FinM 345/Stat 390 Stochastic Calculus, Autumn 2009

Floyd B. Hanson, Visiting Professor

Email: fhanson@uchicago.edu

Master of Science in Financial Mathematics Program
University of Chicago

Lecture 6 (Corrected Post-Lecture)
More Compound Jump-Diffusion Calculus

6:30-9:30 pm, 02 November 2009, Kent 120 in Chicago

7:30-10:30 pm, 02 November 2009 at UBS Stamford

7:30-10:30 am, 03 November 2009 at Spring in Singapore

• Normal-Uniform Hybrid Marks:

{Continuing unfinished part of Lecture 5.}

The very, very thin tails of the normal is consequence of the insistence on infinite domain for exact integrals and for using the large number of statistical tests and methods available. Just truncating the normal to finite range does not fatten the tails noticeably. However, an alternate idea is the combine the truncated normal and the uniform distribution, i.e.,

$$\phi_Q^{(\mathrm{nuq})}(q) = \left(\frac{p_u}{b-a} + \frac{p_n \phi_n(q;\mu_n,\sigma_n^2)}{\Phi_n(a,b;\mu_n,\sigma_n^2)}\right) U(q;(a,b)), \tag{6.1}$$

where $a < 0 < b$, $\Phi_n(a,b;\mu_n,\sigma_n^2)$ is the distribution on (a,b), while p_u and p_n are the respective uniform and normal probabilities such that $p_u + p_n = 1$.

- **MATLAB Mark Simulations:**
  - **Uniform on `(a,<b)`:**
    $Q^{(\mathrm{uq})}$ = `a+(b-a)*rand` = `unifrnd(a,b)`.
  - **Normal for `(mu, sigma)`:**
    $Q^{(\mathrm{nq})}$ = `mu+sigma*randn` = `normrnd(mu,sigma)`.
  - **Double-Uniform for `(a`$<0<$`b;p1)`:**
    $Q^{(\mathrm{duq})}$ = `binornd(1,p1)*unifrnd(a,0)`
    `+(1-binornd(1,p1))*unifrnd(0,b)`.
  - **Double-Exponential for `(mu1`$<0<$`mu2;p1)`:**
    $Q^{(\mathrm{deq})}$ = `binornd(1,p1)*exprnd(-mu1)`
    `+(1-binornd(1,p1))*exprnd(mu2)`.
  - **Normal-Uniform for `(a`$<0<$`b;mu,sigma,pu)`:**
    $Q^{(\mathrm{nuq})}$ = `binornd(1,pu)*unifrnd(a,b)+(1-binornd(1,pu))`
    `*AcceptedOnly`{`normrnd(mu,sigma)` $\in$ `(a,b)`}.
  - For fixed probability values, the `binornd(1,p*)` can be replaced by just `p*`, where `p* = p1 or pu`.

FinM 345 Stochastic Calculus:

6. More Compound Jump-Diffusion Calculus:

• *6.1. State-Dependent Compound Jump-Diffusions:*

(Beginning the section corresponding to the nonlinear mark-jump-diffusions and linear simulations cancelled in Lecture 5.)

- **6.1.1 State-Dependent Generalizations for Compound Poisson:**

  The space-time Poisson process is generalized to include state-dependence with $X(t)$ in both the jump-amplitude and the Poisson measure, such that the **jump-amplitude counter** is

$$d\Pi(t;X(t),t) = \int_{\mathcal{Q}} h(X(t),t,q)\mathcal{P}(\overline{\mathrm{dt}},\overline{\mathrm{dq}};X(t),t) \quad (6.2)$$

  on the Poisson mark space $\mathcal{Q}$ with Poisson random measure $\mathcal{P}(\overline{\mathrm{dt}},\overline{\mathrm{dq}};X(t),t)$, which helps to describe the space-time Poisson mechanism and related calculus.

The space-time state-dependent Poisson mark, $Q = q$, is again the underlying random variable for the **state-dependent and mark-dependent jump-amplitude coefficient** $h(x, t, q)$. The double time t arguments of $d\Pi$, dP and $\mathcal{P}$ are not considered redundant for modeling applications, since the first time t or right-continuous time set $\overline{\mathrm{dt}} = [t, t + dt)$ is the usual Poisson jump process implicit time dependence, similarly $\overline{\mathrm{dq}} = [q, q + dq)$ is the appropriate interval for right-continuous marks for consistency, while the second to the right of the semicolon denotes explicit or parametric time dependence paired with explicit state dependence that is known in advance and is appropriate for the application model.

Alternatively, the Poisson zero-one law form may be used, i.e.,

$$d\Pi(t;X(t),t) \overset{\text{dt}}{\underset{\text{zol}}{=}} h(X(t),t,Q)dP(t;Q,X(t),t) \quad (6.3)$$

with the jump of $\Pi(t;X(t),t)$ being

$$[\Pi](T_k) = h(X(T_k^-),T_k^-,Q_k)$$

at jump-time T_k and jump-mark Q_k. The multitude of random variables in this sum means that expectations or other Poisson integrals will be very difficult to calculate even by conditional expectation iterations. **{Caution: the shorthand form $h\Delta P$ should only be used, say for increments, where ZOL holds, else the compound form $\sum_{j=1}^{1+\Delta P} h_j$ must be used, where $h_j = h(X(T_j^-, T_j^-, Q_j)$.}**

Definition 6.1:

The ***conditional expectation of*** $\mathcal{P}(\overline{\mathrm{dt}}, \overline{\mathrm{dq}}; X(t), t)$ *is*

$$\mathrm{E}[\mathcal{P}(\overline{\mathrm{dt}}, \overline{\mathrm{dq}}; X(t), t) | X(t) = x] = \phi_Q(q; x, t) dq \lambda(t; x, t) dt, \quad (6.4)$$

where $\phi_Q(q; x, t)dq$ *is the probability density of the now state-dependent Poisson amplitude mark and the jump rate* $\lambda(t; x, t)$ *now has state-time dependence. In this notation, the relationship to the simple counting process is given by*

$$\int_{\mathcal{Q}} \mathcal{P}(\overline{\mathrm{dt}}, \overline{\mathrm{dq}}; X(t), t) = dP(t; Q, X(t), t).$$

{Comment: When the current state process $X(t)$ is unknown, as it is prior to a solution, it is necessary to take this kind of conditional expectation with further information.}

Hence, when $h(x,t,q) = \widetilde{h}(x,t)$, i.e., independent of the mark q, the space-time Poisson is the simple jump process with mark-independent amplitude,

$$d\Pi(t;X(t),t) \underset{\text{zol}}{\overset{\text{dt}}{=}} \widetilde{h}(X(t),t)dP(t;Q,X(t),t),$$

but with nonunit jumps in general. Note, that we may write $dP(t;Q,X(t),t) = dP(t;X(t),t)$ here, since the Q as a parameter of dP only denotes the Q-generation capabilities of dP. Effectively the same form is obtained when there is a single discrete mark, e.g., $\phi_Q(q) = \delta(q-1)$, so $h(x,t,q) = h(x,t,1)$ always.

Theorem 6.1 Basic Conditional Infinitesimal Moments of the State-Dependent Poisson Process:

$$\begin{aligned} \mathrm{E}[d\Pi(t;X(t),t)|X(t) = x] &= \int_{\mathcal{Q}} h(x,t,q)\phi_Q(q;x,t)dq \\ &\quad \cdot\lambda(t;x,t)dt \qquad (6.5) \\ &\equiv \mathrm{E}_Q[h(x,t,Q)]\lambda(t;x,t)dt \end{aligned}$$

and

$$\begin{aligned} \mathrm{Var}[d\Pi(t;X(t),t)|X(t)=x] &= \int_{\mathcal{Q}} h^2(x,t,q)\phi_Q(q;x,t)dq \\ &\quad \cdot\lambda(t;x,t)dt \qquad (6.6) \\ &\equiv \mathrm{E}_Q[h^2(x,t;Q)]\lambda(t;x,t)dt. \end{aligned}$$

Proof: The justification is the same justification as for (5.25)–(5.26) {(5.26)-(5.27) in textbook}. It is assumed that the jump-amplitude $h(x,t,Q)$ is independently distributed due to Q from the underlying Poisson counting process here, except that this jump in space is conditional on the occurrence of the jump-time of the underlying Poisson process. □

- **6.1.2 State-Dependent Jump-Diffusion SDEs:**

The general, scalar SDE takes the alternate but symbolic forms,

$$\begin{aligned} dX(t) &= f(X(t),t)dt + g(X(t),t)dW(t) \\ &\quad + \int_{\mathcal{Q}} h(X(t),t,q)\mathcal{P}(\overline{\mathrm{dt}},\overline{\mathrm{dq}};X(t),t) \\ &\overset{\mathrm{dt}}{\underset{\mathrm{zol}}{=}} f(X(t),t)dt + g(X(t),t)dW(t) \\ &\quad + h(X(t),t,Q)dP(t;Q,X(t),t) \end{aligned} \tag{6.7}$$

for the state process $X(t)$ with a set of continuous coefficient functions $\{f,g,h\}$. However, the SDE model is just a useful symbolic model for many applied situations, but the more basic model relies on specifying the method of integration.

So for the **Itô forward approximation**,

$$X(t) = X(t_0) + \int_{t_0}^{t} (f(X(s), s)ds + g(X(s), s)dW(s) + h(X(t), s, Q)dP(s; Q, X(s), s)) \quad (6.8)$$

$$\stackrel{\text{ifa}}{=} X(t_0) + \lim_{n\to\infty} \left[\sum_{i=0}^{n} \left(f_i \Delta t_i + g_i \Delta W_i + \sum_{k=P_i+1}^{P_i+\Delta P_i} h_{i,k}\right)\right],$$

where $f_i = f(X_i, t_i)$, $g_i = g(X_i, t_i)$, $h_{i,k} = h(X_i, T_k, Q_k)$, $\Delta t_i = t_{i+1} - t_i > 0$, $t_0 \geq 0$ given, $t_{n+1} = t$, $\Delta P_i = \Delta P(t_i; Q, X_i, t_i)$ and $\Delta W_i = \Delta W(t_i)$. Here, T_k is the kth jump-time and Q_k is the corresponding random mark.

{Note that when the simple infinitesimal Poisson term, hdP, is expanded into increments, the compound form, $\sum_{k=P_i+1}^{P_i+\Delta P_i} h_{i,k}$, had to be used!}

The **conditional infinitesimal moments for the state process** are

$$\mathrm{E}[dX(t)|X(t)=x] = f(x,t)dt + \overline{h}(x,t)\lambda(t;x,t)dt, \quad (6.9)$$

$$\overline{h}(x,t) \equiv \mathrm{E}_Q[h(x,t,Q)],$$

and

$$\mathrm{Var}[dX(t)|X(t)=x] = g^2(x,t)dt + \overline{h^2}(x,t)\lambda(t;x,t)dt, \quad (6.10)$$

$$\overline{h^2}(x,t) \equiv \mathrm{E}_Q[h^2(x,t,Q)]$$

using (6.5), (6.6), (6.7) and the property that the Poisson process is independent of the Wiener process.

The jump in the state at jump time T_k in the underlying Poisson process is

$$[X](T_k) \equiv X(T_k^+) - X(T_k^-) = h(X(T_k^-), T_k^-, Q_k) \quad (6.11)$$

for $k = 1, 2, \ldots$, now depending on the kth mark Q_k at the prejump-time T_k^- at the kth jump.

Rule 6.1. Stochastic Chain Rule for State-Dependent SDEs: *The* ***stochastic chain rule*** *for a sufficiently differentiable function* $Y(t) = F(X(t), t)$ *has the form*

$$
\begin{aligned}
dY(t) &= dF(X(t),t) = F(X(t)+dX(t), t+dt) - F(X(t),t) \\
&= d_{(\mathrm{cont})}F(X(t),t) + d_{(\mathrm{jump})}F(X(t),t) \\
&\stackrel{\mathrm{dt}}{=} F_t(X(t),t)dt + F_x(X(t),t)(f(X(t),t)dt \\
&\quad + g(X(t),t)dW(t)) \\
&\quad + \tfrac{1}{2}F_{xx}(X(t),t)g^2(X(t),t)dt \\
&\quad + \int_{\mathcal{Q}} (F(X(t) + h(X(t),t,q),t) - F(X(t),t)) \\
&\quad \cdot \mathcal{P}(\overline{\mathrm{dt}}, \overline{\mathrm{dq}}; X(t), t)
\end{aligned}
\tag{6.12}
$$

to precision-dt*. It is sufficient that* F *be twice continuously differentiable in* x *and once in* t*.*

• 6.1.3. Linear State-Dependent SDEs:

Let the state-dependent jump-diffusion process satisfy an SDE linear in the state process $X(t)$ with time-dependent rate coefficients

$$dX(t) \underset{\text{zol}}{\overset{\text{dt}}{=}} X(t)(\mu_d(t)dt + \sigma_d(t)dW(t) + \nu(t,Q)dP(t;Q)) \quad (6.13)$$

for $t > t_0$ with $X(t_0) = X_0$ and $\mathrm{E}[dP(t;Q)] = \lambda(t)dt$, where $\mu_d(t)$ denotes the mean and $\sigma_d^2(t)$ denotes the variance of the diffusion process, while Q_k denotes the kth mark and T_k denotes the kth time of the jump process.

Again, using the log-transformation $Y(t) = \ln(X(t))$ and the stochastic chain rule (6.12),

$$\begin{aligned} dY(t) \underset{\text{zol}}{\overset{\text{dt}}{=}} & (\mu_d(t) - \sigma_d^2(t)/2)dt + \sigma_d(t)dW(t) \\ & + \ln\left(1 + \nu(t,Q)\right) dP(t;Q) \end{aligned} \quad (6.14)$$

with immediate integrals

$$Y(t) = \ln(x_0) + \int_{t_0}^{t} dY(s) \tag{6.15}$$

and

$$X(t) = x_0 \exp\left(\int_{t_0}^{t} dY(s)\right), \tag{6.16}$$

or in recursive form,

$$X(t + \Delta t) = X(t) \exp\left(\int_{t}^{t+\Delta t} dY(s)\right). \tag{6.17}$$

• Linear Mark-Jump-Diffusion Simulation Forms:

For simulations, a small time-step, $\lambda_i \Delta t_i \ll 1$, approximation of the recursive form (6.17) would be more useful with $X_i = X(t_i)$, $\mu_i = \mu_d(t_i)$, $\sigma_i = \sigma_d(t_i)$, $\Delta W_i = \Delta W(t_i)$, $\Delta P_i = \Delta P(t_i; Q)$ and the convenient jump-amplitude coefficientapproximaton, $\nu(t, Q) \simeq \nu_0(Q) \equiv \exp(Q) - 1$, i.e.,

$$X_{i+1} \simeq X_i \exp\left((\mu_i - \sigma_i^2/2)\Delta t_i + \sigma_i \Delta W_i\right) Q^{\Delta P_i} \quad (6.18)$$

for $i = 1 : N$ time-steps, where a zero-one jump law approximation has been used, assuming a **small incremental jump count, $\lambda_i \Delta t_i \ll 1$**.

{Otherwise the compound Poisson form must be used and $X_{i+1} \simeq X_i \exp\left((\mu_i - \sigma_i^2/2)\Delta t_i + \sigma_i \Delta W_i\right) \Pi_{j=1}^{\Delta P_i} Q_j$.}

For the diffusion part, it has been shown that

$$\mathrm{E}\left[e^{\sigma_i \Delta W_i}\right] = e^{\sigma_i^2 \Delta t_i / 2},$$

using the completing the square technique. In addition, there is the following lemma for the jump part of the increment version of (6.18) without ZOL.

Lemma 6.1. Jump Term Expectation:

$$\mathrm{E}\left[\Pi_{j=1}^{\Delta P_i}(1 + \nu_0(Q_j))\right] = e^{\lambda_i \Delta t_i \mathrm{E}[\nu_0(Q)]}, \qquad (6.19)$$

where $\mathrm{E}[\Delta P_i] = \lambda_i \Delta t_i$ *and* $\nu_0(Q) = \exp(Q) - 1 > -1$.

Proof: Using given forms, **iterated expectations**, the Poisson distribution and the IID property of the marks Q_k, we then have

$$\begin{aligned}
\mathrm{E}\Big[\Pi_{j=1}^{\Delta P_i}(1+\nu_0(Q_j))\Big] &= \mathrm{E}_{\Delta P,Q}\Big[e^{\sum_{j=1}^{\Delta P_i} Q_j}\Big] \\
&\overset{\text{iter}}{\underset{\text{exp}}{=}} \mathrm{E}_{\Delta P}\Big[\mathrm{E}_Q\Big[e^{\sum_{j=1}^{\Delta P_i} Q_j}\Big|\Delta P_i\Big]\Big] \\
&\overset{\text{loe}}{=} \mathrm{E}_{\Delta P}\Big[\mathrm{E}_Q\Big[\Pi_{j=1}^{\Delta P_i} e^{Q_j}\Big|\Delta P_i\Big]\Big] \\
&\overset{\text{iid}}{=} \mathrm{E}_{\Delta P}\Big[\Pi_{j=1}^{\Delta P_i}\mathrm{E}_Q\big[e^{Q_j}\big]\Big] \\
&\overset{\text{ltp}}{=} e^{-\lambda_i\Delta t_i}\sum_{k=0}^{\infty}\frac{(\lambda_i\Delta t_i)^k}{k!}\Pi_{j=1}^{k}\mathrm{E}_Q\big[e^{Q}\big] \\
&= e^{-\lambda_i\Delta t_i}\sum_{k=0}^{\infty}\frac{(\lambda_i\Delta t_i)^k}{k!}\mathrm{E}_Q^k\big[e^{Q}\big] \\
&\overset{\text{loe}}{=} e^{-\lambda_i\Delta t_i}\sum_{k=0}^{\infty}\frac{(\lambda_i\Delta t_i\mathrm{E}_Q[e^{Q}])^k}{k!} \\
&= e^{-\lambda_i\Delta t_i}e^{\lambda_i\Delta t_i\mathrm{E}_Q[e^{Q}]} \\
&= e^{\lambda_i\Delta t_i\mathrm{E}_Q[\nu_0(Q)]}. \qquad \square
\end{aligned}$$

An immediate consequence of this result is the following corollary.

Corollary 6.1. Discrete State Expectations.

$$\mathrm{E}[X_{i+1}|X_i] \simeq X_i \exp((\mu_i + \lambda_i \mathrm{E}_Q[\nu_0(Q)])\Delta t_i) \quad (6.20)$$

is the single-step conditional expectation and

$$\mathrm{E}[X_{i+1}] \simeq x_0 \exp\left(\sum_{j=0}^{i} (\mu_j + \lambda_j \mathrm{E}_Q[\nu_0(Q)])\Delta t_j\right) \quad (6.21)$$

is the unconditional or total expectation.

Further, as Δt_i and $\delta t_n \to 0^+$, the continuous form of the expectation follows and is given later in a corollary using other justification.

Example 6.1. Linear, Time-Independent, Constant-Rate Coefficient Case.

In the linear, time-independent, constant-rate coefficient case with $\mu(t)=\mu_0$, $\sigma(t)=\sigma_0$, $\lambda(t)=\lambda_0$ and $\nu(t,Q)=\nu_0(Q)=e^Q-1$,

$$X(t)=x_0\exp\Big((\mu_0-\sigma_0^2/2)(t-t_0) \\ +\sigma_0(W(t)-W(t_0)) \\ +\textstyle\sum_{k=1}^{P(t;Q)-P(t_0;Q)} Q_k\Big), \tag{6.22}$$

where the Poisson counting sum form is now more manageable since the marks do not depend on the prejump-times T_k^-.

Using the independence of the three underlying stochastic processes, $W(t-t_0)=W(t)-W(t_0)$, $P(t-t_0)=P(t;Q)-P(t_0;Q)$ and Q_k , as well as the stationarity of the first two and the law of exponents to separate exponents, leads to partial reduction of the expected state process:

$$\begin{aligned}
\mathrm{E}[X(t)] &\stackrel{\text{iter}}{\underset{\text{exp}}{=}} x_0 e^{(\mu_0-\sigma_0^2/2)(t-t_0)}\mathrm{E}_W\left[e^{\sigma_0 W(t-t_0)}\right] \\
&\quad\cdot \mathrm{E}_P\left[P(t-t_0)\mathrm{E}_Q\left[e^{\sum_{j=1}^{P(t-t_0)}Q_j}\,\middle|\,P(t-t_0)\right]\right] \\
&\stackrel{\text{ltp}}{=} x_0 e^{\mu_0(t-t_0)}\sum_{k=0}^{\infty}\mathrm{E}[P(t-t_0)]\mathrm{E}_P\left[e^{\sum_{j=1}^{k}Q_j}\right] \\
&= x_0 e^{\mu_0(t-t_0)} \\
&\quad\cdot e^{-\lambda_0(t-t_0)}\sum_{k=0}^{\infty}\frac{(\lambda_0(t-t_0))^k}{k!}\prod_{i=1}^{k}\mathrm{E}_Q\left[e^Q\right] \\
&\stackrel{\text{loe}}{=} x_0 e^{\mu_0(t-t_0)}e^{-\lambda_0(t-t_0)}\sum_{k=0}^{\infty}\frac{(\lambda_0(t-t_0))^k}{k!}\mathrm{E}_Q^k\left[e^Q\right] \\
&= x_0 e^{\left(\mu_0+\lambda_0\left(\mathrm{E}_Q\left[e^Q\right]-1\right)\right)(t-t_0)},
\end{aligned} \tag{6.23}$$

where $\lambda_0(t-t_0)$ is the Poisson parameter and $\mathrm{E}[\exp(\sigma_0 W(t-t_0))] = \exp(\sigma_0^2(t-t_0)/2)$ was used.

The $\mathcal{Q} = (-\infty, +\infty)$ is taken as the general mark space range with expectation as with

$$\mathbf{E}_Q\left[e^Q\right] = \int_{\mathcal{Q}} e^q \phi_Q(q) dq.$$

Little more useful simplification can be obtained analytically, except for infinite expansions or equivalent special functions, when the mark density $\phi_Q(q)$ is specified. Numerical procedures may be more useful for practical purposes. The state expectation in this distributed mark case (6.23) should be compared with the pure constant linear coefficient case in L4-p51ff (or p. 146 of the textbook).

• Example 6.2. — Linear Mark-JD Simulatorfor Log-Uniformly Distributed Jump-Amplitudes:

{Continuing unfinished part of Lecture 5.}

The linear SDE jump-diffusion simulator MATLAB code **`linjumpdiff03fig1.m`** in Online Appendix C can be converted from the simple discrete jump process to the distributed jump process here. The primary change is the generation of another set of random numbers for the mark process Q, e.g.,

$$Q = a + (b - a) * \mathtt{rand}(1, n + 1)$$

for a set of $n + 1$ uniformly distributed marks on (a, b) so that the jump-amplitudes of $X(t)$ are log-uniformly distributed.

An example is demonstrated in Figure 6.1 for uniformly distributed marks Q on $(a, b) = (-2, +1)$ and time-dependent coefficients $\{\mu_d(t), \sigma_d(t), \lambda(t)\}$. The MATLAB linear mark-jump-diffusion code *linmarkjumpdiff09fig1.m* , vectorization revision of `linmarkjumpdiff06fig1.m` found in Online Appendix C, is a modification of the pure linear jump-diffusion SDE simulator code `linjumpdiff03fig1.m` illustrated in Figure 5.1 – 5.3, modified for for variable coefficients and mark-independent jumps.

The state exponent $Y(t)$ is simulated, and it is best to simulate $Y(t)$ not $X(t)$, with constant time-steps Δt as

$$YS(i+1) = YS(i) + (\mu(i) - \sigma^2(i)/2) * \Delta t$$
$$+ \sigma(i) * DW(i) + Q(i) * DP(i)$$

with $t(i+1) = t0 + i * \Delta t$ for $i = 1 : n+1$ with $n = 1000$, $t0 = 0$, $0 \le t(i) \le T = 2$ where $Q(i) = \ln(1 + \nu(i))$ and $X(i) = x0 * \exp(Y(i))$.

The mean state is calculated by exponent

$$YM(i+1) = YM(i) + (\mu(i) + \lambda(i) * \overline{\nu}) * \Delta t$$

where $\overline{\nu} \equiv \mathrm{E}[Q]$ and $XM(i) = x0 * \exp(YM(i))$. The `cumsum` can be used to handle the above $i \longrightarrow i+1$ recursions so primarily vector code can be obtained.

The incremental Poisson jump term $\Delta P(t_i) = P(t_i + \Delta t) - P(t_i)$ is simulated by a binomial RNG **`binornd`** for count **`n=1`** and time-dependent vector parameter $\mathbf{\Lambda}$, i.e., the Bernoulli 0-1 process. In the older code used a uniform random number generator on $(0, 1)$ using the **acceptance-rejection technique** (see p. 270, text) to implement the zero-one jump law to obtain the probability of $\lambda(i)\Delta t$ that a jump is accepted there. The same random state is used to obtain the simulations of uniformly distributed Q on (a, b) conditional on a jump event. This technique is useful in problems for which the Statistics Toolbox does not have an appropriate RNG. For consistent usage, the standard MATLAB **`randn`** should be replaced by the **`normrnd`** normal RNG like **`binornd`** usage.

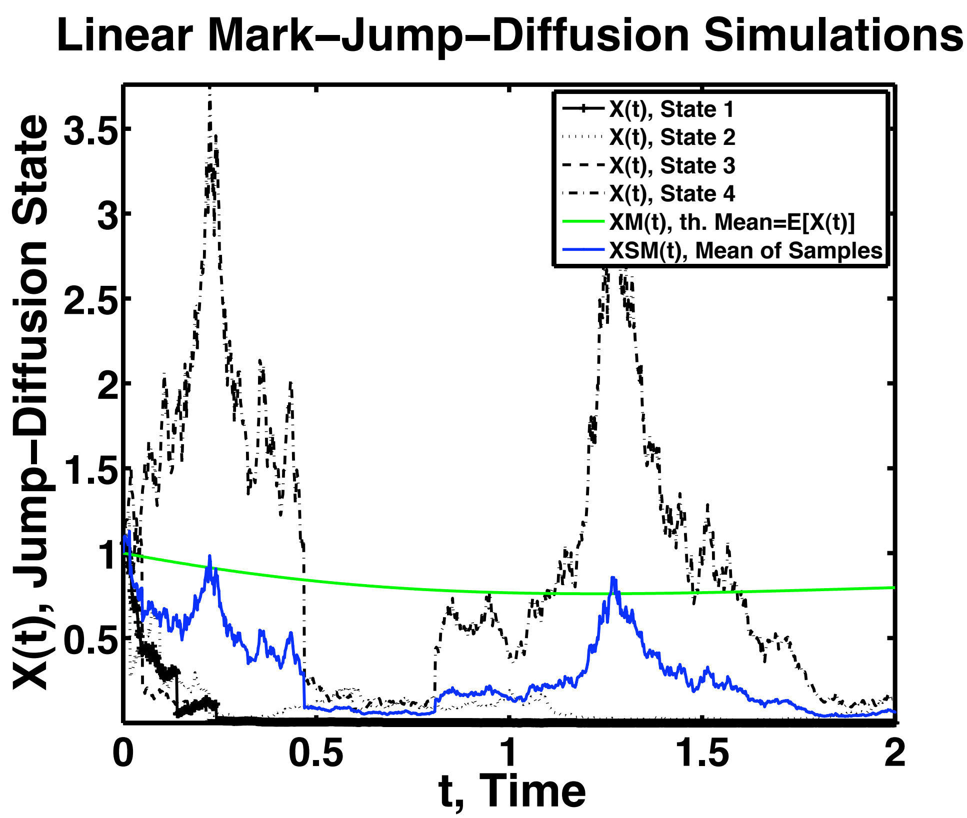

Figure 6.1: Four linear mark-jump-diffusion sample paths for time-dependent coefficients are simulated using MATLAB with $N = 1,000$ time-steps, maximum time $T = 2.0$ and four **randn** and vector **binornd** states. Initially, $x_0 = 1.0$. Parameter values are given in vectorized functions using vector functions and dot-element operations, $\mu_d(t) = 0.1 * \sin(t)$, $\sigma_d(t) = 1.5 * \exp(-0.01 * t)$ and $\lambda = 3.0 * \exp(-t.*t)$. The marks are uniformly distributed on $[-2.0, +1.0]$.

- ***Example 6.3. Linear Mark-Jump-Diffusion Simulations for Variable Coefficients, revised book MATLAB code example (edited)***:

```
function linmarkjumpdiff09fig1
% Revised Linear for Linear Marked-Jump-Diff. 10/09
%   SDE RNG Simulation with variable coefficients
%     for t in [0,T]with sample variation:
%      DX(t) = X(t)*(mu(t)*Dt + sig(t)*DW(t) + nu(Q)*DP(t),
%     X(0) = x0.
%   Or log-state:
%       DY(t)=(mu(t)-sig^2(t)/2)*Dt+sig(t)*DW(t)+Q*DP(t),
%       Y(0) = log(x0) and Q = ln(1+nu(Q)).
%   Generation is by summing Wiener increments DW
%     with Poisson jump increment added .
%   Sufficiently SMALL increments assumed,
%         so zero-one jump law;
%   For demonstration purposes, Q will be assumed to be
%     (qdist =1) UNIFORM on (qparm1,qparm2)=(a,b)
```

```
%     OR
%      (qdist=2) NORMAL  with (qparm1,qparm2)=(muj,sj2).
%   Allows Separate Driver Input and Special Jump
%        or Diffusion Handling.
clc % clear variables,
clf % clear figures
fprintf('\nfunction linmarkjumpdiff09fig1 OutPut:');
%%% Initialize input to jdsimulator with parameters:
N = 1000; t0 = 0; T = 2.0; % Set initial time grid:
idiff = 1; ijump = 1; x0 = 1.0;
qdist = 1;a = -2;b = +1;qparm1 = a;qparm2 = b; %Uniform
%OR E.G., Normal distribution:
%qdist = 2;muj = 0.28;sj2=+0.15;qparm1=muj;qparm2=sj2;
% set constant parameters.
fprintf('\nN=%i; x0=%6.3f;t0=%6.3f;T=%6.3f;',N,x0,t0,T);
fprintf('\nqdist=%i*; qparm1=%6.3f; qparm2=%6.3f;'...
    ,qdist,qparm1,qparm2);
fprintf('\n  * qdist=1 for uniform Q-distribution.');
```

```
fprintf('\n  * qdist=2 for normal Q-distribution.');
%
jdsimulator(idiff,ijump,qdist,qparm1,qparm2 ...
     ,N,x0,t0,T);
%
% END INPUT FOR JUMP-DIFFUSION SIMULATOR.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function jdsimulator(idiff,ijump,qdist,qparm1,qparm2 ...
    ,N,x0,t0,T)
nfig = 0; % initial figure counter.
NI = N + 1;  Dt = (T-t0)/NI;
t = t0:Dt:T; tv = t(1,1:NI); % Compute time vector;
fprintf('\nN=%i; NI=%i; length(t)=%i; ',N,NI,length(t));
kjd = 4 - 2*idiff - ijump;
NP = N + 2; % #plot_points = #time_steps + 1.
muv = mu(tv); % Get time-dependent coefficient vectors
if idiff == 1, sigv = sigma(tv); end
if ijump == 1, lamv = lambda(tv); end
```

```
if qdist==1 % Average nu(Q)=exp(Q)-1, UNIFORM Q-Dist.
    numean=(exp(qparm2)-exp(qparm1))/(qparm2-qparm1)-1;
elseif qdist==2 % Average nu(Q)=exp(Q)-1, NORMAL Q-Dist.
    numean=exp(qparm1-qparm2/2)-1;
end
sqrtDt = sqrt(Dt); % Standard Wiener increment moments.
sigsqrtDt = sqrtDt*sigv;
Lamv = Dt*lamv;
MuDt = Dt*muv;
MuldDt = MuDt - 0.5*sigsqrtDt.^2; % Get Ito correction.
% Begin Sample Path Calculation:
kstates = 4;
XS = zeros(kstates,NI+1); % declare global state vector.
DW = zeros(1,NI);
QDP = zeros(1,NI);
MS = zeros(1,NI+1);
MS(1,2:NI+1) = cumsum(MuldDt);
% Compute Mean State Path:
```

```
MSMDt = MuDt+numean*Lamv; % Mean exponent;
YM = zeros(1,NI+1);
YM(1,2:NI+1) = cumsum(MSMDt);
XM = x0*exp(YM);
for kstate = 1:kstates % Test Simulated Sample Paths:
    if idiff == 1
      randn('state',kstate); % Set initial normal state
      DW = sigsqrtDt.*randn(1,NI);%sig*sqrt(Dt)*dP, NRNG
    end
    if ijump == 1
      if qdist == 1 %Generate Uniform mark vector Q.
        Q = qparm1+(qparm2-qparm1)*rand(1,NI);
      elseif qdist == 2 %Generate Normal mark vector Q.
        sj = sqrt(qparm2); Q = qparm1+sj*randn(1,NI);
      end
      QDP = Q.*binornd(1,Lamv,1,NI); % Q.*DP(t), 0-1.
    end
    WS = zeros(1,NI+1);
```

```
    PS = zeros(1,NI+1);
    WS(1,2:NI+1) = cumsum(DW);% Set Sums
    PS(1,2:NI+1) = cumsum(QDP);
    YS = MS + WS + PS;
    XS(kstate,:) = x0*exp(YS);% Invert exponent to state.
end
fprintf('\nNP=%i;size(t)=[%i,%i];size(XS)=[%i,%i];size(XM)=[%i,
    ,NP,size(t),size(XS),size(XM));
XSM = mean(XS,1);
fprintf('\nNP=%i; size(XSM)=[%i,%i];',NP,size(XM));
% Begin Plot:
scrsize = get(0,'ScreenSize');
ss = 5.2; dss = 0.2; ssmin = 3.0;
nfig = nfig + 1;
stitle = {'Linear Mark-Jump-Diffusion Simulations' ...
        ,'Linear Diffusion Simulations' ...
        ,'Linear Mark-Jump Simulations'};
sylabel = {'X(t), Jump-Diffusion State','X(t) ...
```

```
        ,Diffusion State','X(t), Jump State'};
slegend = {'X(t), State 1  ','X(t), State 2  ' ...
        ,'X(t), State 3  ','X(t), State 4  ' ...
        ,'XM(t), th. Mean=E[X(t)]  '...
        ,'XSM(t), Mean of Samples '};
fprintf('\n\nFigure(%i):  Linear Jump-Diffusion Sims\n' ...
        ,nfig)
figure(nfig)
plot(t,XS(1,1:NP),'k+-' ...
    ,t,XS(2,1:NP),'k:' ...
    ,t,XS(3,1:NP),'k--' ...
    ,t,XS(4,1:NP),'k-.' ...
    ,t,XM(1:NP),'g-' ...
    ,t,XSM(1:NP),'b.-' ...
    ,'LineWidth',2); % Add for more States?
axis tight;
title(stitle(kjd),'FontWeight','Bold','Fontsize',32);
ylabel(sylabel(kjd),'FontWeight','Bold','Fontsize',32);
```

```
xlabel('t, Time','FontWeight','Bold','Fontsize',32);
hlegend=legend(slegend,'Location','NorthEast');
set(hlegend,'Fontsize',16,'FontWeight','Bold');
set(gca,'Fontsize',28,'FontWeight','Bold','linewidth',3);
ss = max(ss - dss,ssmin);
set(gcf,'Color','White','Position' ...
   ,[scrsize(3)/ss 60 scrsize(3)*0.60 scrsize(4)*0.80]);
%
% End JDSimulator Code
%
% linear Time-Dependent SDE Coefficient Functions:
% (Change with application; fns. must be vectorizable,
%  using vector element dot operations or vector fns.)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function M = mu(t)
% drift coefficient example, change with applications:
M = 0.1*sin(t);
% end mu(t)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function S = sigma(t)
% drift coefficient example, change with applications:
S = 1.5*exp(-0.01*t);
% end sigma(t)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function L = lambda(t)
% drift coefficient example, change with applications:
L = 3.0*exp(-t.*t);
% end lambda(t)
% End Variable Coefficients Subfunction Code
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% End function linmarkjumpdiff09fig1.m
```

• Exponential Expectations:

Sometimes it is necessary to get the expectation of an exponential of the integral of a jump-diffusion process. The procedure is much more complicated for distributed amplitude Poisson jump processes than for diffusions since the mark-time process is a product process, i.e., the product of the mark process and the Poisson process. For the time-independent coefficient case, as in a prior example, the exponential processes are easily separable by the law of exponents. However, for the time-dependent case, it is necessary to return to using the space-time process $\mathcal{P}$ and the decomposition approximation used in the mean square limit. The h in the following theorem might be the amplitude coefficient in (6.14) or $h(s,q) = q = \ln(1+\nu(s,q))$.

Theorem 6.2. Expectation for the Exponential of Space-Time Counting Integrals.

Assuming finite second order moments for $h(t,q)$ *and convergence in the mean square limit,*

$$\begin{aligned} &\mathrm{E}\left[\exp\left(\int_{t_0}^{t}\int_{\mathcal{Q}} h(s,q)\mathcal{P}(\mathrm{ds},\mathrm{dq})\right)\right] \\ &\quad = \exp\left(\int_{t_0}^{t}\int_{\mathcal{Q}}\left(e^{h(s,q)}-1\right)\phi_Q(q,s)dq\lambda(s)ds\right) \\ &\quad \equiv \exp\left(\int_{t_0}^{t}\overline{(e^{h}-1)}(s)\lambda(s)ds\right), \end{aligned} \tag{6.24}$$

where $\overline{(e^h-1)}(s) \equiv \mathrm{E}_Q[\exp(h(s,Q))-1]$.

Proof: Let the proper partition of the mark space over disjoint subsets be

$$\mathcal{Q}_m = \{\Delta\mathcal{Q}_j \;\; for \;\; j=1:m \,|\, \cup_{j=1}^{m} \Delta\mathcal{Q}_j = \mathcal{Q}\}.$$

Since Poisson measure is Poisson distributed,

$$\Phi_{\mathcal{P}_j}(k) = \mathrm{Prob}[\mathcal{P}(\mathrm{dt}, \Delta\mathcal{Q}_j) = k] = e^{-\overline{\mathcal{P}_j}} \frac{(\overline{\mathcal{P}_j})^k}{k!}$$

with Poisson parameter

$$\overline{\mathcal{P}_j} \equiv \mathrm{E}[\mathcal{P}(\mathrm{dt}, \Delta\mathcal{Q}_j)] = \lambda(t) dt \Phi_Q(\Delta\mathcal{Q}_j, t_i)$$

for each subset $\{\Delta\mathcal{Q}_j\}$.

Similarly, let the proper partition over the time interval be

$$\mathcal{T}_n = \{t_i | t_{i+1} = t_i + \Delta t_i \; for \; i=0:n, t_0=0, \\ t_{n+1}=t, \max_i[\Delta t_i] \to 0 \; as \; n \to +\infty\}.$$

The disjoint property over subsets and time intervals means $\mathcal{P}([t_i, t_i + \Delta t_i), \Delta\mathcal{Q}_j)$ and $\mathcal{P}([t_i, t_i + \Delta t_i), \Delta\mathcal{Q}'_j)$ will be pairwise independent provided $j' \neq j$ for fixed i corresponding to property (5.19) on L5-46 (or 5.15 in textbook) for infinitesimals, while $\mathcal{P}([t_i, t_i + \Delta t_i), \Delta\mathcal{Q}_j)$ and $\mathcal{P}([t_i, t_i + \Delta t'_i), \Delta\mathcal{Q}'_j)$ will be pairwise independent provided $i' \neq i$ and $j' \neq j$, corresponding to similar property (5.16) in textbook, but omitted from lecture 1, for infinitesimals. For brevity, let $h_{i,j} \equiv h(t_i, q^*_j)$, where $q^*_j \in \Delta\mathcal{Q}_j$, $\mathcal{P}_{i,j} \equiv \mathcal{P}_i([t_i, t_i + \Delta t_i), \Delta\mathcal{Q}_j)$ and $\overline{\mathcal{P}}_{i,j} \equiv \lambda_i \Delta t_i \Phi_Q(\Delta\mathcal{Q}_j)$. The IFA expansion and reassembling limits are used , with $\mathcal{P}_{i,j}$ playing the dual roles of the two increments $(\Delta t_i, \Delta\mathcal{Q}_j)$, the law of exponents and the independence denoted by $\underset{\text{inc}}{\overset{\text{ind}}{=}}$, so we have the following:

$$
\begin{aligned}
&\mathbf{E}\left[\exp\left(\int_{t_0}^{t}\int_{\mathcal{Q}} h\mathcal{P}\right)\right] \\
&\overset{\text{ifa}}{=} \lim_{n\to\infty} \mathbf{E}\left[\exp\left(\sum_{i=0}^{n}\sum_{j=1}^{m} h_{i,j}\mathcal{P}_{i,j}\right)\right] \\
&\underset{\text{inc}}{\overset{\text{ind}}{=}} \lim_{n\to\infty} \Pi_{i=0}^{n}\Pi_{j=1}^{m}\mathbf{E}\left[\exp\left(h_{i,j}\mathcal{P}_{i,j}\right)\right] \\
&= \lim_{n\to\infty} \Pi_{i=0}^{n}\Pi_{j=1}^{m} \exp\left(-\overline{\mathcal{P}}_{i,j}\right)\sum_{k_{i,j}=0}^{\infty}\frac{\overline{\mathcal{P}}_{i,j}^{k_{i,j}}}{k_{i,j}!}\exp\left(h_{i,j}k_{i,j}\right) \\
&= \lim_{n\to\infty} \Pi_{i=0}^{n}\Pi_{j=1}^{m} \exp\left(\overline{\mathcal{P}}_{i,j}\left(\exp(h_{i,j})-1\right)\right) \\
&= \lim_{n\to\infty} \exp\left(\sum_{i=0}^{n}\sum_{j=1}^{m}\left(\exp(h_{i,j})-1\right)\lambda_i\Delta t_i\Phi_Q(\Delta Q_i,t_i)\right) \\
&\overset{\text{ifa}}{=} \exp\left(\int_{t_0}^{t}\int_{\mathcal{Q}}\left(\exp(h(s,q))-1\right)\phi_Q(q,s)dq\lambda(s)ds\right) \\
&\equiv \exp\left(\int_{t_0}^{t}\overline{\left(\exp((h-1)(s))\right)}\lambda(s)ds\right).
\end{aligned}
$$

Thus, the main technique is to unassemble the IFA limit discrete approximation to get at the independent random part, take its expectation and then reassemble the IFA limit, justifying the interchange of expectation and exponent-integration. $\square$

Remarks 6.1:

- *Note that the mark space subset* $\Delta\mathcal{Q}_j$ *is never used directly as a discrete element of integration, since the subset would be infinite if the mark space were infinite. The mark space element is used only through the distribution which would be bounded. This is quite unlike the time domain, where we can select* t *to be finite. If the mark space were finite, say,* $\mathcal{Q} = [a, b]$*, then a concrete partition of* $[a, b]$ *similar to the time-partition can be used.*

- *Also note that the dependence on* $(X(t), t)$ *was not used, but could be considered suppressed but absorbed into the existing* t *dependence of* h *and* $\mathcal{P}$*.*

Corollary 6.2. Expectation of X(t) for Linear SDE:

Let $X(t)$ *be the solution* (6.16) *with* $\overline{\nu}(t) \equiv \mathrm{E}[\nu(t,Q)]$ *of* (6.13). *Then*

$$\begin{aligned} \mathrm{E}[X(t)] &= x_0 \exp\left(\int_{t_0}^{t} (\mu_d(s) + \lambda(s)\overline{\nu}(s))\, ds\right) \\ &= x_0 \exp\left(\int_{t_0}^{t} \mathrm{E}[dX(s)/X(s)] ds\right). \end{aligned} \tag{6.25}$$

Proof: The jump part, i.e., the main part, follows from exponential Theorem 6.2, (6.24) and the lesser part for the diffusion is left as an exercise for the reader.

However, note that the exponent is the time integral of $\mathrm{E}[dX(t)/X(t)]$, the relative conditional infinitesimal mean, which is independent of $X(s)$ and is valid only for the linear mark-jump-diffusion SDE. □

Remark 6.2:

The relationship in (6.25) *is a* ***quasi-deterministic equivalence*** *for linear mark-jump-diffusion SDEs and was shown by Hanson and Ryan (MB 1989) . They also produced a nonlinear jump counterexample that has a formal closed-form solution in terms of the gamma function, for which the result does not hold and a very similar example is given in Exercise 9 in Chapter 4 of the textbook.*

• Moments of Log-Jump-Diffusion Process:

For the log-jump-diffusion process $dY(t)$ in (6.14), suppose that the jump-amplitude is time-independent and that the mark variable was conveniently chosen as

$$Q = \ln(1 + \nu(t, Q))$$

so that the SDE has the form

$$dY(t) \underset{\text{zol}}{\overset{\text{dt}}{=}} \mu_{\text{ld}}(t)dt + \sigma(t)dW(t) + QdP(t;Q), \quad (6.26)$$

where $\mu_{\text{ld}}(t) \equiv \mu(t) - \sigma^2(t)/2$, and in the case of applications for which the time-step Δt is an increment that is not infinitesimal like dt, there is some probability of more than one jump,

$$\Delta Y(t) = \mu_{\text{ld}}(t)\Delta t + \sigma(t)\Delta W(t) + \sum_{k=P(t;Q)+1}^{P(t;Q)+\Delta P(t;Q)} Q_k. \quad (6.27)$$

The results for the infinitesimal case (6.26) are contained in the incremental case (6.27).

The first few moments can be found in general for (6.27), and if up to the fourth moment, then the skew and kurtosis coefficients can be calculated. These calculations can be expedited by the following lemma, concerning sums of zero-mean IID random variables.

Lemma 6.2. Zero-Mean IID Random Variable Sums:

Let $\{X_i|i=1:n\}$ *be a set of zero-mean IID random variables, i.e.,* $\mathrm{E}[X_i]=0$. *Let* $M^{(m)}\equiv E[X_i^m]$ *be the* m*th moment and*

$$S_n^{(m)}\equiv\sum_{i=1}^{n}X_i^m$$

with $S_n^{(1)}=S_n$ *the usual partial sum over the set and*

$$\mathrm{E}[S_n^{(m)}]=nM^{(m)}; \qquad (6.28)$$

then the expectation of powers of S_n *for* $m=1:4$ *is*

$$\mathrm{E}\left[(S_n)^m\right]=\left\{\begin{array}{ll} 0, & m=1 \\ nM^{(2)}, & m=2 \\ nM^{(3)}, & m=3 \\ nM^{(4)}+3n(n-1)\left(M^{(2)}\right)^2, & m=4 \end{array}\right\}. \qquad (6.29)$$

Proof: The proof is done first by the linear property of the expectation and the IID properties of the X_i,

$$\mathrm{E}\left[S_n^{(m)}\right] = \sum_{i=1}^{n} \mathrm{E}[X_i^m] = \sum_{i=1}^{n} M^{(m)} = nM^{(m)}. \quad (6.30)$$

The $m = 1$ case is trivial due to the zero-mean property of the X_i's and the linearity of the expectation operator, $\mathrm{E}[S_n] = \sum_{i=1}^{n} \mathrm{E}[X_i] = 0$.

For $m = 2$, the induction hypothesis from (6.29) is

$$\mathrm{E}\left[S_n^2\right] \equiv \mathrm{E}\left[\left(\sum_{i=1}^{n} X_i^2\right)\right] = nM^{(2)},$$

where the initial condition at $n = 1$ is $\mathrm{E}[S_1^2] = \mathrm{E}[X_1^2] = M^{(2)}$ by definition.

The hypothesis can be proved easily by partial sum recursion $S_{n+1} = S_n + X_{n+1}$, application of the binomial theorem, expectation linearity and the zero-mean IID property:

$$\mathrm{E}\left[S_{n+1}^2\right] = \mathrm{E}\left[(S_n + X_{n+1})^2\right] = \mathrm{E}\left[S_n^2 + 2X_{n+1}S_n + X_{n+1}^2\right]$$
$$= nM^{(2)} + 2\cdot 0\cdot 0 + M^{(2)} = (n+1)M^{(2)}. \quad (6.31)$$

QED for $m = 2$.

This is similar for the power $m = 3$, again beginning with the induction hypothesis

$$\mathrm{E}\left[S_n^3\right] \equiv \mathrm{E}\left[\left(\sum_{i=1}^{n} X_i\right)^3\right] = nM^{(3)}.$$

where the initial condition at $n = 1$ is $\mathrm{E}[S_1^3] = \mathrm{E}[X_1^3] = M^{(3)}$ by definition.

Using the same techniques as in (6.31),

$$\begin{aligned}\mathbf{E}\left[S_{n+1}^3\right] &= \mathbf{E}\left[(S_n+X_{n+1})^3\right] \\ &= \mathbf{E}\left[S_n^3+3X_{n+1}S_n^2+3X_{n+1}^2S_n^2+X_{n+1}^3\right] \\ &= nM^{(3)}+3\cdot 0\cdot nM^{(2)}+3\cdot M^{(2)}\cdot 0+M^{(3)} \\ &= (n+1)M^{(3)}. \qquad (6.32)\end{aligned}$$

QED for $m = 3$.

Finally, the case for the power $m = 4$ is a little different since an additional nontrivial term arises from the product of the squares of two independent variables.

The induction hypothesis for $m = 4$ is

$$\mathbf{E}\left[S_n^4\right] \equiv \mathbf{E}\left[\left(\sum_{i=1}^{n} X_i\right)^4\right] = nM^{(4)} + 3n(n-1)(M^{(2)})^2,$$

where the initial condition at $n = 1$ is $\mathbf{E}[S_1^4] = \mathbf{E}[X_1^4] = M^{(4)}$ by definition. Using the same techniques as in (6.31),

$$\begin{aligned}
\mathbf{E}\left[S_{n+1}^4\right] &= \mathbf{E}\left[(S_n + X_{n+1})^4\right] \\
&= \mathbf{E}\left[S_n^4 + 4X_{n+1}S_n^3 + 6X_{n+1}^2 S_n^2 + 4X_{n+1}^3 S_n^1 + X_{n+1}^4\right] \\
&= nM^{(4)} + 3n(n-1)(M^{(2)})^2 + 4 \cdot 0 \cdot nM^{(3)} \\
&\quad + 6 \cdot M^{(2)} \cdot nM(2) + 4 \cdot M^{(3)} \cdot 0 + M^{(4)} \\
&= (n+1)M^{(4)} + 3n(n+1)(M^{(2)})^2. \qquad (6.33)
\end{aligned}$$

QED for $m = 4$. □

Remark 6.3: *The results here depend on the IID and zero-mean properties, but do not otherwise depend on the particular distribution of the random variables. The results are used in the following theorem.*

Theorem 6.3. Some Moments of the Log-jump-Diffusion (LJD) Process $\Delta Y(t)$:

Let $\Delta Y(t)$ satisfy the stochastic difference equation (6.27) *and let the marks Q_k be IID with mean $\mu_j \equiv \mathrm{E}_Q[Q_k]$ and variance $\sigma_j^2 \equiv \mathrm{Var}_Q[Q_k]$. Then the first four moments, $m = 1:4$, are*

$$\mu_{\mathrm{ljd}}(t) \equiv \mathrm{E}[\Delta Y(t)] = (\mu_{\mathrm{ld}}(t) + \lambda(t)\mu_j)\Delta t; \qquad (6.34)$$

$$\sigma_{\mathrm{ljd}}(t) \equiv \mathrm{Var}[\Delta Y(t)] = \left(\sigma_d^2(t) + \left(\sigma_j^2 + \mu_j^2\right)\lambda(t)\right)\Delta t; \qquad (6.35)$$

$$M^{(3)}_{\mathrm{ljd}}(t) \equiv \mathrm{E}\left[(\Delta Y(t) - \mathrm{E}[\Delta Y(t)])^3\right]$$
$$= \left(M^{(3)}_j + \mu_j\left(3\sigma^2_j + \mu^2_j\right)\right)\lambda(t)\Delta t, \qquad (6.36)$$

where $M^{(3)}_j \equiv \mathrm{E}_Q[(Q_i - \mu_j)^3]$;

$$M^{(4)}_{\mathrm{ljd}}(t) \equiv \mathrm{E}\left[(\Delta Y(t) - \mathrm{E}[\Delta Y(t)])^4\right]$$
$$= \left(M^{(4)}_j + 4\mu_j M^{(3)}_j + 6\mu^2_j\sigma^2_j + \mu^4_j\right)\lambda(t)\Delta t$$
$$+3\left(\sigma^2_d(t) + \left(\sigma^2_j + \mu^2_j\right)\lambda(t)\right)^2(\Delta t)^2, \qquad (6.37)$$

where $M^{(4)}_j \equiv \mathrm{E}_Q[(Q_i - \mu_j)^4]$.

Proof: One general technique for calculating moments of the log-jump-diffusion process is **iterated expectations**. Thus, writing $\Delta P(t;Q) = \Delta P(t)$ to suppress Q as the symbolic generation parameter in ΔP when performing the iterated expectations,

$$\begin{aligned}
\mu_{\mathrm{ljd}}(t) = \mathrm{E}[\Delta Y(t)] &= \mu_{\mathrm{ld}}(t)\Delta t + \sigma_d(t)\cdot 0 \\
&\quad + \mathrm{E}_{\Delta P(t)}\left[\mathrm{E}_Q\left[\sum_{i=1}^{\Delta P(t)} Q_i \middle| \Delta P(t)\right]\right] \\
&= \mu_{\mathrm{ld}}(t)\Delta t + \mathrm{E}_{\Delta P(t)}\left[\sum_{i=1}^{\Delta P(t)} \mathrm{E}_Q[Q_i]\right] \\
&\stackrel{\mathrm{iid}}{=} \mu_{\mathrm{ld}}(t)\Delta t + \mathrm{E}_{\Delta P(t)}[\Delta P(t)\mathrm{E}_Q[Q]] \\
&= (\mu_{\mathrm{ld}}(t) + \mu_j\lambda(t))\,\Delta t,
\end{aligned}$$

proving the first moment formula, using the increment jump-count.

For the higher moments, the main key technique for efficient calculation of the moments is decomposing the log-jump-diffusion process deviation into zero-mean deviation factors, i.e.,

$$\begin{aligned}\Delta Y(t)-\mu_{\text{ljd}}(t) &= \sigma_d(t)\Delta W(t)\\ &+\sum_{i=1}^{\Delta P(t)}(Q_i-\mu_j)\\ &+\mu_j(\Delta P(t)-\lambda(t)\Delta t),\end{aligned}$$

where $\mu_j\Delta P(t)=\sum_{i=1}^{\Delta P(t)}\mu_j$ was used. In addition, the multiple applications of the binomial theorem and the convenient increment power Table 1.1 for $\Delta W(t)$ and Table 1.2 for $\Delta P(t)$ are used.

The incremental process variance is found by

$$
\begin{aligned}
\sigma_{\rm ljd}(t) &\equiv \mathrm{Var}[\Delta Y(t)] \\
&= \mathrm{E}\Big[\Big(\sigma_d(t)\Delta W(t) + \textstyle\sum_{i=1}^{\Delta P(t)}(Q_i-\mu_j) \\
&\quad +\mu_j(\Delta P(t)-\lambda(t)\Delta t)\Big)^2\Big] \\
&= \sigma_d^2(t)\mathrm{E}_{\Delta W(t)}[(\Delta W)^2(t)] + 2\sigma_d\cdot 0 \\
&\quad +\mathrm{E}\left[\left(\textstyle\sum_{i=1}^{\Delta P(t)}(Q_i-\mu_j)+\mu_j(\Delta P(t)-\lambda(t)\Delta t)\right)^2\right] \\
&= \sigma_d^2(t)\Delta t + \mathrm{E}_{\Delta P(t)}\Big[\textstyle\sum_{i=1}^{\Delta P(t)}\sum_{k=1}^{\Delta P(t)} \\
&\quad \cdot\mathrm{E}_Q[(Q_i-\mu_j)(Q_k-\mu_j)] \\
&\quad +2\mu_j(\Delta P(t)-\lambda(t)\Delta t)\textstyle\sum_{i=1}^{\Delta P(t)}\mathrm{E}_Q[(Q_i-\mu_j)] \\
&\quad +\mu_j^2(\Delta P(t)-\lambda(t)\Delta t)^2\Big] \\
&= \sigma_d^2(t)\Delta t + \mathrm{E}_{\Delta P(t)}\Big[\Delta P(t)\sigma_j^2 + 0 \\
&\quad +\mu_j^2(\Delta P(t)-\lambda(t)\Delta t)^2\Big] \\
&= \left(\sigma_d^2(t)+\left(\sigma_j^2+\mu_j^2\right)\lambda(t)\right)\Delta t.
\end{aligned}
$$

The cases of the third and fourth central moments are i similarly calculated, however, they will be omitted for simplicity, but can be found in the textbook sample Chapter 5 found online at

http://www.siam.org/books/dc13/DC13samplechpt.pdf .

SIAM Books currently has a sale on the textbook, code DC13, for non-memberss at $72.80 until 31 December 2009 (the special coupon should have expired) at

http://www.siam.org/catalog/fb09.php

by searching the page for DC13, click on the title and set the pull down menu to *FB09 Sale Price – $72.80*. Sorry, this sale is not available outside North America at www.cambridge.org/siam.

Remarks 6.4:

- Recall that the third and fourth moments are measures of skewness and peakedness (kurtosis), respectively. The normalized representations in the current notation are the coefficient of skewness,

$$\eta_3[\Delta Y(t)] \equiv M^{(3)}_{\mathrm{ljd}}(t)/\sigma^3_{\mathrm{ljd}}(t), \qquad (6.38)$$

from (B.11), and the coefficient of kurtosis from (B.12),

$$\eta_4[\Delta Y(t)] \equiv M^{(4)}_{\mathrm{ljd}}(t)/\sigma^4_{\mathrm{ljd}}(t). \qquad (6.39)$$

- For example, if the marks are normally or uniformly distributed, then

$$M^{(3)}_j = 0,$$

since the normal and uniform distributions are both symmetric about the mean, so they lack skew.

Thus, we have the coefficient of skew

$$\eta_3[\Delta Y(t)] = \frac{\mu_j\left(3\sigma_j^2+\mu_j^2\right)\lambda(t)\Delta t}{\sigma_{\rm ljd}^3(t)}$$
$$= \frac{\mu_j\left(3\sigma_j^2+\mu_j^2\right)\lambda(t)}{\left(\sigma_d^2(t)+\left(\sigma_j^2+\mu_j^2\right)\lambda(t)\right)^3(\Delta t)^2},$$

using $\sigma_{\rm ljd}(t)$ given by (6.35). For the uniform distribution, the mean μ_j is given by (B.15) explicitly in terms of the uniform interval $[a, b]$ and the variance σ_j^2 by (B.16), while for the normal distribution, μ_j and σ_j^2 are the given normal model parameters. In general, the normal and unform distribution versions of the log-jump-diffusion process $X(t)$ will have skew, although the component incremental diffusion and mark processes are skewless.

In the normal and uniform mark cases, the fourth moment of the jump marks are

$$M_j^{(4)}/\sigma_j^4 = \left\{ \begin{array}{ll} 3, & normal\ Q_i \\ 1.8, & uniform\ Q_i \end{array} \right\},$$

which are in fact the coefficients of kurtosis for the normal and uniform distributions, respectively, so

$$\eta_4[\Delta Y(t)] = \left(\left\{ \begin{array}{ll} 3, & normal\ Q_i \\ 1.8, & uniform\ Q_i \end{array} \right\} \sigma_j^4 + 6\mu_j^2\sigma_j^2 + \mu_j^4 \right) \frac{\lambda(t)\Delta t}{\sigma_{\text{ljd}}^4(t)}$$

$$+3\left(\sigma_d^2(t) + \left(\sigma_j^2 + \mu_j^2\right)\lambda(t)\right)^2 \frac{(\Delta t)^2}{\sigma_{\text{ljd}}^4(t)}.$$

- The moment formulas for the differential log-jump-diffusion process $dY(t)$ follow immediately from Theorem 6.2 (5.17 in textbook) by dropping terms $\mathrm{O}((\Delta t)^2)$ and replacing Δt by dt.

Summary of Lecture 6?

1.
2.
3.
4.
5.
6.
7.
8.