the random variable  $f(X_T)$  in (1.1) in practice, straightforward Monte Carlo simulation has the disadvantage of slow convergence. To solve this problem, we transform the process  $X$  into a process  $\tilde{X}$  by a Girsanov transformation such that

$$Ef(X_T^{t,x}) = Ef(\tilde{X}_T^{t,x})\Theta_T, \quad (1.2)$$

where  $\Theta$  is some correction process. Now the variance of the random variable  $f(\tilde{X}_T)\Theta_T$  on

the right-hand side of (1.2) can be reduced considerably if we have a rough estimate for (1.1). There are various ways to get this estimate.

We will solve the backward Kolmogorov equation for (1.1) in  $(t, x)$  by standard numerical methods using a rough space-time grid. For instance, we use finite-difference schemes like the ADI method or the Crank-Nicolson method. We shall also deal with the application of finite-element methods like the Galerkin and the orthogonal collocation methods. These last methods have the advantage of presenting the approximate solution of (1.1) in spline form. As a result the approximation is available at every point  $(t, x)$  directly, i.e. an extra interpolation procedure between the grid points as in the case of finite-difference methods is not required. As an alternative one can implement multinomial tree models based on the Markov property for  $X$ , or, if known, one can use analytical approximations of (1.1).

The method described above is implemented and tested on a variety of nonstandard traded derivatives. Several interesting phenomena can be observed and will be discussed. For instance, if one has an approximation of (1.2) with an error  $\epsilon$  in some suitable (Sobolev) sense, then under some weak restrictions the variance of  $f(\tilde{X}_T)\Theta_T$  on the r.h.s. of (1.2) can be reduced to a factor  $O(\epsilon^2)$  times the original variance of  $f(X_T)$  in (1.1).

In the Monte Carlo simulation of the tracks of the underlying stochastic processes, we used weak schemes of different order. As expected, it turns out that the performance of the variance reduction method is better when higher-order schemes are used.

## 2. EVALUATION OF DERIVATIVES: SOME GENERAL RESULTS

Here, we consider European-style derivative securities. A European-style derivative security, also called a European option, is a contingent claim with exercise date  $T$  and payoff function  $f(X_T)$ , where  $X_t := (X_t^{(1)}, \dots, X_t^{(d)})$  represents a set of underlying financial quantities such as stock prices interest rates, etc. A derivative which may be exercised at a time prior to  $T$  is called American-style security, or an American option.

As an example we consider the  $X^{(i)}$  to represent the prices of a set of stocks. We assume that the stock prices follow the set of stochastic differential equations given by

$$dX_t^{(i)} = \mu_i X_t^{(i)} dt + \sum_{r=1}^d \sigma_{ir} X_t^{(i)} dW_t^{(r)}, \quad i = 1, \dots, d, \quad X_{t_0} = x_0, \quad t_0 \leq t \leq T. \quad (2.1)$$

In this model the driving process  $W$  is the  $d$ -dimensional standard Brownian motion, the quantities  $\sigma_{ir}$  determine the correlation structure of the stock prices and the  $\mu_i$  are the expected return rates of the stocks. For simplicity we assume that the stocks pay no dividend. Then, along the lines of standard Black-Scholes theory, one can show that the price of a European option specified by the payoff function  $f$  at a time point  $t < T$  is a function  $P$  of  $t$  and  $x$ , where  $x$  is the state of the art of the stock prices at time  $t$  and  $P(t, x)$  satisfies a parabolic differential equation. Using the so-called 'risk-neutral valuation argument', one can then show that the solution of this differential equation is represented by

$$P(t, x) = e^{-r(T-t)} E f(\hat{X}_T^{t,x}). \quad (2.2)$$

In this expression,  $r$  is the risk-free interest rate and  $\hat{X}$  is the price process of the same stocks in a risk-neutral world, by which we mean that  $\hat{X}$  satisfies the stochastic differential equation (2.1) with every  $\mu_i$  replaced by  $r$ . Details about the Black-Scholes theory and risk-

neutral valuation can be found in Black and Scholes (1973) and Cox and Ross (1976), respectively. The valuation of options for which the price can be represented as in (2.2) is the central theme of our research. We will give some examples.

**Example 2.1. Standard put and call options**

In the case of a single stock  $S$  and a payoff function given by  $f(S_T) = \max(K - S_T, 0)$  or  $f(S_T) = \min(S_T - K, 0)$ , we speak of a (standard) European put or call option, respectively.  $K$  is called the strike price and is specified in the option contract. European call and put options can be evaluated analytically by the well-known Black-Scholes formulas.

**Example 2.2. Asian put and call options**

Suppose the payoff of an option is specified by

$$\max\left(K - \frac{1}{T} \int_0^T S_t dt, 0\right) \quad \text{or} \quad \max\left(\frac{1}{T} \int_0^T S_t dt - K, 0\right).$$

Then we speak of an Asian put or call option, respectively, with strike price  $K$ . Let us consider the two-dimensional process  $(\hat{S}_t, \hat{A}_t)$  which is governed by the equations

$$\begin{aligned} d\hat{S}_t &= r\hat{S}_t dt + \sigma\hat{S}_t dW_t, \\ d\hat{A}_t &= \frac{1}{T} \hat{S}_t dt, \quad \hat{A}_0 = 0, \end{aligned}$$

where  $\sigma$  is the volatility of the stock price  $S$ . Then, again by Black-Scholes and risk-neutral valuation arguments, it can be shown that the price of the Asian option can be represented as in (2.2), namely

$$\begin{aligned} P(t, s, a) &= e^{-r(T-t)} \mathbb{E} f(\hat{S}_T^{t,s}, \hat{A}_T^{t,a}) \\ &= e^{-r(T-t)} \mathbb{E} \max(K - \hat{A}_T^{t,a}, 0), \end{aligned}$$

where  $f(x, y) := \max(K - y, 0)$  for an Asian put, or  $f(x, y) := \max(y - K, 0)$  for an Asian call, option. Although for this Asian option analytical representations for the option prices have been recently found, their computation is highly intractable, so we should use numerical approximation methods, Monte Carlo methods, or known approximation formulas, or we use the method as developed and described in this paper.

**Example 2.3. Generalized Asian options**

As a test problem for our new valuation method we consider a generalization of the standard Asian option. We consider two stocks  $S^{(1)}$  and  $S^{(2)}$  and specify the option contract by the payoff value

$$\max\left(K - \frac{1}{T} \int_0^T \min(S_t^{(1)}, S_t^{(2)}) dt, 0\right).$$

We suppose that the two stock prices follow the system of equations given in (2.1) for  $d = 2$ . As before, by introducing the risk-neutral stock prices  $\hat{S}^{(1)}$  and  $\hat{S}^{(2)}$  and a new process  $G_t$  which satisfies

$$dG_t = \frac{1}{T} \min(\hat{S}_t^{(1)}, \hat{S}_t^{(2)}) dt, \quad G_0 = 0,$$

we can show that the option price of this generalized Asian option is represented by

$$\begin{aligned} P(t, s_1, s_2, g) &= e^{-r(T-t)} E f(\hat{S}_T^{1;t,s_1}, \hat{S}_T^{2;t,s_2}, G_T^{t,g}) \\ &= e^{-r(T-t)} E \max(K - G_T^{t,g}, 0), \end{aligned}$$

with  $f(x, y, z) := \max(K - z, 0)$ . Note that for  $s_2 \uparrow \infty$  this generalized Asian put collapses to a standard Asian put and we have  $P(t, s_1, \infty, g) = A(t, s_1, g)$ , with  $\sigma = \sqrt{\sigma_{11}^2 + \sigma_{12}^2}$ . For  $s_1 \uparrow \infty$ , we have a similar result.

The derivative valuation method developed in this paper is based on the Monte Carlo method combined with a variance reduction technique. This method is tested on the examples above and some results are reported in Section 6. In the next section we study the accuracy of the Monte Carlo method in more detail.

### 3. THE MONTE CARLO METHOD IN PRACTICE

Suppose for some real-valued random variable  $X$  we want to estimate

$$E X \quad (3.1)$$

by Monte Carlo simulation. So, if  $X_1, X_2, \dots$  is an independent sequence of realizations of  $X$ , we consider the unbiased estimator

$$\hat{s}_N = \frac{1}{N} \sum_{i=1}^N X_i \quad (3.2)$$

for (3.1). However, in many practical situations, especially in those which occur here, the distribution of  $X$  is not exactly known, and so it is not possible to sample directly from a distribution equal to the distribution of  $X$ . Instead we make the following assumption.

**Assumption 3.1.** For each  $\Delta > 0$  independent realizations of a random variable  $X_\Delta$  can be generated, where  $X_\Delta$  is an approximation in the distribution of  $X$  of order  $O(\Delta^p)$  ( $p > 0$ ) as  $\Delta \downarrow 0$ .

Now we consider the generally biased estimator

$$\hat{s}_N^\Delta = \frac{1}{N} \sum_{i=1}^N X_i^\Delta \quad (3.3)$$

and we observe that

$$\begin{aligned} E X - \hat{s}_N^\Delta &= E X - E X_\Delta + E X_\Delta - \hat{s}_N^\Delta \\ &=: \epsilon_{\text{syst}} + \epsilon_{\text{stat}}. \end{aligned}$$

So we have a deterministic error  $\epsilon_{\text{syst}} = O(\Delta^p)$ , independent of  $N$ , and a statistical error  $\epsilon_{\text{stat}}$ , for which we have  $E \epsilon_{\text{stat}} = 0$  and  $\text{Var } \epsilon_{\text{stat}} = N^{-1} \text{Var } X_\Delta$ . We conclude that if one has to attain a certain accuracy in the evaluation of (3.1) by means of Monte Carlo simulation then one has to keep both the systematic error and the statistical error small.

It is clear that for a small systematic error we have to take  $\Delta$  small, but generally the computational effort needed for the generation of an individual realization of  $X_\Delta$  will increase as  $\Delta$  tends to zero. Once a small  $\Delta > 0$  is chosen, then we have to choose a (large) number  $N$  for the sample size to obtain a small statistical error. We see that an optimization

problem arises: is it possible to choose  $\Delta$  and  $N$  in such a way that the effort needed to reach a certain accuracy in some sense is minimal?

We will solve this optimization problem in a statistical model that is specified by the following definitions and assumptions.

Definition 3.2. Let:

$$\sigma := \sqrt{\text{Var } X};$$

$$\sigma_\Delta := \sqrt{\text{Var } X_\Delta};$$

$C(\Delta)$ : the computational time needed for one sample of  $X_\Delta$ ;

$C_{MM}(\Delta, N) := NC(\Delta)$ : the total computational time needed for the Monte Carlo method.

The total error of the Monte Carlo method will be *defined* as:

$$\varepsilon_{MM} := \varepsilon_{\text{syst}} + \text{deviation of } \varepsilon_{\text{stat}}.$$

Assumption 3.3. For some  $q > 0$  and  $\Delta \downarrow 0$ , we assume that

$$C(\Delta) = (\beta/\Delta^q)[1 + o(1)], \quad \sigma_\Delta = \sigma[1 + o(1)], \quad \varepsilon_{\text{syst}} = \alpha\Delta^p[1 + o(1)].$$

From these assumptions it follows that, for  $\Delta \downarrow 0$ ,

$$C_{MM}(\Delta, N) = \frac{\beta N}{\Delta^q}[1 + o(1)], \quad (3.4)$$

$$\varepsilon_{MM}(\Delta, N) = \left( \alpha\Delta^p + \frac{\sigma}{\sqrt{N}} \right) [1 + o(1)]. \quad (3.5)$$

Now we can deal with the following two optimization problems:

$$C^*(\epsilon) := \min_{\{(\Delta, N): \varepsilon_{MM}(\Delta, N) \leq \epsilon\}} C_{MM}(\Delta, N), \quad (3.6)$$

$$\varepsilon^*(c) := \min_{\{(\Delta, N): C_{MM}(\Delta, N) \leq c\}} \varepsilon_{MM}(\Delta, N). \quad (3.7)$$

Under the previous assumptions, the following lemma can be verified by elementary asymptotic calculus.

Lemma 3.4. There exist a right-hand neighborhood  $U_0$  of 0 and a neighborhood  $U_\infty$  of  $+\infty$  such that

- (i)  $\epsilon \in U_0 \implies C^*(\epsilon) \in U_\infty$  and  $c \in U_\infty \implies \varepsilon^*(c) \in U_0$ .
- (ii) If  $\epsilon \in U_0$  in problem (3.6) or  $c \in U_\infty$  in problem (3.7), there exist unique minimizing pairs  $(\Delta_1^*, N_1^*) \in U_0 \times U_\infty$  and  $(\Delta_2^*, N_2^*) \in U_0 \times U_\infty$ , respectively.
- (iii) The functions  $C^*$  and  $\varepsilon^*$  are each other's inverse on the sets  $U_0$  and  $U_\infty$ .
- (iv)  $\epsilon \downarrow 0 \iff \Delta^* \downarrow 0 \wedge N^* \rightarrow \infty \iff C^*(\epsilon) \rightarrow \infty$ . Moreover,

$$\Delta^*(\epsilon) = \epsilon^{1/p} \lambda_{p,q,\alpha} [1 + o(1)], \quad (3.8)$$

$$N^*(\epsilon) = \frac{\sigma^2}{\epsilon^2} (1 + q/2p)^2 [1 + o(1)], \quad (3.9)$$

$$C^*(\epsilon) = \frac{\beta\sigma^2}{\epsilon^{2+q/p}} \mu_{p,q,\alpha} [1 + o(1)], \quad (3.10)$$

for  $\epsilon \downarrow 0$ , where

$$\lambda_{p,q,\alpha} := \left( \frac{q/2p}{\alpha(1 + q/2p)} \right)^{1/p} \quad \text{and} \quad \mu_{p,q,\alpha} := \alpha^{q/p} \frac{(1 + q/2p)^{2+q/p}}{(q/2p)^{q/p}}. \quad (3.11)$$

Remark 3.5. For practical applications Lemma 3.4 is best interpreted in the following way. Let  $D_0$  and  $N_0$  be some positive constants. Then, by choosing

$$\Delta(\epsilon) = D_0 \epsilon^{1/p} [1 + o(1)] \quad \text{and} \quad N(\epsilon) = \frac{N_0}{\epsilon^2} [1 + o(1)],$$

the computation time becomes

$$C(\epsilon) = O\left(\frac{1}{\epsilon^{2+q/p}}\right) \quad \text{as } \epsilon \downarrow 0,$$

and this is the most efficient order which can be achieved.

In Duffie and Glynn (1995) several error coefficients are studied in detail for Monte Carlo simulation of security prices.

#### 4. VARIANCE REDUCTION BY GIRSANOV'S TRANSFORMATION

In this section we introduce the variance reduction method, which we use for the Monte Carlo valuation of derivatives. The method is based on a Girsanov transformation of the original stochastic process  $X$ . To explain this transformation we need to use Itô's lemma from stochastic calculus. We consider the Itô stochastic differential equation for the  $d$ -dimensional process  $X$ ,

$$dX_t = a(t, X_t) dt + \sigma(t, X_t) dW_t, \quad X_{t_0} = x_0, \quad t_0 \leq t \leq T, \quad (4.1)$$

where  $W_t$  is a standard  $m$ -dimensional Wiener process,  $x_0$  is an  $\mathbb{R}^d$ -valued random variable independent of  $W_t - W_{t_0}$  ( $t \geq t_0$ ). The vector-valued function  $a : [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$  and the matrix-valued function  $\sigma : [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}^{d \times m}$  are supposed to satisfy sufficient conditions for the existence and uniqueness for the solution of (4.1) (see Arnold 1974). The solution of the system (4.1) will be denoted by  $X_{t_0, x_0}$ .

Lemma 4.1 (Itô's formula). Suppose  $v : [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$  has continuous partial derivatives up to second order. If  $X$  is the solution of the system (4.1), then the real-valued process  $Y_t := v(t, X_t)$  satisfies the system

$$dY_t = \mathcal{L}v(t, X_t) dt + \sum_{i=1}^d \sum_{r=1}^m v_{x_i}(t, X_t) \sigma_{ir}(t, X_t) dW_t^r, \quad (4.2)$$

with initial condition  $Y_{t_0} = v(t_0, x_0)$ , where the operator  $\mathcal{L}$  is defined by

$$\mathcal{L}v := v_t + \sum_{i=1}^d a_i v_{x_i} + \frac{1}{2} \sum_{i=1}^d \sum_{j=1}^d \sum_{r=1}^m \sigma_{ir} \sigma_{jr} v_{x_i x_j}.$$

From Itô's lemma we can derive immediately a representation for the solution of the backward Kolmogorov problem.

Corollary 4.2 (Backward Kolmogorov problem). Suppose  $u : [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$  belongs to  $C_2([0, T] \times \mathbb{R}^d)$  and satisfies

$$\mathcal{L}u \equiv 0 \quad \text{in } [0, T] \times \mathbb{R}^d, \quad u(T, x) = f(x), \quad x \in \mathbb{R}^d, \quad (4.3)$$

where the function  $f: \mathbb{R}^d \rightarrow \mathbb{R}$  belongs to  $C_2(\mathbb{R}^d)$ . Then, from Itô's lemma, it follows that the process  $Y_t := u(t, X_t)$  is a martingale and consequently

$$\mathbb{E} u(t_0, x_0) = \mathbb{E} Y_{t_0} = \mathbb{E} Y_T = \mathbb{E} u(T, X_T^{t_0, x_0}) = \mathbb{E} f(X_T^{t_0, x_0}).$$

Taking deterministic initial conditions, it follows that

$$u(t, x) = \mathbb{E} f(X_T^{t, x}), \quad t \in [0, T], \quad x \in \mathbb{R}^d. \quad (4.4)$$

It can be shown that, with some smoothness and growth conditions on  $f$  also, the converse is true (see Kloeden and Platen 1992).

As we have seen in Section 2, many practical derivative problems can be reformulated as a backward Kolmogorov problem (4.3). Although direct Monte Carlo simulation of (4.1) looks attractive, in many cases the variance of the random variable  $f(X_T^{t, x})$  turns out to be so high that a very large number of simulations are needed for a satisfactory estimation. We will reduce this variance by a method known as the Girsanov transformation method, by which  $f(X_T^{t, x})$  is transformed into a new random variable with the same expectation but a substantially smaller variance.

Let  $X^{t_0, x_0}$  be the solution of the system (4.1) and let  $u: [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$  be a solution of the backward Kolmogorov problem (4.3) belonging to (4.1). In system (4.1) we introduce an extra drift term by replacing the driving Wiener processes  $W_t$  by  $-v + W_t$ , where the vector function  $v: [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}^m$  is supposed to satisfy similar conditions to  $a$  and  $\sigma$ . Next, let  $\tilde{X}^{t_0, x_0}$  be the solution of the system

$$d\tilde{X}_t = [a(t, \tilde{X}_t) - \sigma(t, \tilde{X}_t)v(t, \tilde{X}_t)]dt + \sigma(t, \tilde{X}_t)dW_t, \quad \tilde{X}_{t_0} = x_0, \quad t_0 \leq t \leq T, \quad (4.5)$$

and let the scalar process  $\Theta_t$  be the solution of the one-dimensional system

$$d\Theta_t = \Theta_t \sum_{r=1}^m v_r(t, \tilde{X}_t) dW_t^r, \quad \Theta_{t_0} = \theta_0. \quad (4.6)$$

Then, by Lemma 4.1, it follows that

$$d[u(t, \tilde{X}_t)\Theta_t] = \Theta_t \sum_{r=1}^m \left( u(t, \tilde{X}_t)v_r(t, \tilde{X}_t) + \sum_{i=1}^d u_{x_i}(t, \tilde{X}_t)\sigma_{ir}(t, \tilde{X}_t) \right) dW_t^r, \quad (4.7)$$

with  $u(t_0, \tilde{X}_{t_0})\Theta_{t_0} = u(t_0, x_0)\theta_0$ .

Hence, if  $x_0 =: x$  and  $\theta_0 =: \theta \neq 0$  are deterministic, it follows by the martingale property of (4.7) that

$$\begin{aligned} u(t_0, x) &= \mathbb{E} f(X_T^{t_0, x}) \\ &= \mathbb{E} u(T, \tilde{X}_T^{t_0, x}) \Theta_T^{t_0, \theta} / \theta. \end{aligned} \quad (4.8)$$

It is important to note that (4.8) holds regardless the choice of  $v$ . Now, what we hope for is that the variance of the right-hand side of (4.8) is much smaller than the variance of  $f(X_T^{t_0, x})$ . It can immediately be seen that this variance is even reduced to zero if  $v$  satisfies

$$uv_r = - \sum_{i=1}^d u_{x_i} \sigma_{ir}, \quad r = 1, \dots, m, \quad (4.9)$$

in which case

$$u(t, \tilde{X}_t)\Theta_t = u(t_0, x)\theta \quad \text{a.s. for every } 0 \leq t \leq T. \quad (4.10)$$

Unfortunately this choice of  $v$ , if possible, requires the complete knowledge of the solution of the original backward Kolmogorov problem (4.3). However, we can imagine that we have an approximate solution of (4.3) obtained from other sources, in which case we can use the following theorem.

**Theorem 4.3 (Variance reduction)**

- (i) We assume that there is a constant  $c > 0$  such that  $f \geq c$ .
- (ii) Let  $\hat{u}$  be an approximation of  $u$  in the following sense: if the real function  $\eta$  and the vector function  $\rho$  are defined by

$$\hat{u} = u + \eta u \quad \text{and} \quad \nabla \hat{u} = \nabla u + \rho \|\nabla u\|,$$

where  $\|\cdot\|$  denotes the Euclidean norm and  $\nabla$  denotes the spatial gradient operator, then we assume that, for  $0 < \eta_0 \leq \frac{1}{2}$  and  $\rho_0 > 0$ , we have  $|\eta| \leq \eta_0$  and  $\|\rho\| \leq \rho_0$  uniformly on  $[0, T] \times \mathbb{R}^d$ .

- (iii) For the controls

$$\hat{v} := -\frac{1}{\hat{u}} \sigma^T \nabla \hat{u} \quad \text{and} \quad v := -\frac{1}{u} \sigma^T \nabla u,$$

let  $\tilde{X}_t^{\hat{v}}, \Theta_T^{\hat{v}}$  and  $\tilde{X}_t^v, \Theta_T^v$  be the solution of systems (4.5) and (4.6), respectively.

Then, from (i), (ii), and (iii), it follows that

$$\text{Var } u(T, \tilde{X}_T^{\hat{v}^{0,x}}) \Theta_T^{\hat{v}^{0,x}} / \theta \leq (\eta_0 + \rho_0)^2 K [1 + o(1)], \quad \eta, \rho \downarrow 0,$$

where

$$K := 4\theta^{-2} \mathbb{E} \int_0^T (\Theta_t^{\hat{v}})^2 \|\sigma(t, \tilde{X}_t^{\hat{v}})\|^2 \|\nabla u(t, \tilde{X}_t^{\hat{v}})\|^2 dt < \infty.$$

The constant  $K$  can also be expressed as

$$K = 4u^2(t_0, x) \mathbb{E} \int_{t_0}^T \|\sigma(t, \tilde{X}_t^{\hat{v}})\|^2 \|\nabla \log u(t, \tilde{X}_t^{\hat{v}})\|^2 dt.$$

The proof of this theorem follows from Itô calculus, (4.1), and the fact that  $\tilde{X}^{\hat{v}} \rightarrow \tilde{X}^v$  and  $\Theta^{\hat{v}} \rightarrow \Theta^v$  in distribution as  $\eta_0, \rho_0 \downarrow 0$ . The last assertion follows from (4.10).

**Remark:** Assumption (i) is not really a restriction, since, if we split  $f$  into its positive and negative parts by  $f = f^+ - f^-$ , then, for every  $c > 0$ , we may write

$$f = (f^+ + c) - (f^- + c).$$

Next we apply the variance reduction method for problem (4.3) separately for  $f^+ + c$  and  $f^- + c$  and add the results.



## 5. ON THE EFFICIENCY OF THE MONTE CARLO METHOD COMBINED WITH VARIANCE REDUCTION

We will study the involved minimal computational effort for the application of the variance reduction method of the previous section to the Monte Carlo problem

$$Ef(X_T^{t_0,x}). \quad (5.1)$$

We assume that the control function  $\hat{v}$  is computed in such a way that

$$\text{Var } u(T, \tilde{X}_T^{\hat{v}^{t_0,x}}) \Theta_T^{\hat{v}^{t_0,x}} / \theta = \varrho^2 \text{Var } f(X_T^{t_0,x}), \quad (5.2)$$

where  $\varrho$  denotes the factor by which the original deviation is reduced and satisfies  $0 < \varrho < 1$ . It is plausible to assume that this variance reduction is realized at the cost of

$$\frac{R}{\varrho^l} (1 - \varrho) [1 + o(1)], \quad \varrho \downarrow 0,$$

computation units, where  $R, l > 0$  are positive constants. In this assumption we have taken into account that the required computational effort for  $\hat{v}$  vanishes as  $\varrho \uparrow 1$ . Indeed, for a factor  $\varrho = 1$ , we only have to take  $\hat{v} \equiv 0$ .

Now suppose we want to end up with an estimation of (5.1) with accuracy  $\epsilon$  in the sense of Definition 3.2. Then, once the function  $\hat{v}$  is computed to give a variance reduction factor  $\varrho^2$ , it follows from Lemma 3.4 that the Monte Carlo method requires an amount

$$\frac{\beta \varrho^2 \sigma_0^2}{\epsilon^{2+1/p}} \mu_{p,1,\alpha} [1 + o(1)] \quad \text{for } \epsilon \downarrow 0$$

of computational units. Here  $\sigma_0^2 := \text{Var } f(X_T^{t_0,x})$ . Note that we have taken  $q = 1$  in Lemma 3.4. This can be argued as follows. The  $\Delta$  in Lemma 3.4 is now represented by the time step in the stochastic discretization scheme which we are using for the simulation of

$$u(T, \tilde{X}_T^{\Delta, \hat{v}^{t_0,x}}) \Theta_T^{\Delta, \hat{v}^{t_0,x}} / \theta,$$

and the computation time needed for one track is always proportional to the number of time steps, regardless of the weak order  $p$  of the scheme.

Now we conclude that, dependent on the choice of  $\varrho$ , the combined method requires a computation time

$$C_{\text{total}}(\epsilon, \varrho) := \frac{R}{\varrho^l} (1 - \varrho) [1 + o(1)] + \frac{\beta \varrho^2 \sigma_0^2}{\epsilon^{2+1/p}} \mu_{p,1,\alpha} [1 + o(1)], \quad \varrho, \epsilon \downarrow 0. \quad (5.3)$$

By similar asymptotic techniques as used in the proof of Lemma 3.4, it can be shown that, for fixed  $\epsilon$ , the function  $C_{\text{total}}(\epsilon, \cdot)$  has a minimum for

$$\varrho^* = K^* \epsilon^{\frac{2+1/p}{2+l}} [1 + o(1)] \quad (5.4)$$

and we denote the minimum by

$$C_{\text{total}}^*(\epsilon) := L^* \epsilon^{-l \frac{2+1/p}{2+l}} [1 + o(1)], \quad \epsilon \downarrow 0. \quad (5.5)$$

The constants  $K^*$  and  $L^*$  depend on several parameters and are given by

$$K_{R,l,\beta,\sigma_0,\mu}^* := \left( \frac{Rl}{2\beta\sigma_0^2\mu_{p,1,\alpha}} \right)^{\frac{1}{2+l}}, \quad (5.6)$$

$$L_{R,l,\beta,\sigma_0,\mu}^* := R^{\frac{2}{2+l}}(\beta\sigma_0^2\mu_{p,1,\alpha})^{\frac{l}{2+l}}[(2/l)^{\frac{l}{2+l}} + (l/2)^{\frac{2}{2+l}}]. \quad (5.7)$$

The corresponding time step size  $\Delta^*(\epsilon)$  and the number of tracks  $N^*(\epsilon)$  that has to be generated can be derived by using the formulas in Lemma 3.4, where one has to take  $\sigma = \varrho^*\sigma_0$ .

Since  $0 < l/(2+l) < 1$  always, we see that the standard Monte Carlo order  $O(\epsilon^{-(2+1/p)})$  for the computation time is improved to  $O(\epsilon^{-l\frac{2+1/p}{2+l}})$ . So the combined method is always superior to standard Monte Carlo simulation without variance reduction. However, the gain is lost if the variance reductor is too expensive, i.e. as  $l \rightarrow \infty$ . In that case we observe that, for fixed  $\epsilon$ ,  $\varrho^* \uparrow 1$  consistently and  $C_{\text{total}}^*$  tends to the standard Monte Carlo computation time given in Lemma 3.4. Next, let us consider a very profitable variance reductor, i.e. we study the case  $l \downarrow 0$  while  $\epsilon$  is fixed. Indeed, it then follows that  $\varrho^* \downarrow 0$ . But, in the model around Lemma 3.4, we have  $\sigma \downarrow 0$  and we have to be careful with interpretation. For instance, from Lemma 3.4, it would now follow that  $N \downarrow 0$ , whereas  $N$  is the number of tracks which is an integer at least equal to 1. The reason for this inconsistency is that the solution of the optimization problem holds only for larger  $N$ . Anyway, we will show that there is a lower bound  $l_0$  for  $l$ , below which it is no longer advantageous to apply the variance reduction method. From Theorem 4.3, it follows that the achievement of a variance reduction factor  $\varrho^* = O(\epsilon^{\frac{2+1/p}{2+l}})$  to end up finally with an accuracy  $\epsilon$  for the resulting estimation is guaranteed if we approximate both the solution  $u$  of (4.3) and its gradient  $\nabla u$  at least to that same order. Here we assume for a moment that  $u$  and  $\|\nabla u\|$  are bounded and bounded away from 0, so the orders of relative error and the absolute error can be considered as the same. We therefore see that applying the variance reduction method makes no sense if  $(2+1/p)/(2+l) \geq 1$  or  $l \leq l_0 := 1/p$ . In this case it is better to solve the problem by using the numerical approximation method alone.

We are going to extend the previous efficiency considerations and formulate some useful results which can be used to decide whether the application of a certain numerical procedure as variance reductor is advantageous or not. In order to simplify our formulations we first introduce some notions.

**Definition 5.1.** Two functions  $f(x)$  and  $g(x)$  are said to be asymptotically proportional for  $x \downarrow 0$ , if there is a constant  $A \neq 0$  such that  $f(x) = Ag(x)[1 + o(1)]$  as  $x \downarrow 0$ . If  $f$  is asymptotically proportional to  $g$ , we will write  $f \propto g$ .

It will be convenient to fix a class of numerical procedures for the backward Kolmogorov equation by the next definition.

**Definition 5.2.** We say that a numerical procedure for the solution  $u$  of the backward Kolmogorov equation is of class  $BK(\gamma, \nu)$ , where  $\gamma, \nu > 0$ , if

- (i) for every sufficiently small  $\epsilon > 0$ , the procedure is able to compute an approximate solution  $u^{(\epsilon)}$  and its gradient  $\nabla u^{(\epsilon)}$  simultaneously such that pointwise

$$u^{(\epsilon)} - u \propto \epsilon u \quad \text{and} \quad \|\nabla u^{(\epsilon)} - \nabla u\| \propto \epsilon^\nu \|\nabla u\| \quad \text{as } \epsilon \downarrow 0,$$

where the pointwise proportional constants are supposed to be bounded and bounded away from zero on  $[0, T] \times \mathbb{R}^d$ ;

- (ii) for the computation in (i), the number of computational units required by the procedure is proportional to  $1/\epsilon^\nu$ .

If a certain computational method requires a computing time  $C(\epsilon)$  to achieve an accuracy  $\epsilon$  in the result, we will call  $C$  the complexity of the method. Now we are ready to state the next theorem which is evident from Theorem 4.3 and the expressions (5.4) and (5.5).

**Theorem 5.3 (Complexity).** Suppose a numerical procedure of class  $BK(\gamma, \nu)$  is given. Then this procedure can be used as a variance reductor for which  $l = \nu / \min(1, \gamma)$ . If, in addition,

$$\nu > \frac{1}{p} + 2 \max(0, 1 - \gamma),$$

then the computation of (5.1) up to a given accuracy  $\epsilon$  is more efficient by using Monte Carlo simulation with the procedure as variance reductor than by solving the Kolmogorov equation (4.3) up to accuracy  $\epsilon$  using the numerical procedure alone. Furthermore, the complexity of the combined method, which we will denote by  $C_{VRMC}$  where VRMC stands for 'variance reduced Monte Carlo', can be expressed as

$$C_{VRMC}(\epsilon) \propto \epsilon^{-\frac{\nu}{\nu+2\min(1,\gamma)}(2+1/p)}.$$

This complexity has to be compared with both the complexity of the pure Monte Carlo method, denoted by  $C_{MC}$ , and the complexity of the pure numerical method, denoted by  $C_{num}$ , for which we have

$$C_{MC}(\epsilon) \propto \epsilon^{-(2+1/p)} \quad \text{and} \quad C_{num}(\epsilon) \propto \epsilon^{-\nu}.$$

Then, it is immediately apparent that the gain of the combined method over the other two is maximal for those numerical BK-procedures for which

$$\nu = \nu_0 := 2 + \frac{1}{p},$$

and in that case a complexity of

$$C_{VRMC}^{(0)}(\epsilon) \propto \epsilon^{-\frac{(2+1/p)^2}{2+1/p+2\min(1,\gamma)}}$$

results.

If a method has complexity proportional to  $\epsilon^{-\kappa}$  for some  $\kappa > 0$ , we will call  $\kappa$  the exponent of complexity of the method. So, the higher  $\kappa$ , the greater the computation time required to achieve a certain accuracy  $\epsilon$ .

Theorem 5.3 and the interplay of the complexity exponent  $\kappa$  with the parameters  $\nu$  and  $\gamma$  is best displayed visually.

If the Monte Carlo tracks are generated by the weak order-one Euler scheme,  $p = 1$ , and if also the parameter  $\gamma$  is relatively unfavourable, e.g.  $\gamma = 0.5$ , the complexity parameter is shown in Figure 1. For  $\gamma \approx 1$ , the situation looks better. This is shown in Figure 2. If we use a higher weak order scheme, e.g.  $p = 2$ , for the Monte Carlo tracks, a further substantial improvement is obtained, as is seen from Figures 3 and 4.

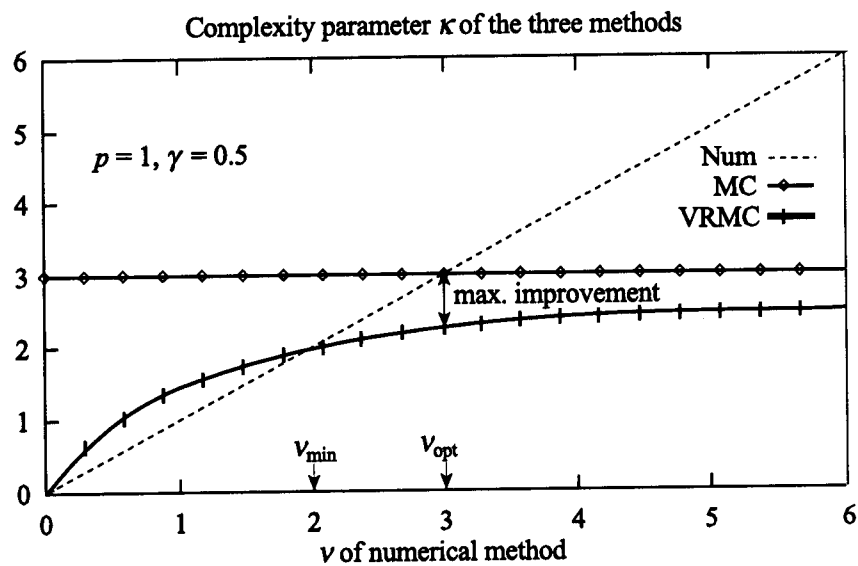


FIGURE 1

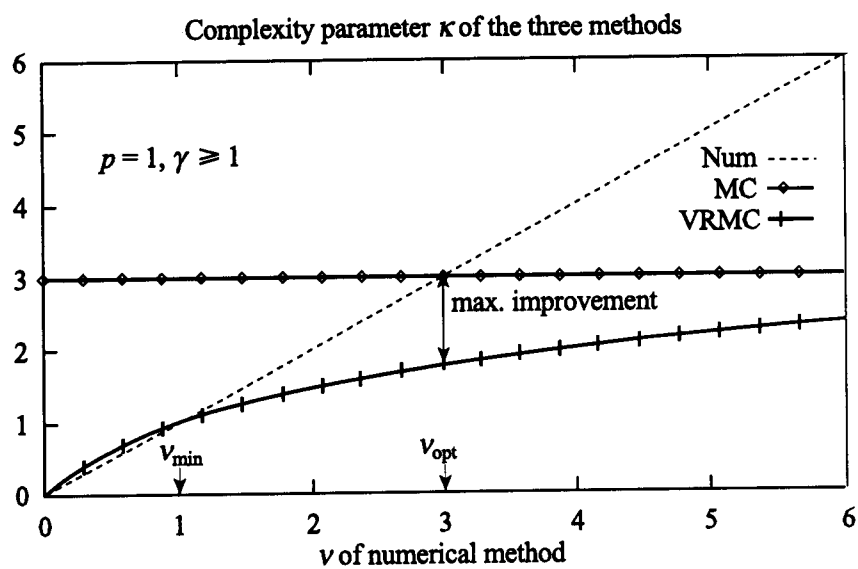


FIGURE 2

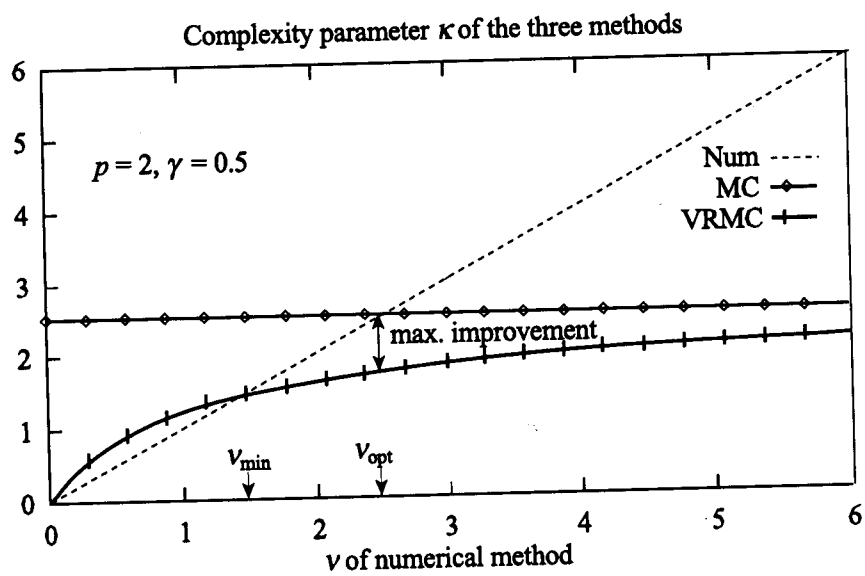


FIGURE 3

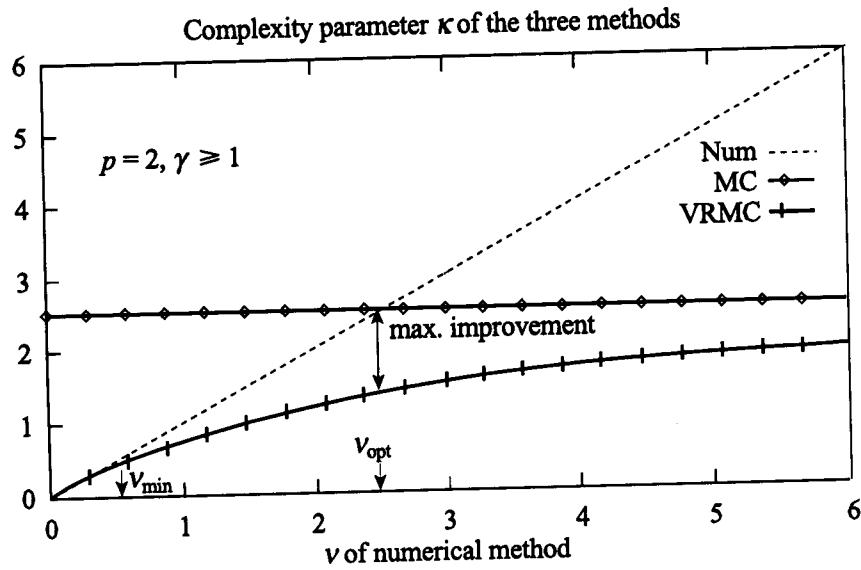


FIGURE 4

### 5.1 Conclusions and Remarks

If we need to evaluate  $u(t, x) = \mathbb{E}f(X_T^{t,x})$  in one or a few time-space points  $(t, x)$  and a numerical procedure for the accompanying Kolmogorov equation has a complexity comparable to the Monte Carlo method, it is very profitable to choose for the Monte Carlo method and use the numerical procedure as variance reductor. If the track generation scheme in the Monte Carlo method has weak order  $p$  and the numerical method is of class BK( $\nu, \gamma$ ) with  $\nu \approx 2 + 1/p$ , both methods have approximately the same complexity exponent and, by combining these methods, this exponent is reduced by a factor

$$\tau := \frac{2 + 1/p}{2 + 1/p + 2 \min(1, \gamma)}.$$

Obviously,  $\tau$  is always between  $\frac{1}{2}$  and 1. In the most ideal case, that is, when  $p$  is large,  $p \uparrow \infty$  and  $\gamma \geq 1$ , we have  $\tau \downarrow \frac{1}{2}$ . In the worst case, that is, when we have some  $p \geq 1$  but  $\gamma \downarrow 0$ , we have  $\tau \uparrow 1$ , and then there is not much gained. However, typical values for  $\tau$  are:  $\tau = 0.75$  ( $p = 1, \gamma = 0.5$ );  $\tau = 0.667$  ( $p = 1, \gamma = 0.75$ );  $\tau = 0.6$  ( $p = 1, \gamma \geq 1$ );  $\tau = 0.71$  ( $p = 2, \gamma = 0.5$ );  $\tau = 0.625$  ( $p = 1, \gamma = 0.75$ );  $\tau = 0.55$  ( $p = 2, \gamma \geq 1$ ); and so on.

## 6. IMPLEMENTATION OF SOME EXAMPLES

The valuation method developed here is implemented as a Delphi application and tested on several derivatives. For the simulation of the Monte Carlo tracks, we have used schemes of first and second order in the weak sense. The first-order scheme is the standard Euler method; we discretize the time interval  $[0, T]$  by equidistant time points  $t_n$  and step length  $\Delta$ . Depending on the noise dimension  $m$ , we sample an  $m$ -dimensional standard Gaussian random variable  $\xi$  and then take  $\Delta W_n = \sqrt{\Delta} \xi$  for the driving Brownian motion. For each component of  $\xi$ , we may also take a two- or three-point random variable with the correct moments (for details, see Kloeden and Platen 1992). Then, for equation (4.1), the Euler scheme written in matrix notation is

$$X_{n+1} = X_n + a(t_n, X_n)\Delta + \sigma(t_n, X_n)\Delta W_n.$$

For the second-order scheme, we choose a predictor-corrector method from Kloeden and Platen (1992). This scheme looks fairly complicated and much more computational work is needed to simulate one Monte Carlo track. However, besides its second-order accuracy, the predictor-corrector method has better stability properties than the Euler method. For the numerical approximation procedure, we have used a finite-difference method and a finite-element method. For the finite-difference method, we have implemented the Du Fort-Frankel scheme (see e.g. Strikwerda 1989). We choose this explicit scheme because it is computationally efficient and it is unconditionally stable. The accuracy, however, is moderate, but we don't care very much about that since the numerical procedure only has to calculate a rough estimate for the final solution. Intermediate values between the grid points are calculated by second-order interpolation methods. For the finite-element method, we have used the orthogonal collocation method (for details, see e.g. Lapidus and Pinder 1982). Collocation methods have the advantage that they are relatively easy to implement and that intermediate values can be calculated directly from the spline representation of the approximate solution. A disadvantage of this method, however, is that in general the arising

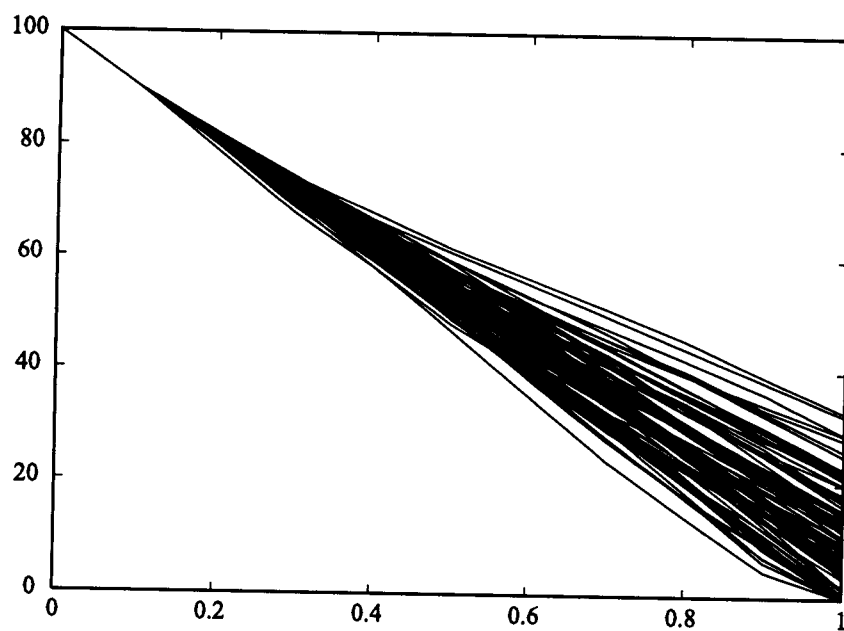


FIGURE 5

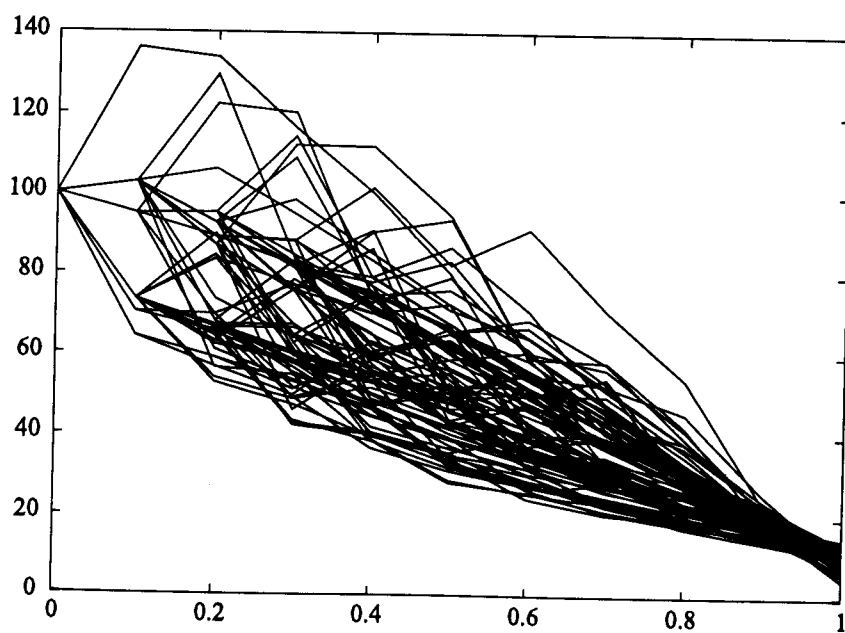


FIGURE 6

collocation matrix, which has to be inverted, has no convenient properties like diagonal dominance, and inverting this matrix by direct methods takes up a lot of computation time.

Now we will give some results from the application of our developed valuation method to the Asian option and generalized Asian option described in Section 2. The method is implemented on a DX-4 PC desktop computer.

**Example 6.1.** For the Asian put option in Section 2, we take  $r = 9\%$  annual,  $\sigma = 30\%$  annual,  $S_0 = \$100$ ,  $K = \$100$ , and  $T = 1$  year. With these data the *undiscounted* Asian put value is known to be \$5.00. After running 10 000 standard Monte Carlo tracks with 10 time steps and using the weak second-order predictor-corrector method, we find in about 4 minutes' computation time an approximation of the option value of  $4.96 \pm 0.09$ . Then we run the Du Fort-Frankel variance reductor on a  $100 \times 20 \times 20$  grid and find in about 20 seconds a rough approximation of 4.81. Next, running *variance-reduced* Monte Carlo with the same data, we find by simulating 200 tracks in 22 seconds (!) the value of  $4.99 \pm 0.07$ .

**Example 6.2.** The generalized Asian put option in Section 2 is a problem that takes much more computation time because there are two sources of randomness in the underlying processes. Besides, the solution space for the parabolic differential equation is one dimension higher. As an example, we take the following data:  $r = 9\%$  annual,  $\sigma_{11} = \sigma_{22} = 28\%$ ,  $\sigma_{12} = \sigma_{21} = -10\%$  annual,  $S_0^{(1)} = \$100$ ,  $S_0^{(2)} = \$100$ ,  $K = \$100$ , and  $T = 1$  year. After 10 000 standard Monte Carlo tracks using the predictor-corrector method with 10 time steps, we get after 8 minutes' computation time the value  $10.64 \pm 0.09$ . Next, running the Du Fort-Frankel variance reductor on a  $100 \times 11 \times 11 \times 11$  grid, we find in about 1 minute the rough approximation 11.6. Then, by running variance reduced Monte Carlo with 400 tracks we find in 2 minutes the value  $10.77 \pm 0.1$ . The performance of both methods for this example is shown in Figures 5 and 6. In Figure 5,  $f(X_t)$  is plotted against  $t$ , and, in Figure 6,  $f(\tilde{X}_t) \Theta_t$  is plotted against  $t$ . The values in Figure 6 at the terminal time  $T = 1$  are clearly much more concentrated around the expected value than in the case of Figure 5.

## REFERENCES

- Arnold, L. *Stochastic Differential Equations*. Wiley, New York, 1974.
- Black, F., and Scholes, M. The pricing of options and corporate liabilities. *Journal of Political Economy*, **81** (1973), 637-659.
- Cox, J. C., and Ross, S. A. The valuation of options for alternative stochastic processes. *Journal of Finance and Economics*, **3** (1976), 145-166.
- Duffie, D., and Glynn, P. Efficient Monte Carlo simulation of security prices. *The Annals of Applied Probability*, **5** (1995), 897-905.
- Kloeden, P.E., and Platen, E. *Numerical Solution of Stochastic Differential Equations*, Applications of Mathematics No. 23, Springer, Berlin, 1992.
- Lapidus, L., and Pinder, G. *Numerical Solution of Partial Differential Equations in Science and Engineering*. Wiley, New York, 1982.
- Strikwerda, J. C. *Finite Difference Schemes and Partial Differential Equations*. Wadsworth & Brooks, 1989.