ELSEVIER Journal of Computational and Applied Mathematics 100 (1998) 93–109

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

Step size control in the numerical solution of stochastic differential equations

Susanne Mauthner

Technische Universität Darmstadt, Fachbereich Mathematik, AG Stochastik und Operations Research, Schloßgartenstraße 7, D-64289 Darmstadt, Germany

Received 21 April 1998; received in revised form 12 July 1998

Abstract

We introduce a variable step size algorithm for the pathwise numerical approximation of solutions to stochastic ordinary differential equations. The algorithm is based on a new pair of embedded explicit Runge–Kutta methods of strong order 1.5(1.0), where the method of strong order 1.5 advances the numerical computation and the difference between approximations defined by the two methods is used for control of the local error. We show that convergence of our method is preserved though the discretization times are not stopping times any more, and further, we present numerical results which demonstrate the effectiveness of the variable step size implementation compared to a fixed step size implementation.

Keywords: Stochastic differential equations; Runge–Kutta methods; Step size control

1. Introduction

We introduce a variable step size method for the pathwise (or strong) numerical approximation of the solution to the stochastic ordinary differential equation (SODE)

$$dX_t = f(X_t)dt + g(X_t) \circ dW_t, \quad X_{t_0} = X_0, \tag{1}$$

in Stratonovich form, where f and g are real valued functions and $(W_t)_{t\geqslant 0}$ is a scalar Wiener process. The random variable X_0 denotes the random initial value at time $t = t_0$. Step size control is an important technique widely used in the numerical solution of ordinary differential equations (ODEs) (see for example [6, 11]), but existing implementations for the numerical solution of SODEs nearly always use a fixed step size (for exceptions see [9, 12]). Our method for step size control when solving a SODE (1) is based on a pair of embedded explicit stochastic Runge–Kutta methods (SRK methods) of strong order 1.5(1.0), where one of the methods is used for error control and the other advances the numerical computation.

PII: S0377-0427(98)00139-3

An outline of the paper is as follows: In Section 2 we review existing approaches to the construction of explicit SRK methods. In particular we give the order conditions derived by K. Burrage and P.M. Burrage [2] for strong order 1.0 and strong order 1.5 explicit SRK methods. On the basis of these order conditions we construct a pair of embedded SRK methods of strong order 1.5(1.0) in Section 3. The deterministic components of these methods are Runge–Kutta methods of order 4 and order 2, respectively.

Then, in Section 4, we present our variable step size method. First, in Section 4.1, we show how to simulate a trajectory of the two-dimensional normally distributed random variable

$$\left(\int_t^{t+h} \circ\, \mathrm{d}W_s, \int_t^{t+h}\int_t^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2\right),$$

which is contained in the embedded SRK method, over all time intervals of the form $[k/2^m, k+1/2^m]$ for $k,m \in \mathbb{N}, m \leqslant m_{\max}$. This special structure of the available time steps leaves us with the restriction that new step sizes are only derived from previous ones by halving or doubling. Section 4.2 describes our actual step size control mechanism. In every integration step we take the difference between the two approximations defined by the embedded SRK method as an estimate for the local error. If the estimate of the local error is smaller than a given tolerance, then we accept the step, otherwise we reject it. The estimate of the local error is used furthermore for choosing a new step size taking into account the restriction mentioned above. Section 4.3 deals with the problem of convergence of our variable step size method. In general, proofs of strong convergence of numerical methods for SODEs are based on the assumption of a fixed step size or at least on the assumption that the discretization points are all stopping times for the forward motion. This is obviously not the case for a method with step size control. However, it follows from a result of Gaines and Lyons [9] that our embedded SRK method converges to the true solution of (1), even if the discretization times are not stopping times. In the final section, Section 5, we present numerical results that demonstrate the effectiveness of the variable step size implementation compared to a fixed step size implementation.

2. Runge–Kutta methods for SODEs

An explicit s-stage Runge–Kutta method (RK method) for calculating a numerical approximation to the solution of an autonomous ODE

$$\dot{x} = f(x), \qquad x(t_0) = x_0, \tag{2}$$

is given by the recursive formula

$$\begin{aligned} y_0 &= x_0 \\ y_{n+1} &= y_n + h_n \sum_{i=1}^{s} \alpha_i f(\eta_i) \end{aligned} \tag{3}$$

with

$$\eta_i = y_n + h_n \sum_{j=1}^{i-1} a_{ij} f(\eta_j), \qquad i = 1, \ldots, s,$$

where $t_0 < t_1 < \cdots < t_N = T$ is a discretization of the integration interval $[t_0, T]$, $h_n = t_{n+1} - t_n$ and a_{ij}, $\alpha_i \in \mathbb{R}$ for $1 \leqslant i \leqslant s$, $1 \leqslant j \leqslant i-1$. To simplify notation we set $a_{ij} = 0$ for $j \geqslant i$, $A = (a_{ij})$, and $\alpha^{\mathrm{T}} = (\alpha_i)$. With the paper of Butcher [4] it became customary to symbolize method (3) by the tableau

$$\begin{array}{|llll}
0 & & & \\
a_{21} & 0 & & \\
a_{31} & a_{32} & & \\
\cdots & \cdots & \cdots & \cdots \\
a_{s1} & a_{s2} \ldots & a_{ss-1} & 0 \\
\hline
\alpha_1 & \alpha_2 \ldots & \alpha_{s-1} & \alpha_s
\end{array}$$

For an autonomous Stratonovich SODE (1) we obtain by a straightforward generalization of (3) the class of methods

$$\begin{aligned}
&Y_0 = X_0 \\
&Y_{n+1} = Y_n + h_n \sum_{i=1}^{s} \alpha_i f(H_i) + \Delta W_n \sum_{i=1}^{s} \gamma_i g(H_i)
\end{aligned} \tag{4}$$

with

$$H_i = Y_n + h_n \sum_{j=1}^{i-1} a_{ij} f(H_j) + \Delta W_n \sum_{j=1}^{i-1} b_{ij} g(H_j),$$

$i = 1, \ldots, s$, where $\Delta W_n = W_{t_{n+1}} - W_{t_n} = \int_{t_n}^{t_{n+1}} \circ \, \mathrm{d}W_s$ is the increment of the Wiener process from t_n to t_{n+1} and b_{ij}, $\gamma_i \in \mathbb{R}$ for $1 \leqslant i \leqslant s$, $1 \leqslant j \leqslant i-1$. Again we set $b_{ij} = 0$ for $j \geqslant i$. A SRK method of this type is thus symbolized by the tableau

$$\begin{array}{|llll|llll}
0 & & & & 0 & & & \\
a_{21} & 0 & & & b_{21} & 0 & & \\
a_{31} & a_{32} & & & b_{31} & b_{32} & & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_{s1} & a_{s2} \ldots & a_{ss-1} & 0 & b_{s1} & b_{s2} \ldots & b_{ss-1} & 0 \\
\hline
\alpha_1 & \alpha_2 \ldots & \alpha_{s-1} & \alpha_s & \gamma_1 & \gamma_2 \ldots & \gamma_{s-1} & \gamma_s
\end{array}$$

Rümelin [17] has shown, however, that the local error of a method which contains only the increments of the Wiener process as stochastic components converges to 0 in the mean square sense with order at most 1.5. (See [14] for the different notions of convergence in the stochastic setting.) To break this order barrier, the class of methods (4) has to be modified in some way so as to include further multiple stochastic integrals of the stochastic Taylor formula apart from just ΔW_n. This has been done by K. Burrage and P.M. Burrage in [2]. They proposed the following class of methods:

$$\begin{aligned}
&Y_0 = X_0 \\
&Y_{n+1} = Y_n + h_n \sum_{i=1}^{s} \alpha_i f(H_i) + \sum_{i=1}^{s} \left(\gamma_i^{(1)} J_1 + \gamma_i^{(2)} \frac{J_{10}}{h_n} \right) g(H_i)
\end{aligned} \tag{5}$$

with

$$H_i = Y_n + h_n \sum_{j=1}^{i-1} a_{ij}\, f(H_j) + \sum_{j=1}^{i-1} \left(b_{ij}^{(1)} J_1 + b_{ij}^{(2)} \frac{J_{10}}{h_n} \right) g(H_j)$$

for $i = 1, \ldots, s$. Here

$$J_1 = \int_{t_n}^{t_{n+1}} \circ\, \mathrm{d}W_s, \qquad J_{10} = \int_{t_n}^{t_{n+1}} \int_{t_n}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2,$$

and $b_{ij}^{(1)}, b_{ij}^{(2)}, \gamma_i^{(1)}, \gamma_i^{(2)} \in \mathbb{R}$ for $1 \leqslant i \leqslant s$, $1 \leqslant j \leqslant i-1$. Once again we set $b_{ij}^{(1)} = b_{ij}^{(2)} = 0$ for $j \geqslant i$, $B^{(1)} = (b_{ij}^{(1)})$, $B^{(2)} = (b_{ij}^{(2)})$, $\gamma^{(1)\mathrm{T}} = (\gamma_i^{(1)})$ and $\gamma^{(2)\mathrm{T}} = (\gamma_i^{(2)})$. Besides the random variable J_1, the class of methods (5) also contains the random variable J_{10}. A method of type (5) is symbolized by the tableau

$$\begin{array}{|llll|llll|llll}
0 & & & & 0 & & & & 0 & & & \\
a_{21} & 0 & & & b_{21}^{(1)} & 0 & & & b_{21}^{(2)} & 0 & & \\
a_{31} & a_{32} & & & b_{31}^{(1)} & b_{32}^{(1)} & & & b_{31}^{(2)} & b_{32}^{(2)} & & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_{s1} & a_{s2} \ldots & a_{ss-1} & 0 & b_{s1}^{(1)} & b_{s2}^{(1)} \ldots & b_{ss-1}^{(1)} & 0 & b_{s1}^{(2)} & b_{s2}^{(2)} \ldots & b_{ss-1}^{(2)} & 0 \\
\hline
\alpha_1 & \alpha_2 \ldots & \alpha_{s-1} & \alpha_s & \gamma_1^{(1)} & \gamma_2^{(1)} \ldots & \gamma_{s-1}^{(1)} & \gamma_s^{(1)} & \gamma_1^{(2)} & \gamma_2^{(2)} \ldots & \gamma_{s-1}^{(2)} & \gamma_s^{(2)}
\end{array} \tag{6}$$

Note that the class of methods (4) is contained in (5). The rest of this section is concerned with the problem of determining the strong order of convergence of SRK methods (5). In the case of RK methods for deterministic problems the order of accuracy is found by comparing the Taylor series expansion of the approximate solution to the Taylor series expansion of the exact solution over one step assuming exact initial values. In 1963 Butcher [3] introduced the theory of rooted trees in order to compare these two Taylor series expansions in a systematic way. In [2] K. Burrage and P.M. Burrage have extended this idea of using rooted trees to the stochastic setting. They used the set of bi-coloured rooted trees, i.e., the set of rooted trees with black (deterministic) and white (stochastic) nodes to derive a Stratonovich Taylor series expansion of the exact solution and a Stratonovich Taylor series expansion of the approximation defined by the numerical method (5). By comparing these two expansions, they could prove the following theorem:

Theorem 1. *The SRK method* (5) *is of strong order* 1.0, *if*

$$\alpha^{\mathrm{T}} e = 1,$$

$$\gamma^{(1)\mathrm{T}}(e, d, b) = (1, -\gamma^{(2)\mathrm{T}} b, \tfrac{1}{2}),$$

$$\gamma^{(2)\mathrm{T}}(e, d) = (0, 0).$$

The SRK method is of strong order 1.5, *if in addition*

$$\alpha^{\mathrm{T}}(d,b)=(1,0),$$

$$\gamma^{(1)\mathrm{T}}(c,b^2,B^{(1)}b,d^2,B^{(2)}d)=(1,\tfrac{1}{3},\tfrac{1}{6},-2\gamma^{(2)\mathrm{T}}bd,-\gamma^{(2)\mathrm{T}}(B^{(2)}b+B^{(1)}d)),$$

$$\gamma^{(2)\mathrm{T}}(c,b^2,B^{(1)}b,d^2,B^{(2)}d)=(-1,-2\gamma^{(1)\mathrm{T}}bd,-\gamma^{(1)\mathrm{T}}(B^{(2)}b+B^{(1)}d),0,0).$$

Here, $e^{\mathrm{T}}=(1,\dots,1)$, $c=Ae$, $b=B^{(1)}e$ *and* $d=B^{(2)}e$.

Proof. See [2], but note that the orders of convergence, which are given there and are claimed to be orders of strong convergence, are in fact orders of the local error in the mean square sense. By [16] one has to subtract $\frac{1}{2}$ to get the order of the global error in the mean square sense and Jensen's inequality [1] shows that this order gives a lower bound for the order of strong convergence of the method. □

3. An embedded stochastic Runge–Kutta method

An embedded SRK method consists of two SRK methods which both use the same function values of f and g. We are thus looking for a scheme of coefficients

$$\begin{array}{|l|l|l}
0 & 0 & 0 \\
a_{21}\ \ 0 & b^{(1)}_{21}\ \ 0 & b^{(2)}_{21}\ \ 0 \\
a_{31}\ a_{32} & b^{(1)}_{31}\ b^{(1)}_{32} & b^{(2)}_{31}\ b^{(2)}_{32} \\
\dots\dots\dots\dots & \dots\dots\dots\dots & \dots\dots\dots\dots \\
a_{s1}\ a_{s2}\ \dots\ a_{ss-1}\ \ 0 & b^{(1)}_{s1}\ b^{(1)}_{s2}\ \dots\ b^{(1)}_{ss-1}\ \ 0 & b^{(2)}_{s1}\ b^{(2)}_{s2}\ \dots\ b^{(2)}_{ss-1}\ \ 0 \\
\hline
\alpha_1\ \alpha_2\ \dots\ \alpha_{s-1}\ \alpha_s & \gamma^{(1)}_1\ \gamma^{(1)}_2\ \dots\ \gamma^{(1)}_{s-1}\ \gamma^{(1)}_s & \gamma^{(2)}_1\ \gamma^{(2)}_2\ \dots\ \gamma^{(2)}_{s-1}\ \gamma^{(2)}_s \\
\hat{\alpha}_1\ \hat{\alpha}_2\ \dots\ \hat{\alpha}_{s-1}\ \hat{\alpha}_s & \hat{\gamma}^{(1)}_1\ \hat{\gamma}^{(1)}_2\ \dots\ \hat{\gamma}^{(1)}_{s-1}\ \hat{\gamma}^{(1)}_s & \hat{\gamma}^{(2)}_1\ \hat{\gamma}^{(2)}_2\ \dots\ \hat{\gamma}^{(2)}_{s-1}\ \hat{\gamma}^{(2)}_s
\end{array} \tag{7}$$

such that

$$Y_1=Y_0+h\sum_{i=1}^{s}\alpha_i f(H_i)+\sum_{i=1}^{s}\left(\gamma_i^{(1)}J_1+\gamma_i^{(2)}\frac{J_{10}}{h}\right)\,g(H_i) \tag{8}$$

is of strong order p, and

$$\hat{Y}_1=Y_0+h\sum_{i=1}^{s}\hat{\alpha}_i\,f(H_i)+\sum_{i=1}^{s}\left(\hat{\gamma}_i^{(1)}J_1+\hat{\gamma}_i^{(2)}\frac{J_{10}}{h}\right)\,g(H_i) \tag{9}$$

is of strong order $\hat{p}<p$. Using the same terminology as in the deterministic setting we call such a method an embedded SRK method of strong order $p(\hat{p})$. We choose $\hat{p}=1.0$. By Theorem 1 the

method given by the tableau

$$\begin{array}{|cc|cc|cc}
0 & & 0 & & 0 & \\
\frac{2}{3} & 0 & \frac{2}{3} & 0 & 0 & 0 \\
\hline
\frac{1}{4} & \frac{3}{4} & \frac{1}{4} & \frac{3}{4} & 0 & 0
\end{array} \tag{10}$$

has strong order of convergence $\hat{p} = 1.0$. Since $B^{(2)} = 0$ and $\gamma^{(2)} = 0$, method (10) is actually a method of type (4). K. Burrage and P.M. Burrage [2] have shown that method (10) is optimal in that class regarding the principal local truncation error. We choose now $p = 1.5$. Since it is not possible to construct a SRK method of type (5) of order $p = 1.5$ with $s \leqslant 3$ (see [2]), we set $s = 4$. Accordingly, we try to find a scheme of coefficients

$$\begin{array}{|cccc|cccc|cccc}
0 & & & & 0 & & & & 0 & & & \\
\frac{2}{3} & 0 & & & \frac{2}{3} & 0 & & & 0 & 0 & & \\
a_{31} & a_{32} & 0 & & b_{31}^{(1)} & b_{32}^{(1)} & 0 & & b_{31}^{(2)} & b_{32}^{(2)} & 0 & \\
a_{41} & a_{42} & a_{43} & 0 & b_{41}^{(1)} & b_{42}^{(1)} & b_{43}^{(1)} & 0 & b_{41}^{(2)} & b_{42}^{(2)} & b_{43}^{(2)} & 0 \\
\hline
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \gamma_1^{(1)} & \gamma_2^{(1)} & \gamma_3^{(1)} & \gamma_4^{(1)} & \gamma_1^{(2)} & \gamma_2^{(2)} & \gamma_3^{(2)} & \gamma_4^{(2)}
\end{array} \tag{11}$$

which complies with the conditions of Theorem 1. We have 27 free parameters and there are 18 equations to be satisfied. We choose the deterministic part of (11) such that it yields a RK method of order 4. This ensures that our method works well in the case of small stochastic influence. If we require in addition to $a_{21} = \frac{2}{3}$ that $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1$, $\alpha_1 = \alpha_2$ and $\alpha_3 = \alpha_4$ we get

$$\begin{array}{|cccc}
0 & & & \\
\frac{2}{3} & 0 & & \\
\frac{1}{12} & \frac{1}{4} & 0 & \\
-\frac{5}{4} & \frac{1}{4} & 2 & 0 \\
\hline
\frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8}
\end{array}$$

as the only solution for the deterministic part of (11). The remaining system, which consists of 17 equations ($\alpha^{\mathrm{T}} e = 1$ is already fulfilled) with 18 free parameters, was solved using MAPLE. This leads to a number of possible methods and the following method was selected due to its symmetry in $B^{(2)}$:

$$\begin{array}{|cccc|cccc|cccc}
0 & & & & 0 & & & & 0 & & & \\
\frac{2}{3} & 0 & & & \frac{2}{3} & 0 & & & 0 & 0 & & \\
\frac{1}{12} & \frac{1}{4} & 0 & & -\frac{1}{2} & -\frac{1}{6} & 0 & & 1 & 1 & 0 & \\
-\frac{5}{4} & \frac{1}{4} & 2 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 1 & 1 & 0 & 0 \\
\hline
\frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & -\frac{1}{4} & \frac{3}{4} & 0 & \frac{1}{2} & \frac{3}{4} & -\frac{3}{4} & \frac{3}{4} & -\frac{3}{4}
\end{array} \tag{12}$$

The constructed embedded SRK method of strong order 1.5(1.0) is thus symbolized by the tableau

$$\begin{array}{|cccc|cccc|cccc}
0 &&&& 0 &&&& 0 &&& \\
\frac{2}{3} & 0 &&& \frac{2}{3} & 0 &&& 0 & 0 && \\
\frac{1}{12} & \frac{1}{4} & 0 && -\frac{1}{2} & -\frac{1}{6} & 0 && 1 & 1 & 0 & \\
-\frac{5}{4} & \frac{1}{4} & 2 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 1 & 1 & 0 & 0 \\
\hline
\frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & -\frac{1}{4} & \frac{3}{4} & 0 & \frac{1}{2} & \frac{3}{4} & -\frac{3}{4} & \frac{3}{4} & -\frac{3}{4} \\
\frac{1}{4} & \frac{3}{4} & 0 & 0 & \frac{1}{4} & \frac{3}{4} & 0 & 0 & 0 & 0 & 0 & 0
\end{array} \tag{13}$$

The deterministic part of (13) is an embedded RK method of order 4(2). Therefore, the inequalities

$$\hat{p} < p \leqslant 2\hat{p},$$

which were derived by Deufelhard and Bornemann [6] for a deterministic algorithm that uses two approximations of order p and $\hat{p}$ for step size control, are satisfied.

4. Step size control

Using the difference between the two approximations, given by the embedded SRK method of the previous section, as an estimate for the local error we now want to write a code which automatically adjusts the step size in order to achieve a prescribed tolerance of the local error. Whenever a step is tried for which the estimate of the local error is greater than the prescribed tolerance, the step is rejected and a new step with a smaller step size is tried. This leads to difficulties in the simulation of the random variables J_1 and J_{10} which occur in (8) and (9). In Section 4.1 it is shown how these difficulties can be overcome. Next, in Section 4.2 the automatic step size control mechanism is introduced. Finally, Section 4.3 deals with the question as to whether numerical approximations defined by our algorithm converge to the true solution of the given SODE (1).

4.1. Simulation of the random variables J_1 and J_{10}

Since $J_1 = \int_{t_0}^{t_0+h} \circ\, \mathrm{d}W_s$ and $J_{10} = \int_{t_0}^{t_0+h} \int_{t_0}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2$, the two-dimensional random variable (J_1, J_{10}) is normally distributed with expectation and covariance matrix

$$E((J_1,J_{10})) = (0,0) \quad \text{and} \quad \mathrm{Cov}((J_1,J_{10})) = \begin{pmatrix} h & \frac{1}{2}h^2 \\ \frac{1}{2}h^2 & \frac{1}{3}h^3 \end{pmatrix}$$

(see [14]). Accordingly, one gets (J_1, J_{10}) by means of the transformation

$$J_1 = N_1\sqrt{h} \quad \text{and} \quad J_{10} = \frac{1}{2}\sqrt{h^3}\left(N_1 + \frac{1}{\sqrt{3}}N_2\right) \tag{14}$$

with two independent standard normally distributed random variables N_1 and N_2.

Remark 2 (*Simulation of* (J_1,J_{10})). We use an inversive congruential generator (with parameters $p=2^{31}-1$, $a=b=1$, see [8]) for generating a sequence of independent uniformly distributed pseudorandom numbers in the interval $[0,1)$. By the polar method [7], we transform in each step two successive pseudorandom numbers into two independent standard normally distributed pseudorandom numbers and we obtain two pseudorandom numbers which simulate a realization of (J_1,J_{10}) by transformation (14).

Whenever the algorithm tries a step of step size h, the simulation according to Remark 2 yields a value $j_1 \in \mathbb{R}$ for the random variable $J_1 = \int_{t_0}^{t_0+h} \circ\, \mathrm{d}W_s$ and a value $j_{10} \in \mathbb{R}$ for the random variable $J_{10} = \int_{t_0}^{t_0+h} \int_{t_0}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2$. In case the chosen accuracy criterion for the local error is not met, the algorithm has to repeat the step with a smaller step size $\tilde{h}<h$. Thus the problem of simulating

$$\left(\int_{t_0}^{t_0+\tilde{h}} \circ\, \mathrm{d}W_s, \int_{t_0}^{t_0+\tilde{h}} \int_{t_0}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2\right) \tag{15}$$

under the condition

$$\left(\int_{t_0}^{t_0+h} \circ\, \mathrm{d}W_s, \int_{t_0}^{t_0+h} \int_{t_0}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2\right) = (j_1, j_{10}) \tag{16}$$

arises. A method for the simulation of

$$\int_{t_0}^{t_0+\tilde{h}} \circ\, \mathrm{d}W_s \quad \text{under the condition} \quad \int_{t_0}^{t_0+h} \circ\, \mathrm{d}W_s = j_1$$

has been proposed by Lévy [15]. As far as we know there is no corresponding method for the simulation of the two-dimensional random variable (15) under the condition (16). We decided therefore to simulate (J_1,J_{10}) according to Remark 2 relative to a time discretization $t_0<t_1<\cdots<t_N=T$, $T\in\mathbb{N}$, with a fixed step size h of the form

$$h=\frac{1}{2^{m_{\max}}} \quad \text{for } m_{\max}\in\mathbb{N}.$$

Values of the same realization of (J_1,J_{10}) to step sizes of the form $1/2^m$, $0\leqslant m<m_{\max}$, are then obtained recursively by the following consideration: Let $t_1<t_2<t_3$. We set $J_{1,t_i,t_j} = \int_{t_i}^{t_j} \circ\, \mathrm{d}W_s$ and $J_{10,t_i,t_j} = \int_{t_i}^{t_j} \int_{t_i}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2$ for $i,j=1,2,3$. Then

$$\begin{aligned}
J_{1,t_1,t_3} &= J_{1,t_1,t_2} + J_{1,t_2,t_3},\\
J_{10,t_1,t_3} &= \int_{t_1}^{t_3} \int_{t_1}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2\\
&= \int_{t_1}^{t_2} \int_{t_1}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2 + \int_{t_2}^{t_3} \int_{t_1}^{s_2} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s_2\\
&= J_{10,t_1,t_2} + \int_{t_2}^{t_3} \left(\int_{t_1}^{t_2} \circ\, \mathrm{d}W_{s_1} + \int_{t_2}^{s_2} \circ\, \mathrm{d}W_{s_1}\right) \mathrm{d}s_2\\
&= J_{10,t_1,t_2} + J_{1,t_1,t_2}(t_3-t_2) + J_{10,t_2,t_3}.
\end{aligned}$$

Accordingly, during the numerical integration we can use all time steps of the form $t_{\text{start}} \rightsquigarrow t_{\text{end}}$ with

$$t_{\text{start}} = i + k/2^m \quad \text{and} \quad t_{\text{end}} = i + (k+1)/2^m, \tag{17}$$

where $i \in \mathbb{N},\ 0 \leqslant i \leqslant T-1,\ m \in \mathbb{N},\ 0 \leqslant m \leqslant m_{\max}$ and $k \in \mathbb{N},\ 0 \leqslant k \leqslant 2^m - 1$.

4.2. *Automatic step size control*

Whenever a starting step size h has been chosen, method (13) of Section 3 yields two approximations to the solution, y_1 and $\hat{y}_1$. An estimate of the error for the less precise result $\hat{y}_1$ is $|y_1 - \hat{y}_1|$. We require that the step size control routine accepts only steps with

$$|y_1 - \hat{y}_1| \leqslant \text{tol}, \tag{18}$$

where tol denotes the desired tolerance. We have chosen

$$\text{tol} = \text{Atol} + \max\{|y_0|, |y_1|\}\,\text{Rtol}, \tag{19}$$

where Atol and Rtol are tolerances prescribed by the user (relative errors are considered for Atol $= 0$, absolute errors for Rtol $= 0$). For choosing a step size we proceed as follows: As a measure of the error we take

$$\text{err} = |y_1 - \hat{y}_1|/\text{tol} \tag{20}$$

and compare err to 1 in order to find an optimal step size. Since $\hat{p} = 1.0$, the local error of the less precise method converges in the mean square sense to 0 with order 1.5. It follows therefore from Jensen's inequality that the expectation of the local error converges to 0 with order 1.5, too. As $|y_1 - \hat{y}_1|$ gives an estimate for the local error, we assume

$$\text{err} \approx C\,h^{1.5} \tag{21}$$

for some constant $C > 0$. Note that we use thereby an estimate for the average local error as a rough estimate for the local error in a particular realization. Since we require err ≈ 1 for an optimal step size $\tilde{h}_{\text{opt}}$, we get $1 \approx C\tilde{h}_{\text{opt}}^{1.5}$. This implies

$$\tilde{h}_{\text{opt}} = h\left(\frac{1}{\text{err}}\right)^{1/1.5}$$

(see [5]). We multiply $\tilde{h}_{\text{opt}}$ by a safety factor fac < 1 (for example fac $= 0.8$) so that the step size h_{opt}

$$h_{\text{opt}} = \text{fac}\,h\left(\frac{1}{\text{err}}\right)^{1/1.5} \tag{22}$$

will be acceptable the next time with high probability. In the deterministic setting the following method for choosing a new step size is well known (see for example [11]): If err > 1, the step with step size h is rejected and the computations are repeated with the step size $h_{\text{opt}} < h$. In case err $\leqslant 1$, the computed step is accepted and the solution is advanced with y_1 as new initial value and a new step is tried with step size h_{opt}. But on account of the special structure of the available time steps one cannot transfer this method directly to the stochastic setting. New step sizes are only derived from previous ones by halving or doubling. Therefore we propose the following procedure: In case $\mathrm{err} > 1$, the step with step size h is rejected and a new step with step size $h/2$ is tried, as long as $h/2 \geqslant 1/2^{m_{\max}}$, otherwise the code stops. If $\mathrm{err} \leqslant 1$, the step is accepted. For the next step one takes y_1 as new initial value and the step size h either remains unchanged or is doubled. We double the step size only if the following three conditions are fulfilled, where the first and second condition result from the special structure of the available time steps, and the third condition comes from (22):

(1) $2h \leqslant 1$.

(2) The current time has to be the initial point of an admissible time step of length $2h$. For example, if an initial step of length 0.125 is taken at time $t = 0$, then the second step, taken at time $t = 0.125$, cannot be of length 0.25, since steps of length 0.25 can only be taken over the time intervals $[0, 0.25], [0.25, 0.5], \ldots$.

(3) We require that $h_{\mathrm{opt}} \geqslant 2h$, hence by (22) that

$$\mathrm{fac}\left(\frac{1}{\mathrm{err}}\right)^{1/1.5} \geqslant 2,$$

which together with (19) and (20) yields

$$|y_1 - \hat{y}_1| \leqslant \left(\frac{\mathrm{fac}}{2}\right)^{1.5} (\mathrm{Atol} + \max\{|y_0|, |y_1|\}\,\mathrm{Rtol}).$$

If we doubled the step size after each accepted step as long as the structure of the available time steps allowed for an increase in step size at that point and if we did not require condition 3 to be complied with, the double step size would be too large in many cases and would consequently result in a rejection of this step (see the negative numerical results in [9]). Condition 2 ensures that the step size cannot be increased after a step which was carried out right after a step rejection. This is advisable according to [18]. The fact that the step size is at most doubled after an accepted step of step size h, even if $h_{\mathrm{opt}} > 2h$, prevents the code from too large step increases and contributes to its safety. On the other hand, after a rejected step of step size h the step size is only halved even for $h_{\mathrm{opt}} < \frac{1}{2}h$. This prevents the code from an unnecessary increase in computational work.

4.3. Convergence of the embedded SRK method with step size control

In Sections 4.1 and 4.2 we have outlined an algorithm using variable time steps. Naturally the question arises as to whether this algorithm converges to the true solution of the given SODE (1). In general, proofs of strong convergence of numerical methods, like the proof of Theorem 1, are based on the assumption of a fixed step size or at least on the assumption that the discretization points are all stopping times for the forward motion. In the case of an algorithm with automatic step size control the discretization points are obviously not stopping times, since it is only once a step has been taken and the error estimated that the decision is made to continue or to retreat and take a smaller time step. But the following proposition ensures that the embedded SRK method (13) with the automatic step size control mechanism described in the previous subsection yields approximations which converge to the true solution of (1).

Proposition 3. *Each SRK method* (5) *which satisfies the* 6 *conditions of Theorem* 1 *for an order* 1.0 *strong method yields approximations converging to the solution of* (1), *as long as the maximum step size converges to* 0, *even if the discretization points are not stopping times.*

Proof. Let β_0 and β_1 denote the rooted trees which consist only of a black and white root, respectively, and β_{11} the rooted tree which consists of a white root and another white node. By [2] and with the notation introduced there we obtain as Stratonovich Taylor series expansion of the numerical method (5)

$$\begin{aligned} Y(t_0+h) &= X_{t_0} + \Phi(\beta_0)F(\beta_0)h + \Phi(\beta_1)F(\beta_1) + \Phi(\beta_{11})F(\beta_{11})\tfrac{1}{2} + \mathcal{O}(h^{1.5}) \\ &= X_{t_0} + \alpha^{\mathrm{T}} e f(X_{t_0})h + z^{\mathrm{T}} e g(X_{t_0}) + 2z^{\mathrm{T}} Z e g' g(X_{t_0})\tfrac{1}{2} + \mathcal{O}(h^{1.5}), \end{aligned}$$

where $Z = B^{(1)}J_1 + B^{(2)}J_{10}/h$, $z = \gamma^{(1)}J_1 + \gamma^{(2)}J_{10}/h$ and $\mathcal{O}(h^{1.5})$ denotes the order in the mean square sense. From the 6 conditions of Theorem 1 for an order 1.0 method it follows that $\alpha^{\mathrm{T}}e = 1$, $z^{\mathrm{T}}e = J_1$ and $z^{\mathrm{T}}Ze = \frac{1}{2}J_1^2 = J_{11}$ and thus Corollary 4.4 in [9] yields the assertion. □

5. Numerical results

In this section, numerical results from the implementation of the embedded SRK method (13) of strong order 1.5(1.0) with the automatic step size control mechanism presented in Section 4 are compared to those from the implementation of the SRK method (12) of strong order 1.5 with fixed step size. These methods will be denoted by M1 and M2, respectively. We compare the average error of these two methods when using the same amount of computational work. To this end we proceed as follows: As test problems we take several problems from [14], for which the exact solution in terms of the Wiener process is known. We first compute with method M1 and prescribed tolerances Atol and Rtol for $N = 1000$ trajectories of the Wiener process approximate solutions over the interval $[0, 10]$. We denote by $\bar{S}_{\text{tried}}$ and $\bar{S}_{\text{taken}}$ the average number of steps tried and taken, respectively. By comparing the approximate solutions at $t = 0, 1, 2, \dots, 10$ to the exact solutions at the same points of time, we get the average error at these times. This is exactly the error one wants to minimize by constructing methods of strong convergence. This error will be indicated by a solid line in the following figures. We want to compare this error of method M1 to the error of method M2 when using the same amount of computational work. Since the amount of computational work of both methods is more or less directly proportional to the number of steps tried and since, because of the embedded structure of M1, a step with M1 is as expensive as a step with M2, we choose as fixed step size h for method M2

$$h = T/\bar{S}_{\text{tried}}. \tag{23}$$

Consequently, the numbers of steps taken by M2 for an integration along each trajectory equals the average number of steps tried by an integration with M1.

Remark 4. Method M1 can adjust the number of integration steps according to the structure of the respective trajectory of the Wiener process. For an integration along a trajectory which makes the integration difficult M1 uses more steps than for an integration along a trajectory with a simpler

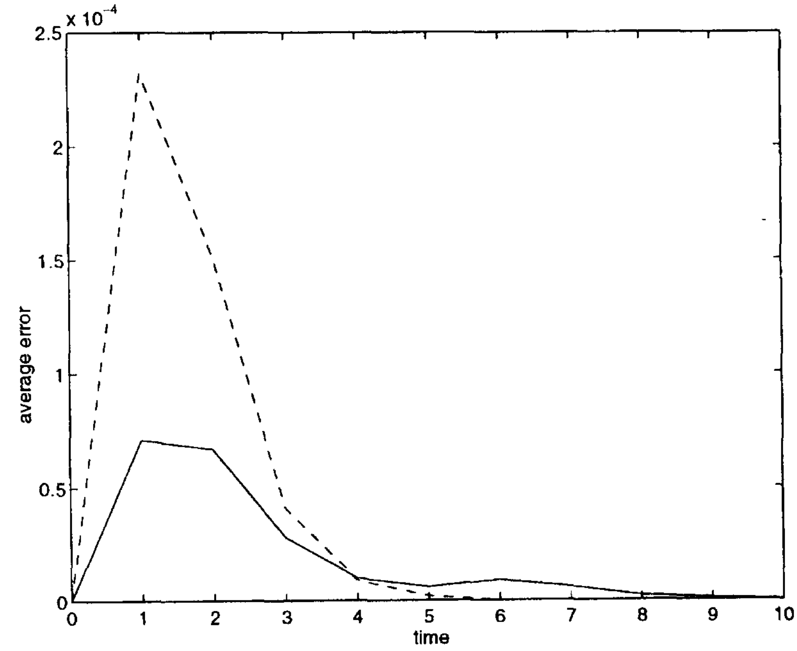

Fig. 1. $\beta = 0.1$, Atol $= 10^{-5}$, Rtol $= 10^{-4}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}} = 0.82$.

structure. Whereas a method which uses only a fixed step size cannot adjust the number of steps to the structure of the different trajectories.

We compute with method M2 and fixed step size (23) once again for $N = 1000$ trajectories of the Wiener process the approximate and the exact solution at $t = 0, 1, 2, \ldots, 10$ and determine the average error at these times. This error will be indicated by a dashed line in the following figures.

Test Problem 1 ([14, Problem 4.4.46])

$$\mathrm{d}X_t = -(1 + \beta^2 X_t)(1 - X_t^2)\,\mathrm{d}t + \beta(1 - X_t^2)\,\mathrm{d}W_t, \quad X_0 = 0,$$

with solution

$$X_t = \frac{\exp(-2t + 2\beta W_t) - 1}{\exp(-2t + 2\beta W_t) + 1}. \tag{24}$$

This problem was solved numerically twice, first with $\beta = 0.1$ and secondly with $\beta = 1$. This demonstrates the variation in emphasis of the stochastic and deterministic parts of the SODE. For $\beta = 0.1$ (relatively weak stochastic influence) solution (24) converges rapidly to -1 and is almost constant after $t = 4$. The errors of method M1 and M2 are shown in Fig. 1. For $t \leqslant 4$ method M1 is superior to method M2, whereas for $t > 4$ M2 gives better results than M1, since M2 takes smaller steps in $[4, 10]$. For $\beta = 1$ (moderately large stochastic influence) the solution shows the asymptotic behaviour considerably later, especially when the Wiener process takes on large positive values. In this case method M1 is superior to method M2 on the whole interval $[0, 10]$ (see Fig 2). For both $\beta = 0.1$ and $\beta = 1$ the maximum value of the error of method M2 is larger than that of M1, namely

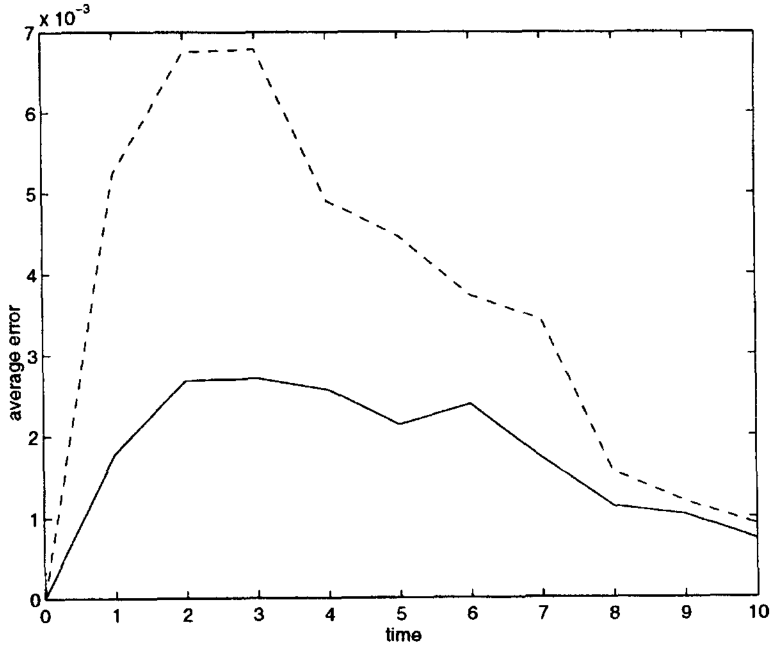

Fig. 2. $\beta = 1$, Atol $= 10^{-4}$, Rtol $= 10^{-3}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}} = 0.77$.

for $\beta = 0.1$ approximately 3 times as large as that for M1 and for $\beta = 1$ approximately 2.5 times as large.

Test Problem 2 ([14, Problem 4.4.6])

$$\mathrm{d}X_t = \alpha X_t\,\mathrm{d}t + \beta X_t\,\mathrm{d}W_t, \quad X_0 = 1,$$

with solution

$$X_t = \exp((\alpha - \tfrac{1}{2}\beta^2)t + \beta W_t).$$

For $\alpha = 1$ and $\beta = 0.5$ we get $X_t = \exp(0.875t + 0.5W_t)$, essentially an increasing exponential function. Since this function does not have "smooth" and "nonsmooth" sections, step size control is not of much use. This is shown in Fig. 3. If we choose $\alpha = -0.5$ and $\beta = 1$, we get $X_t = \exp(-t + W_t)$, a decreasing exponential function disturbed by the Wiener process. Method M1 is superior to M2 until $t = 7$. The maximum value of the error with M2 is almost twice as large as the corresponding value of M1 (see Fig. 4).

Test Problem 3 ([14, Problem 4.4.31])

$$\mathrm{d}X_t = -0.25X_t(1 - X_t^2)\mathrm{d}t + 0.5\,(1 - X_t^2)\,\mathrm{d}W_t, \quad X_0 = 0,$$

with solution

$$X_t = \tanh(0.5\,W_t).$$

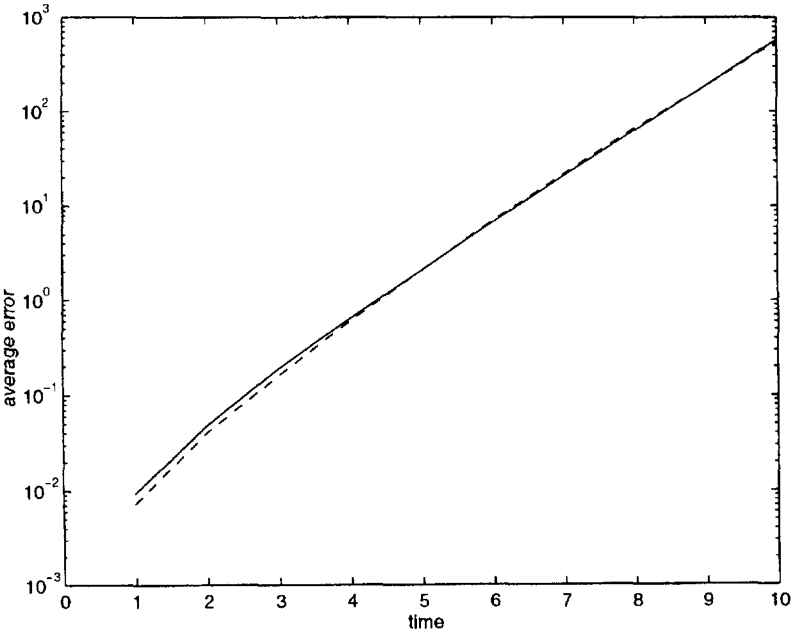

Fig. 3. $\alpha=1$, $\beta=0.5$, Atol $=10^{-3}$, Rtol $=10^{-3}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}}=0.77$.

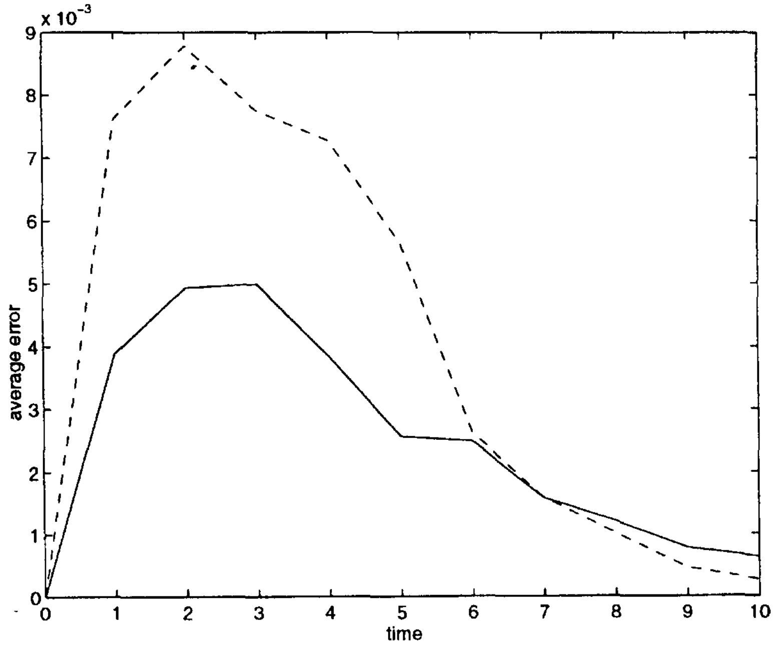

Fig. 4. $\alpha=-0.5$, $\beta=1$, Atol $=10^{-3}$, Rtol $=10^{-3}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}}=0.78$.

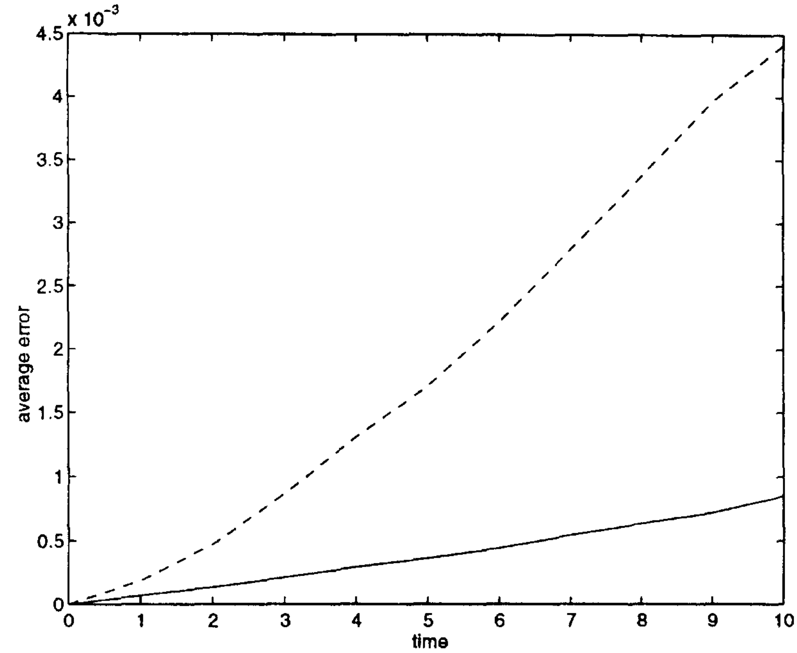

Fig. 5. Atol $= 10^{-4}$, Rtol $= 10^{-3}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}} = 0.75$.

The solution of this SODE is a Wiener process which is scaled by the factor 0.5 and which is, in addition to the scaling, damped by the tangens hyperbolicus. The effectiveness of a variable step size implementation for that problem is shown in Fig. 5.

Test Problem 4 ([14, Problem 4.4.28])

$$\mathrm{d}X_t = \cos X_t \sin^3 X_t \,\mathrm{d}t - \sin^2 X_t \,\mathrm{d}W_t, \quad X_0 = \frac{\pi}{2},$$

with solution

$$X_t = \operatorname{arccot}(W_t).$$

The arcuscotangens function damps — in a manner a bit different to that of the tangens hyperbolicus — the function $t \mapsto W_t$. Also for this test problem method M1 is clearly superior to method M2 (see Fig. 6).

The results obtained show that a significant gain in efficiency can be achieved by step size control not only in the numerical solution of deterministic differential equations, but also in the numerical solution of stochastic differential equations. However, future work will be needed to extend this method of step size control to multidimensional stochastic differential equations.

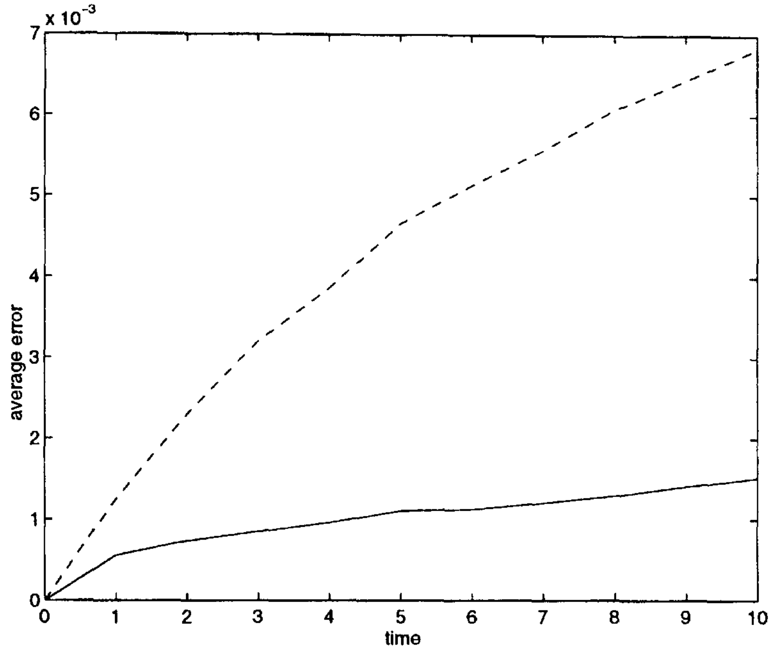

Fig. 6. Atol $= 10^{-4}$, Rtol $= 10^{-4}$, $\bar{S}_{\text{taken}}/\bar{S}_{\text{tried}} = 0.75$.

Acknowledgements

The author wishes to express her gratitude to Prof. Dr. J. Lehn for suggesting the problem and for many stimulating conversations during the preparation of this paper.

References

[1] H. Bauer, Wahrscheinlichkeitstheorie, de Gruyter, Berlin, New York, 1991.

[2] K. Burrage, P.M. Burrage, High strong order explicit Runge–Kutta methods for stochastic ordinary differential equations, Appl. Numer. Math. 22 (1996) 81–101.

[3] J.C. Butcher, Coefficients for the study of Runge–Kutta integration processes, J. Austral. Math. Soc. 3 (1963) 185–201.

[4] J.C. Butcher, On Runge–Kutta processes of high order, J. Austral. Math. Soc. 4 (1964) 179–194.

[5] F. Ceschino, Modification de la longueur du pas dans l' intégration numérique par les méthodes à pas liés, Revue Assoc. Franc. 4 (1961) 101–106.

[6] P. Deufelhard, F. Bornemann, Numerische Mathematik II, de Gruyter, Berlin, New York, 1994.

[7] L.P. Devroye, Non-Uniform Random Variate Generation, Springer, New York, 1986.

[8] J. Eichenauer-Herrmann, Inversive congruential pseudorandom numbers: a tutorial, Internat. Statist. Rev. 60 (1992) 167–176.

[9] J.G. Gaines, T.J. Lyons, Variable step size control in the numerical solution of stochastic differential equations, SIAM J. Appl. Math. 5 (1997) 1455–1484.

[10] K. Gustafsson, M. Lundh, G. Söderlind, A PI stepsize control for the numerical solution of ordinary differential equations, BIT 28 (1988) 270–287.

[11] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations, Springer, Berlin, 1987.

[12] N. Hofmann, Beiträge zur schwachen Approximation stochastischer Differentialgleichungen, Dissertation, Humboldt-Universität, Berlin, 1995.
[13] P.E. Kloeden, E. Platen, Stratonovich and Itô Taylor expansion, Math. Nachr. 151 (1991) 33–50.
[14] P.E. Kloeden, E. Platen, Numerical Solution of Stochastic Differential Equations, Springer, Berlin, 1995.
[15] P. Lévy, Processus Stochastiques et Mouvement Brownien, Monographies des Probabilités, Gauthier-Villars, Paris, 1948.
[16] G.N. Milstein, Numerical Integration of Stochastic Differential Equations, Kluwer, Dordrecht, 1995.
[17] W. Rümelin, Numerical treatment of stochastic differential equations, SIAM J. Numer. Anal. 19 (1982) 604–613.
[18] L.F. Shampine, H.A. Watts, The art of writing a Runge–Kutta code, Appl. Math. Comput. 5 (1979) 93–121.