COMMUN. MATH. SCI.
Vol. 10, No. 4, pp. 1317–1329

FAST COMMUNICATION

TWO-STAGE STOCHASTIC RUNGE-KUTTA METHODS FOR STOCHASTIC DIFFERENTIAL EQUATIONS WITH JUMP DIFFUSION*

CAN HUANG†, JING SHI‡, AND ZHIMING ZHANG§

Abstract. In this paper, we propose explicit two-stage Runge-Kutta schemes of strong order one for solutions of stochastic differential equations driven by jump-diffusion processes. By using rooted trees, we obtain the convergence rate. Our numerical tests verify our theoretical results.

Key words. Stochastic differential equation, numerical approximation, stochastic Runge-Kutta methods, jump-diffusion.

AMS subject classifications. 65C30, 60H30.

1. Introduction

This paper is concerned with strong approximation of stochastic differential equations (SDE) driven by Levy jump-diffusion processes.

There has recently been rising attention paid to stochastic jump-diffusion models, or more general Levy processes, in physical as well as social sciences. For instance, in financial modeling the jump component can be used to model shocks and other event-driven uncertainties ([19]), which has driven many recent research activities (see, for example, [12]). In physics and chemistry, Levy processes have been exploited to model continuous random walks beyond Gaussian approximation (diffusion scaling) ([4, 28]). One successful application, among others, is the control of translational motion of atoms by laser cooling in the deep quantum regime ([3, 5]).

In general the numerical simulation of the Levy processes is highly non-trivial except for some particular cases (for example, stable processes, etc.). One popular strategy is perhaps to use diffusion to approximate "small jumps", which amounts to approximating Levy processes by jump-diffusion ([2, 26]). Then Levy-driven process is approximated by a jump-diffusion stochastic differential equation.

There has been a considerable amount of work, and much progress has been made, in the numerical approximation of SDE of diffusion type, which can be found in [15, 20, 21], for example. Both strong and weak discrete time approximations of various types and orders have been constructed. The development of numerical methods for SDE with jumps is still limited, partly due to the complicated structure of multiple stochastic integrals when both Poisson and Wiener measures are present. The stochastic Taylor expansion for semi-martingales was constructed in [22, 23], and was used to derive strong Taylor schemes ([13, 14, 17, 18, 6, 7]). The recent book by Platen and his colleague summarizes the efforts along this line [24]. In the

*Received: February 22, 2011; accepted (in revised form); January 10, 2012. Communicated by Tiejun Li.

†Corresponding author. Department of Mathematics, Wayne State University, Detroit, MI 48202, USA (huangcan@msu.edu).

‡Department of Mathematics, Wayne State University, Detroit, MI 48202, USA (jshi@math.wayne.edu).

§Department of Mathematics, Wayne State University, Detroit, MI 48202, USA (ag7761@wayne.edu).

construction of strong Taylor schemes of higher order, derivatives of coefficients are required. It is desirable to have derivative-free higher order schemes.

Runge-Kutta methods are a class of derivative free discretization schemes. Both strong and weak stochastic Runge-Kutta methods (SRK) for diffusion SDE have been considered in detail in [8, 9, 16, 25, 27], and the order condition has been obtained by colored trees analysis. In this paper, we derive a two-stage explicit SRK of strong order 1 for jump-diffusion SDE.

The rest of this paper is organized as follows: Section 2 begins with some preliminary notes, and in Section 3, we will compose a class of strong order 1 SRK methods for (2.1). In Section 4, we illustrate the convergence of our SRK schemes with three examples and compare their numerical behavior with two existing strong order 1 schemes from [13] and [6], respectively. Section 5 concludes the paper and suggests topics for future work.

2. Preliminaries

In this section, we review the notion of jump-diffusion processes and stochastic Runge-Kutta methods.

We work with a finite-time horizon $[0,T]$ for some $T>0$. Let Q be a subset of $R\backslash\{0\}$, and let $W(t)$ be a standard Wiener process on R. $N(dt,dz)$ is a Poisson random measure on $[0,\infty)\times Q$ with mean measure $\nu(dz)$. Furthermore we assume ν is state independent and finite. Let $f,g:R\to R$, and $h:R\times Q\to R$. Consider the scalar Stratonovich SDE (SSDE) given by

$$\begin{cases} dx(t)=f(x(t-))dt+g(x(t-))\circ dW(t)+\int_Q h(x(t-),z)N(dt,dz), \\ x(0)=x_0, \end{cases} \tag{2.1}$$

or, in its integral form,

$$\begin{aligned} x(t)=x_0&+\int_0^t f(x(s-))ds+\int_0^t g(x(s-))\circ dW(s) \\ &+\int_0^t\int_Q h(x(s-),z)N(ds,dz). \end{aligned} \tag{2.2}$$

Throughout the paper, we assume that W and N are independent.

By the properties of Poisson random measures, the last term in equation (2.2) can be constructed as

$$\int_0^t\int_Q h(x(t),z)N(dt,dz)=\sum_{\tau_n\le t,\ z_n\in Q} h(x(\tau_n-),z_n),$$

where $\{(\tau_n,z_n),n=1,2,\cdots\}$ is the Poisson point process generated by N (see, for example, [1]). Next, we define the the strong order of convergence of a discrete approximation.

DEFINITION 2.1. *Let x_N be the numerical approximation to $x(t_N)$ after N steps with constant stepsize h. Then y_N is said to converge strongly to $y(t_N)$ with order p if $\exists C>0$ (independent of h) and $\delta>0$ such that*

$$E(|x_N-x(t_N)|^2)\le Ch^{2p},\quad h\in(0,\delta).$$

Now let us introduce stochastic Runge-Kutta methods (SRK) for equation (2.1). For stochastic differential equations of the form

$$dx(t)=f(x(t))dt+g(x(t))\circ dW(t),$$

Rümelin [27] gave a class of SRK methods of the form

$$Y_i = y_n + h\sum_{j=1}^{s} a_{ij} f(Y_j) + J_1 \sum_{j=1}^{s} b_{ij} g(Y_j), \quad i = 1,2,\cdots,s,$$
$$y_{n+1} = y_n + h\sum_{j=1}^{s} \alpha_j f(Y_j) + J_1 \sum_{j=1}^{s} \beta_j g(Y_j), \tag{2.3}$$

where s is the number of states, h is the step size, and $J_1 = \int_{t_n}^{t_{n+1}} \circ dW(s)$. Applying the tableau introduced by Butcher [11], we obtain

$$\begin{array}{c|cc} & A & B \\ \hline & \alpha^T & \beta^T \end{array},$$

where $A = (a_{ij})$ and $B = (b_{ij})$ are $s \times s$ matrices of real elements while $\alpha^T = (\alpha_1, \alpha_2, \cdots, \alpha_s)$ and $\beta^T = (\beta_1, \beta_2, \cdots, \beta_s)$ are row vectors.

Similar to the above case, we propose a class of SRK methods

$$Y_i = y_n + h\sum_{j=1}^{s} a_{ij} f(Y_j) + J_1 \sum_{j=1}^{s} b_{ij} g(Y_j) + J_2 \sum_{z} \sum_{j=1}^{s} c_{ij} h(Y_j, z), \quad i = 1,2,\cdots,s.$$
$$y_{n+1} = y_n + h\sum_{j=1}^{s} \alpha_j f(Y_j) + J_1 \sum_{j=1}^{s} \beta_j g(Y_j) + J_2 \sum_{z} \sum_{j=1}^{s} \gamma_j h(Y_j, z), \tag{2.4}$$

where $J_2 = \int_{t_n}^{t_{n+1}} \int_Q N(ds,dz)$, $\gamma^T = (\gamma_1, \gamma_2, \cdots, \gamma_s)$.

The corresponding Butcher tableau is

$$\begin{array}{c|ccc} & A & B & C \\ \hline & \alpha^T & \beta^T & \gamma^T \end{array},$$

where $C = (c_{ij})$ is a $s \times s$ matrix. In the article, we only consider two-stage methods. Therefore, our explicit Butcher tableau is of the form

$$\begin{array}{c|cc|cc|cc} & 0 & 0 & 0 & 0 & 0 & 0 \\ & a_1 & 0 & b_1 & 0 & c_1 & 0 \\ \hline & \alpha_1 & \alpha_2 & \beta_1 & \beta_2 & \gamma_1 & \gamma_2 \end{array}. \tag{2.5}$$

3. Two-stage stochastic RK methods for jump-diffusion processes

Now let us focus on the equation (2.2). A general version of Itô's formula states that a given function V of the solution x can be written as

$$V(x(t)) = V(x_0) + \int_0^t L_0 V(x(s))ds + \int_0^t L_1 V(x(s))dW(s) + \int_0^t \int_Q L_2 V(x(s))N(ds,dz), \tag{3.1}$$

where

$$L_0 V = f\frac{\partial V}{\partial x},\ L_1 V = g\frac{\partial V}{\partial x},\ L_2 V = V(x(t) + h(x(t), z)) - V(x(t)). \tag{3.2}$$

Applying (3.1) and (3.2) with $V(x(t)) = x(t)$ gives

$$x(t) = x_0 + f(x_0)J_0 + g(x_0)J_1 + \sum_{z \in Q} h(x_0, z)J_2 + f(x_0)f'(x_0)J_{00} + g(x_0)f'(x_0)J_{10}$$

$$+\sum_{z\in Q}(f(x_0+h(x_0,z))-f(x_0))J_{20}+f(x_0)g'(x_0)J_{01}+g(x_0)g'(x_0)J_{11}$$
$$+\sum_{z\in Q}(g(x_0+h(x_0,z))-g(x_0))J_{21}$$
$$+\sum_{z\in Q}\sum_{z_1}(h(x_0+h(x_0,z_1),z)-h(x_0,z))J_{22}+R. \tag{3.3}$$

Here R is the remainder term and J_{j_1,j_2} represents the multiple integral, where the integration is with respect to ds if $j_i=0$, $\circ dW(t)$ if $j_i=1$, or $N(dt,dz)$ if $j_i=2$. Hence, for example,

$$J_{21}=\int_0^t\int_0^s\int_Q N(ds_1,dz)\circ dW(s).$$

Now, we are ready to express the solution of (2.2) by applying of the set of tri-colored trees. Let T denote the colored trees with a root $\odot$ and which may also contain some deterministic nodes $\oplus$ (τ) and Wiener process nodes $\oslash$ (σ) and Poisson process nodes $\otimes$ (μ). Thus, if $t_1,\cdots,t_m$ are tri-colored trees then $\langle t_1,\cdots,t_m\rangle$, $[t_1,\cdots,t_m]$, $\{t_1,\cdots,t_m\}$, and $(t_1,\cdots,t_m)$ are trees in which $t_1,\cdots,t_m$ are each joined by a single branch to $\odot$, $\oplus$, or $\oslash$, or $\otimes$. Define two functionals $F_1(V)$ and $F_2(V)$ by $\frac{\partial V}{\partial x}$ and $V(x+h(x,z))-V(x)$, respectively. Then, an elementary differential can be defined recursively for any $t\in T$ by

$$F(\tau)(y)=f(y),\qquad F(\sigma)(y)=g(y),\qquad F(\mu)(y)=h(y),$$
$$F(t)(y)=\begin{cases} f^{(m)}[F(t_1)(y),\cdots,F(t_m)(y)], & t=[t_1,\cdots,t_m],\\ g^{(m)}[F(t_1)(y),\cdots,F(t_m)(y)], & t=\{t_1,\cdots,t_m\},\\ h^{(m)}[F(t_1)(y),\cdots,F(t_m)(y)], & t=(t_1,\cdots,t_m),\end{cases} \tag{3.4}$$

where $f^{(m)}$ means F_1 and F_2 act on the function f m times together and $g^{(m)}$ and $h^{(m)}$ are defined similarly.

Next, an elementary weight needs to be associated with each elementary differential. Here, we associate 0 with a deterministic node $\oplus$, 1 with a Wiener process node $\oslash$, and 2 with a Poisson process node $\otimes$. Then, if we let

$$J_0(F)=\int_0^t F(y(s))ds,$$
$$J_1(F)=\int_0^t F(y(s))\circ dW(s),$$
$$J_2(F)=\int_0^t\int_Q F(h(y(s),z)N(ds,dz), \tag{3.5}$$

then elementary weights can be defined recursively as

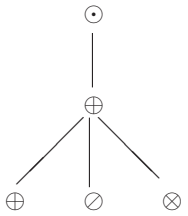
$$\theta(\tau)=J_0(1),\quad \theta(\sigma)=J_1(1),\quad \theta(\mu)=J_2(1),$$
$$\theta(t)=\begin{cases} J_0\Big(\prod_{j=1}^m\theta(t_j)\Big), & t=[t_1,\cdots,t_m],\\ J_1\Big(\prod_{j=1}^m\theta(t_j)\Big), & t=\{t_1,\cdots,t_m\},\\ J_2\Big(\prod_{j=1}^m\theta(t_j)\Big), & t=(t_1,\cdots,t_m).\end{cases} \tag{3.6}$$

Denote by $d(t)$ the number of deterministic nodes, by $w(t)$ the number of Wiener process node, and by $n(t)$ the number of Poisson process nodes in t. Then the order $ord(t)$ for a tree t is defined as

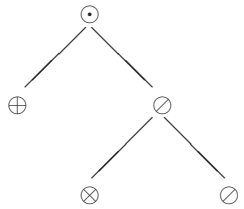
$$ord(t) = d(t) + \frac{1}{2}w(t) + \frac{1}{2}n(t). \tag{3.7}$$

Let $L(t)$ be the number of leaves of a tree t, where a leaf is defined as a node without any "children", i.e. no successor node. Let $\alpha(t)$ be the number of ways of labeling t with a set of $ord(t)$ ordered symbols such that along each outwardly directed arc the labels increase; then we can write $x(t)$ (as in [8]) as

$$x(t) = \sum_{t\in T} \alpha(t)F(t)(x_0)\theta(t). \tag{3.8}$$



Sample tree 1



Sample tree 2

It is easy to see that the mathematical expression of of the first sample tree is $\langle[\oplus,\oslash,\otimes]\rangle$, $ord(\langle[\oplus,\oslash,\otimes]\rangle)=3$, $\alpha(\langle[\oplus,\oslash,\otimes]\rangle)=1$, and $L(\langle[\oplus,\oslash,\otimes]\rangle)=3$, and that the mathematical expression of the second tree is $\langle\oplus,\{\otimes,\oslash\}\rangle$, $ord(\langle\oplus,\{\otimes,\oslash\}\rangle)=\frac{5}{2}$, $\alpha(\langle\oplus,\{\otimes,\oslash\}\rangle)=4$, and $L(\langle\oplus,\{\otimes,\oslash\}\rangle)=3$.

REMARK 3.1. In the expression of $ord(t)$ in (3.7), a deterministic node is weighted with 1 because we get a h if we integrate 1 with respect to ds on $[0,h]$, both a Wiener process node and a jump node are weighted with $\frac{1}{2}$ because of isometries $E[\int_0^h 1\cdot dW(t)]^2 = h$ and $E[\int_0^h \int_Q 1\cdot N(dt,dz)]^2 = \lambda\nu(Q)h$.

For the rest of this section, we focus on the expression of the solution of SRK methods, i.e. equation (2.4). For simplicity, we will replace t_n with t_0, and for a given $t = t_0 + h$, x_{n+1} will be expressed as $X(t)$ with intermediate states $X_1(s),\cdots,X_s(t)$.

From (2.4),

$$\begin{aligned}
X(t) &= X_0 + \sum_{j=1}^{s}\alpha_j f(X_j(t)) + \sum_{j=1}^{s}\beta_j g(X_j(t)) + \sum_{z\in A}\sum_{j=1}^{s}\gamma_j h(X_j(t),z)\\
&= X_0 + \sum_{j=1}^{s}\alpha_j[f(X_0) + L_0 f(X_0) + L_1 f(X_0) + L_2 f(X_0) + H.O.T]\\
&\quad + \sum_{j=1}^{s}\beta_j[g(X_0) + L_0 g(X_0) + L_1 g(X_0) + L_2 g(X_0) + H.O.T]\\
&\quad + \sum_{z\in A}\sum_{j=1}^{s}\gamma_j[h(X_0,z) + L_0 h(X_0,z) + L_1 h(X_0,z) + L_2 h(X_0,z) + H.O.T]
\end{aligned}$$

$$
\begin{aligned}
&= X_0 + \alpha^T e f(X_0) + \alpha^T A e f(X_0) f'(X_0) + \sum_{z\in A} \alpha^T C e [f(X_0 + h(X_0, z)) - f(X_0)] \\
&\quad + \alpha^T B e f'(X_0) g(X_0) + \beta^T e g(X_0) + \beta^T A e f(X_0) g'(X_0) + \beta^T B e g'(X_0) g(X_0) \\
&\quad + \sum_{z\in A} \beta^T C e [g(X_0 + h(X_0, z)) - g(X_0)] + \sum_{z\in A} \gamma^T e h(X_0, z) \\
&\quad + \sum_{z\in A} \gamma^T A e f(X_0) h'(X_0, z) + \sum_{z\in A} \gamma^T B e g(X_0) h'(X_0, z) \\
&\quad + \sum_{z\in A} \sum_{z_1\in A} \gamma^T C e [h(X_0 + h(X_0, z_1), z) - h(X_0, z)] + H.O.T.
\end{aligned}
$$

Furthermore, as in [8], it can be shown that the Taylor expansion for the numerical solution can be written as

$$
X(t) = \sum_{t\in T} \Phi(t) F(t)(X_0) \frac{h^{\rho_1(t)}}{\rho(t)!}, \tag{3.9}
$$

where $\Phi(t)$ is defined recursively by

$$
k(\phi) = e, \tag{3.10}
$$

$$
\Phi(t) = \begin{cases}
\rho(t)\alpha^T \prod_{i=1}^{m} k(t_i), & t = [t_1, \cdots, t_m], \\
\rho(t)\beta^T \prod_{i=1}^{m} k(t_i), & t = \{t_1, \cdots, t_m\}, \\
\rho(t)\gamma^T \prod_{i=1}^{m} k(t_i), & t = (t_1, \cdots, t_m),
\end{cases} \tag{3.11}
$$

where

$$
k(t) = \begin{cases}
\rho(t) A \prod_{i=1}^{m} k(t_i), & t = [t_1, \cdots, t_m], \\
\rho(t) B \prod_{i=1}^{m} k(t_i), & t = \{t_1, \cdots, t_m\}, \\
\rho(t) C \prod_{i=1}^{m} k(t_i), & t = (t_1, \cdots, t_m),
\end{cases} \tag{3.12}
$$

and where $\rho(t)$ is the number of leaves of t while $\rho_1(t)$ is the number of deterministic leaves of t.

Hence, the truncation error at $t = t_n$ is

$$
e_n = \sum_{t\in T} \left(\alpha(t)\theta(t) - \Phi(t) \frac{h^{\rho_1(t)}}{\rho(t)!} \right) F(t)(x(t_n)). \tag{3.13}
$$

Thus, if

$$
E(|e_n|^2) \le C h^{2p}, \tag{3.14}
$$

then a method will have strong order p.

To find a good RK method, we list the local error of all trees with $ord(t) \le 1.5$; see table 3.1.

TABLE 3.1. *Local truncation errors*

$\sharp$	t	$e(t)$
1	$\oplus$	$J_0-\alpha^T e$
2	$\oslash$	$J_1-\beta^T e$
3	$\otimes$	$J_2-\gamma^T e$
4	$[\oslash]$	$J_{10}-\alpha^T Be$
5	$\{\oplus\}$	$J_{01}-\beta^T Ae$
6	$[\otimes]$	$J_{20}-\alpha^T Ce$
7	$(\oplus)$	$J_{02}-\gamma^T Ae$
8	$(\oslash)$	$J_{12}-\gamma^T Be$
9	$\{\otimes\}$	$J_{21}-\beta^T Ce$
10	$\{\oslash\}$	$J_{11}-\beta^T Be$
11	$(\otimes)$	$J_{22}-\gamma^T Ce$
12	$\{\{\oslash\}\}$	$J_{111}-\beta^T B^2 e$
13	$\{\{\otimes\}\}$	$J_{211}-\beta^T BCe$
14	$\{(\oslash)\}$	$J_{121}-\beta^T CBe$
15	$\{(\otimes)\}$	$J_{221}-\beta^T C^2 e$
16	$(\{\oslash\})$	$J_{112}-\gamma^T B^2 e$
17	$(\{\otimes\})$	$J_{212}-\gamma^T BCe$
18	$((\oslash))$	$J_{122}-\gamma^T CBe$
19	$((\otimes))$	$J_{222}-\gamma^T C^2 e$
20	$\{\oslash,\oslash\}$	$J_{111}-\frac{1}{2}\beta^T(Be)^2$
21	$\{\oslash,\otimes\}$	$J_{211}-\frac{1}{2}\beta^T(Be)(Ce)$
22	$\{\otimes,\otimes\}$	$J_{221}-\frac{1}{2}\beta^T(Ce)^2$
23	$(\oslash,\oslash)$	$J_{112}-\frac{1}{2}\gamma^T(Be)^2$
24	$(\oslash,\otimes)$	$J_{212}-\frac{1}{2}\gamma^T(Be)(Ce)$
25	$(\otimes,\otimes)$	$J_{222}-\frac{1}{2}\gamma^T(Ce)^2$

THEOREM 3.2. *There is no 1.5 order Runge-Kutta method of form (2.5) for equation (2.2).*

Proof. If there exists an order 1.5 scheme, then all trees of order 1 in Table 3.1 must be accurately approximated, so that the remainder, in the L^2 sense, is of order 1.5 or higher. In particular, tree 9 in Table 3.1 is order 1. Hence, we require

$$E(J_{21}-\beta^T Ce)^2=O(h^3) \quad \text{or} \tag{3.15}$$

$$E(J_{21}^2)-2E(\beta^T CeJ_{21})+E(\beta^T Ce)^2)=O(h^3). \tag{3.16}$$

Let $\alpha=\hat{\alpha}h$, $\beta=\hat{\beta}J_1$, $\gamma=\hat{\gamma}J_2$, $A=\hat{A}h$, $B=\hat{B}J_1$, and $C=\hat{C}J_2$, and let M be an event such that only one jump occurs in $[0,h]$. Now let us calculate the left-hand side

of (3.16):

$$\begin{aligned}E(J_{21}^2) &= E\Big(\int_0^h \int_0^s \int_Q N(ds_1,dz)\circ dW(s)\Big)^2,\\ &= E\Big(\int_0^h N(s,Q)\circ dW(s)\Big)^2\\ &= E\Big(\Big(\int_0^h N(s,Q)\circ dW(s)\Big)^2|M\Big)P(M)+H.O.T\\ &\simeq \lambda\frac{h^2}{2}\nu(Q),\end{aligned} \tag{3.17}$$

$$\begin{aligned}&E(\beta^T C e J_{21})\\ &= E(\hat\beta^T \hat C e J_1 J_2 J_{21})\\ &= \hat\beta^T \hat C e E\Big(\int_0^h \circ dW(s)\int_0^h\int_Q N(ds,dz)\int_0^h\int_0^s\int_Q N(ds_1,dz)\circ dW(s)\Big)\\ &= \hat\beta^T \hat C e E\Big(\int_0^h \circ dW(s)\int_0^h\int_Q N(ds,dz)\int_0^h\int_0^s\int_Q N(ds_1,dz)\circ dW(s)|M\Big)P(M)\\ &\quad +H.O.T\\ &\simeq \hat\beta^T \hat C e\cdot\lambda\frac{h^2}{2}\nu(Q).\end{aligned} \tag{3.18}$$

Finally,

$$\begin{aligned}E(\beta^T C e)^2 &= (\hat\beta^T \hat C e)^2 E(J_1 J_2)^2\\ &= (\hat\beta^T \hat C e)^2 E(W(h)N(h,Q))^2\\ &= (\hat\beta^T \hat C e)^2 E(W(h)^2 N(h,Q)^2|M)P(M)+H.O.T\\ &\simeq (\hat\beta^T \hat C e)^2\lambda h^2\nu(Q),\end{aligned} \tag{3.19}$$

where $e=(1,1)^T$. Here, we take advantage of the fact that, for a Poisson process, if there are n arrivals on [0,h] then the n arrival times are uniformly distributed on the interval $[0,h]$ and also the fact that there exists only one jump on the interval if h is small enough.

Writing $\phi=\hat\beta^T\hat C e$, (3.16) becomes

$$h^2\left(\phi^2-\phi+\frac{1}{2}\right)=0.$$

Obviously, this has no real root. Hence, an order 1.5 scheme does not exist. This completes the proof. ∎

Now let us construct an order 1 scheme. Note that from table 3.1, trees 2 and 3 are of order $\frac{1}{2}$, so we require

$$\hat\beta^T e=1,\qquad \hat\gamma^T e=1. \tag{3.20}$$

The order 1 trees in table 3.1 are trees 1, 8, 9, 10, 11. For optimization, one should find parameters to minimize

$$E(J_0-\alpha^T e)^2,\quad E(J_{21}-\beta^T C e)^2,$$

$$E(J_{11}-\beta^T Be)^2,\quad E(J_{22}-\gamma^T Ce)^2,\quad E(J_{12}-\gamma^T Be)^2. \tag{3.21}$$

as Burrage did in [8, 9]. For simplicity, we choose

$$\gamma_2=0,\ b_1=\frac{1}{2},\ \beta_2=1,\ c_1=1. \tag{3.22}$$

Then, we obtain the tableau

$$\begin{array}{c|cc|cc|cc} & 0 & 0 & 0 & 0 & 0 & 0 \\ & 1 & 0 & \frac{1}{2} & 0 & 1 & 0 \\ \hline & \alpha_1 & \alpha_2 & 0 & 1 & 1 & 0 \end{array}, \tag{3.23}$$

where $\alpha_1+\alpha_2=1$, i.e

$$\begin{aligned} Y&=y_n+hf(y_n)+\frac{J_1}{2}g(y_n)+J_2\sum_z h(y_n,z),\\ y_{n+1}&=y_n+h(\alpha_1 f(y_n)+\alpha_2 f(Y))+J_1 g(Y)+J_2\sum_z h(y_n,z). \end{aligned} \tag{3.24}$$

4. Numerical experiments

In this section, we implement (3.24) with $\alpha_1=\frac{1}{3},\ \alpha_2=\frac{2}{3}$, i.e.

$$\begin{array}{c|cc|cc|cc} & 0 & 0 & 0 & 0 & 0 & 0 \\ & 1 & 0 & \frac{1}{2} & 0 & 1 & 0 \\ \hline & \frac{1}{3} & \frac{2}{3} & 0 & 1 & 1 & 0 \end{array}. \tag{4.1}$$

We compare the numerical errors of the SRK scheme above, Gardoń's strong order 1 scheme [13], and Bruti-Liberati and Platen's adapted strong order 1 scheme [6]. We remark that Gordoń's scheme was originally proposed and analyzed for homogeneous Poisson processes but could be extended to Poisson random measures as is done here. The first two examples in this section are from Gardoń's [14] and the last one is from Bruti-Liberati's work [6]. In this section, for all problems and all methods, 500 trajectories are computed for each step size. Hence, the error

$$E|x(T)-X(T)|=\sqrt{\frac{\sum_{i=1}^{500}|x_i(T)-X_i(T)|^2}{500}}.$$

Example 1. Consider an autonomous linear equation as follows:

$$\begin{cases} dx(t)=\frac{15}{8}x(t)dt+\frac{1}{2}x(t)\circ dW(t)+\int_Q -\frac{1}{10}x(t)N(dt,dz), & t\in[0,1],\\ x(0)=10, \quad \lambda=50. \end{cases} \tag{4.2}$$

The analytical solution of (4.2) is

$$x(t)=10\left(\frac{9}{10}\right)^{N(t)}\exp\left(\frac{15}{8}t+\frac{1}{2}W(t)\right),$$

where $N(t)$ denotes the number of jumps.

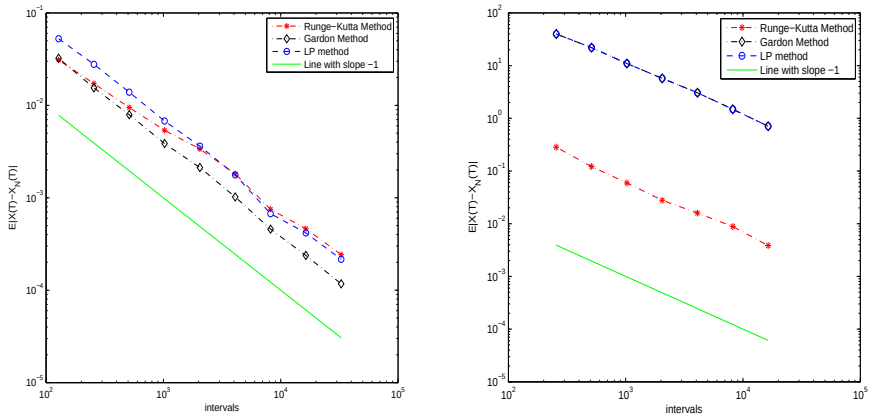


FIG. 4.1. *The portfolio of error for different numerical methods at the end time for Example 1 (Left) and for Example 2 (Right).*

Then the numerical results of the Runge-Kutta method for different step sizes in table 4.1 and errors for three different methods are illustrated in the left part of figure 4.1. From the portfolio for this example, we observe that the errors of the RK method gradually deviate from the line with slope -1 for the first five points. The reason for this is that we obtain our Butcher tableau (4.1) based upon the assumption that there exists only one jump on the time interval if the length of the interval is small enough. However, this condition might be violated for the choice of the first five time steps whereas with refined time steps, the error line of the RK method goes back to the line with slope -1.

TABLE 4.1. *Numerical results of Example 1 and Example 2*

step size	$E\|x(1)-X(1)\|$ (Ex1)	$E\|x(T)-X(T)\|$ (Ex2)
2^{-7}	0.0310	20.5485
2^{-8}	0.0172	0.2829
2^{-9}	0.0095	0.1209
2^{-10}	0.0054	0.0592
2^{-11}	0.0034	0.0278
2^{-12}	0.0018	0.0159
2^{-13}	0.0008	0.0088
2^{-14}	0.0005	0.0038

Example 2. Consider the nonlinear jump-diffusion equation

$$\begin{cases} dx(t) = a(t,x(t))dt + b(t,x(t)) \circ dW(t) + c(t,x)dN(t), & t \in [0, \frac{2}{5}], \\ x(0) = 100, \quad \lambda = 30, \end{cases} \tag{4.3}$$

where

$$
\begin{aligned}
a(t,x) &= \frac{10\pi\cos(10\pi t)}{r(t)}x + 12t\sqrt[3]{r(t)x^2},\\
b(t,x) &= 3\sqrt[3]{r(t)x^2},\\
c(t,x) &= \frac{-3t}{10}\sqrt[3]{r(t)x^2} + \frac{3t^2}{100}\sqrt[3]{r^2(t)x} - \frac{t^3 r(t)}{1000},\\
r(t) &= \frac{11}{10} + \sin(10\pi t),
\end{aligned}
\tag{4.4}
$$

and with analytical solution

$$
x(t) = \left(\frac{11}{10} + \sin(10\pi t)\right)\left(\frac{10}{\sqrt[3]{11}} + 2t^2 + W(t) - \frac{1}{10}\int_0^t s dN_s\right)^3.
$$

Numerical results of different methods are reported in table 4.1 (Runge-Kutta) and the right part of figure 4.1. From the figure, we observe that errors of Gardoń's method are large, which coincides with the numerical results in [14].

TABLE 4.2. *Numerical results of Example 3*

step size	$E\lvert x(1)-X(1)\rvert$ (Example 3)
2^{-7}	0.0435
2^{-8}	0.0231
2^{-9}	0.0098
2^{-10}	0.0053
2^{-11}	0.0027
2^{-12}	0.0013
2^{-13}	0.0007
2^{-14}	0.0003

Example 3. Consider an jump-diffusion example of the form

$$
\begin{cases} dX_t = 2X_t dt + \frac{1}{2}X_t dW(t) + \int_Q X(t)(z-1)N(dt,dz), \\ X(0)=1,\ \lambda = 1 \end{cases}
$$

on $[0,1]$. From [6], this equation admits the explicit solution

$$
X_t = \exp\left(\frac{15}{8}t + \frac{1}{2}W(t)\right)\prod_{i=1}^{p_\phi(t)}\xi_i, \tag{4.5}
$$

where $p_\phi(t)$ is the number of jumps up to time 1 and $\xi_i = e^{\zeta_i}$ is the i-th lognormal realization of the jump size, with $\zeta_i \sim \mathcal{N}(\rho,\varsigma)$ denoting an independent Gaussian random

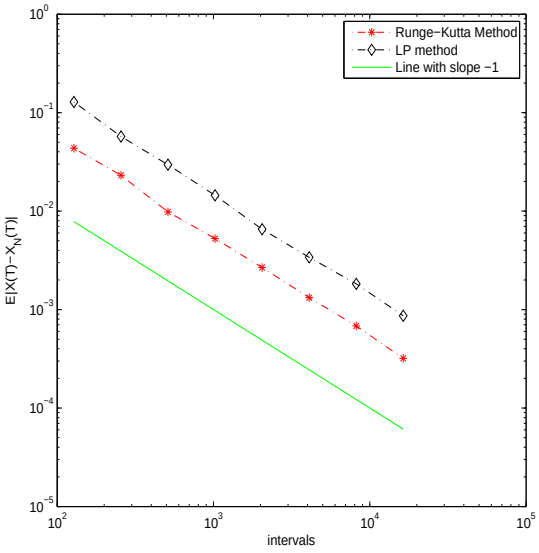


FIG. 4.2. *The portfolio of error for different numerical methods at the end time for Example 3.*

variable with mean ρ and variance ζ. The distribution of jumps is

$$P(z)=\begin{cases} 0.25, & \text{if } z=2,\\ 0.5, & \text{if } z=1,\\ 0.25, & \text{if } z=0.1. \end{cases}$$

Note that Gardoń's method can not be applied to this example since the coefficient of the jump term is jump-dependent, so we compare our Runge-Kutta scheme with Bruti-Liberati and Platen's scheme only. Table 4.2 shows the numerical errors for our Runge-Kutta method with different step sizes.

5. Concluding remarks

In this paper, we construct a two-stage strong order 1 stochastic Runge-Kutta scheme to solve stochastic differential equations with jump-diffusion. By the colored rooted tree analysis, we obtain an extended tableau as in [8, 9]. During the process of construction, we also prove that order 1.5 scheme does not exist for (2.5). In the numerical experiment section, we compare the numerical errors of our SRK scheme above with Gardoń's strong order 1 scheme [13] and Bruti-Liberati and Platen's adapted strong order 1 scheme [6]. From numerical experiments, we observe that our stochastic Runge-Kutta scheme performs as well of better than the other two schemes. Another merit of our scheme is that it is given by a tableau, and thus is concise and derivative free. As is known, the computational cost is high for this type of approximation when the jump intensity is high. Future work includes extending the current scheme to multidimensional problems as well as high jump intensities.

REFERENCES

[1] D. Applebaum, *Lévy Processes and Stochastic Calculus, Second edition*, Cambridge University Press, 2009.

[2] S. Asmussen and J. Rosinski, *Approximations of small jumps of Levy processes with a view towards simulation*, J. Appl. Prob., 38, 482–493, 2001.

[3] F. Bardou, J.-P. Bouchaud, A. Aspect, and C. Cohen-Tannoudji, *Levy Statistics and Laser Cooling*, Cambridge, 2002.

[4] J.-P. Bouchard and A. Georges, *Anomalous diffusion in disordered media: Statistical mechanics, model and physical applications*, Phys. Rep., 195, 127–293, 1990.

[5] H.-P. Breuer and F. Petruccione, *The Theory of Open Quantum Systems*, Oxford, 2002.

[6] N. Bruti-Liberati and E. Platen, *On the Strong Apprximation of Jump-Diffusion Process*, Technical Report, Quantitative Finance Research Paper, 157, University of Technology, Sydney, 2005.

[7] N. Bruti-Liberati and E. Platen, *Strong approximations of stochastic differential equations with jumps*, J. Comput. Appl. Math., 205, 982–1001, 2007.

[8] K. Burrage and P.M. Burrage, *High strong order explicit Runge-Kutta methods for stochastic differenital equations*, Appl. Numer. Math., 22, 81-101, 1996.

[9] K. Burrage and P.M. Burrage, *Order conditions of stochastic Runge-Kutta mehods by B -series*, SIAM J. Numer. Anal, 38, 1626–1646, 2000.

[10] K. Burrage, P.M. Burrage, and T. Tian, *Numerical methods for strong solutions of stochastic differential equations: An overview*, Proc. R. Soc. Lond. A, 460, 373–402, 2004.

[11] J.C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*, London, Wiley, 1987.

[12] R. Cont and P. Tankov, *Financial Modeling with Jump Processes*, Chapman & Hall/CRC, London, Boca Raton, 2004.

[13] A. Gardoń, *The order of approximations for solutions of Itô-type stochastic differential equations with jumps*, Stoch. Anal. Appl., 22, 679–699, 2004.

[14] A. Gardoń, *The order 1.5 approximation for solution of jump-diffusion equations*, Stoch. Anal. Appl., 24, 1147–1168, 2006.

[15] P.E. Kloeden, and E. Platen, *Numerical Solution of Stochastic Differential Equations*, 4th printing, New York: Springer-Verlag, 2010.

[16] C.W. Li, S.C. Wu, and X.Q. Liu, *Discretization of jump stochastic differential equations in terms of multiple stochastic integrals*, J. Comput. Math., 16, 375–384, 1998.

[17] Y. Maghsoodi, *Mean square efficient numerical solution of jump-diffusion stochastic differential equations*, Sankhya, 58(A), 25–47, 1996.

[18] Y. Maghsoodi, *Exact solutions and doubly efficient approximations of jump-diffusion Itô equations*, Stoc. Anal. Appl., 16, 1049–1072, 1998.

[19] R.C. Merton, *Option pricing when underlying stock return rates are discontinuous*, J. Finan. Econ., 3, 141–183, 1976.

[20] G.N. Milstein, *Numerical Integration of Stochastic Differential Equations*, New York: Kluwer, 1995.

[21] G.N. Milstein and M.V. Tretyakov, *Stochastic Numerics for Mathematical Physics*, Springer, 2004.

[22] E. Platen, *An approximation method for a class of Itô processes with jump component*, Liet. Mat. Rink., 22(2), 124–136, 1982.

[23] E. Platen, *A generalized Taylor Formula for solutions of stochastic equations*, Sankhya, 44, 163–172, 1982.

[24] E. Platen and E. Bruti-Liberati, *Numerial Solution of Stochastic Differential Equations with Jumps in Finance*, Springer, 2010.

[25] A. Rößler, *Runge-Kutta methods for the strong approximation of solutions of stochastic differential equtions*, SIAM J. Numer. Anal., 48, 922–952, 2010.

[26] S. Rubenthaler, *Numerical simulation of the solution of a stochastic differential equation driven by a Levy process*, Stoch. Proc. Appl., 103, 311–349, 2002.

[27] W. Rümelin, *Numerical treatment of stochastic differential equations*, SIAM J. Numer. Anal., 19, 604–613, 1982.

[28] W.A. Woyczynski, *Levy processes in the physical sciences*, in Levy Processes: Theory and Applications, O.E. Baarndorff-Nielsen, T. Mikosch and S.I. Resnick (ed.), Birkhauser, 2001.