On the Weak Approximation of Jump-Diffusion Processes with Applications in Finance

Nicola Bruti-Liberati
School of Finance & Economics,
University of Technology, Sydney

December 16, 2005

Joint work with Eckhard Platen

QMF05

Outline

- **jump-diffusion stochastic differential equations in finance**

- **strong and weak numerical schemes**

- **weak numerical schemes**
  - Taylor and jump-adapted weak schemes
  - derivative free, implicit, predictor-corrector schemes
  - numerical results

Literature on Jumps in Finance

- Merton (1976) $\Rightarrow$ Merton model
- Madan & Seneta (1990), Eberlein & Keller (1995) $\Rightarrow$ Lévy
- Bates (1996) $\Rightarrow$ stochastic volatility
- Björk, Kabanov & Runggaldier (1997) $\Rightarrow$ term structure
- Cont & Tankov (2003) $\Rightarrow$ jumps in finance
- Glasserman (2004) $\Rightarrow$ Monte Carlo
- Geman & Roncoroni (2005) $\Rightarrow$ electricity prices

Problem Setting

- **SDE**:

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t + c(t-, X_{t-})\, dJ_t$$

- $J_t = N_t$: Poisson process, intensity $\lambda < \infty$

- $J_t = \sum_{i=1}^{N_t}(\xi_i - 1)$: compound Poisson, ξ_i i.i.d r.v.

- Poisson random measure

- $\{(\tau_i, \xi_i), i = 1, 2, \ldots, N_T\}$

Examples

- **Merton Model**

$$dX_t = X_{t-}\left(\mu dt + \sigma dW_t + dJ_t\right),$$

$$\Downarrow$$

$$X_t = X_0\, e^{(\mu - \frac{1}{2}\sigma^2)t + \sigma W_t} \prod_{i=1}^{N_t} \xi_i$$

- **Bates Model**

- **Credit Risk Models**

Numerical Approximations

- time discretization

$$t_n = n\Delta$$

- discrete time approximation

$$Y_{n+1} = Y_n + a(Y_n)\Delta + b(Y_n)\Delta W_n + c(Y_n)\Delta J_n$$

- strong schemes $\Longrightarrow$ pathwise approximations

- weak schemes $\Longrightarrow$ probability approximations

Strong Convergence

- **Applications:** scenario analysis, filtering and hedge simulation
- **Strong Convergence:**

$$\lim_{\Delta \to 0} E(|X_T - Y_N^{\Delta}|^2) = 0$$

- **Order:** $Y^{\Delta} \longrightarrow X$ with **strong order** γ if

$$\varepsilon_s(\Delta) = \sqrt{E(|X_T - Y_N^{\Delta}|^2)} \leq K \Delta^{\gamma}$$

Weak Convergence

- **Applications:** derivative pricing and evaluation of risk measures
- **Weak Convergence:**

$$\lim_{\Delta \to 0} |E(g(X_T)) - E(g(Y_N^\Delta))| = 0$$

example: $g(x) = x^q, \quad g(x) = (x-K)^+, \quad g(x) = I_{\{x \leq c\}}$

- **Order:** $Y^\Delta \longrightarrow X$ with **weak order** β if

$$\varepsilon_w(\Delta) = |E(g(X_T)) - E(g(Y_N^\Delta))| \leq K\Delta^\beta$$

Example on a Diffusion

- **geometric Brownian motion**

$$dX_t = r\, X_t dt + \sigma\, X_t dW_t$$

- **Euler scheme** $(\gamma = 0.5,\ \beta = 1)$

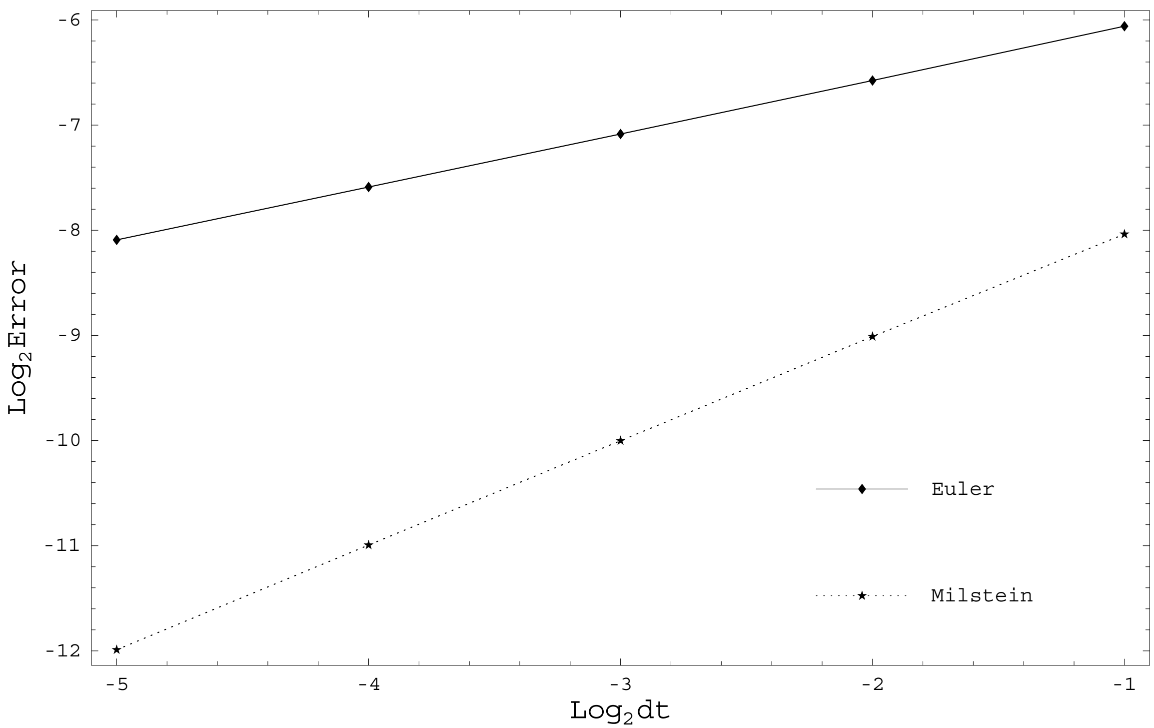
$$Y_{n+1} = Y_n + r\, Y_n \Delta + \sigma\, Y_n \Delta W_n$$

- **Milstein scheme** $(\gamma = 1,\ \beta = 1)$

$$Y_{n+1} = Y_n + r\, Y_n \Delta + \sigma\, Y_n \Delta W_n + \frac{\sigma^2}{2} Y_n \big((\Delta W_n)^2 - \Delta \big)$$

Strong Error

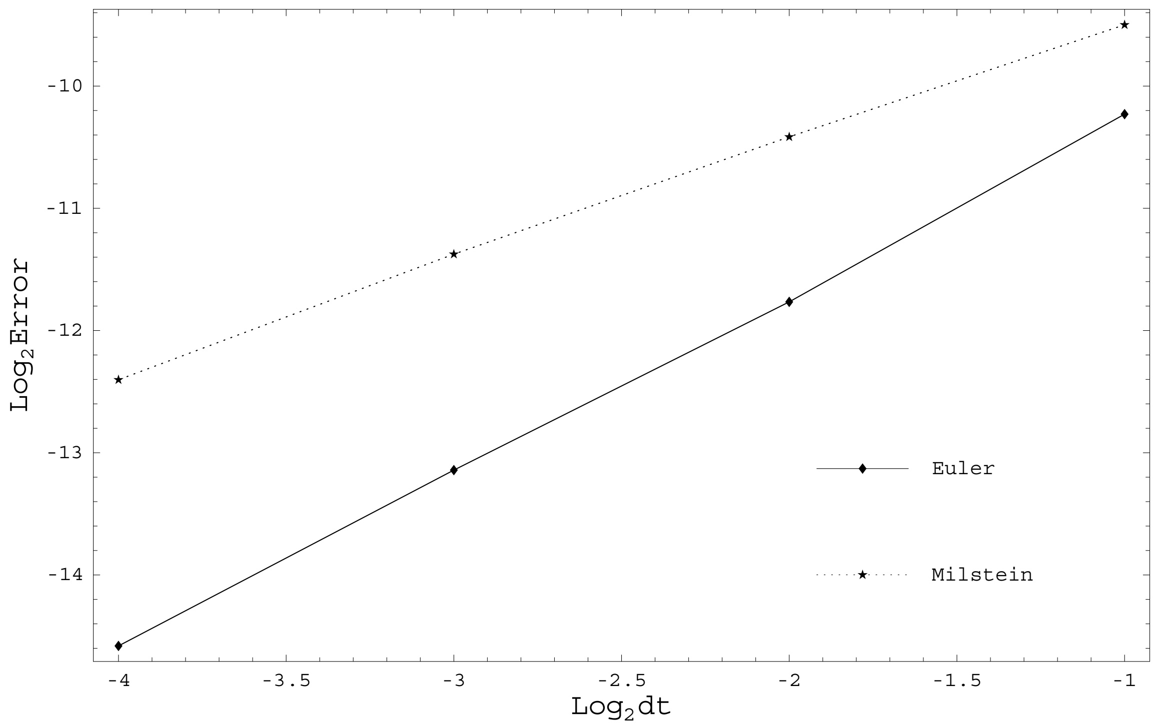
geometric brownian motion : $r = 0.05$, $\sigma = 0.2$, $X_0 = 1$, $T = 0.5$



Log-log plot of the strong error.

Weak Error

call payoff : $r = 0.05$, $\sigma = 0.2$, $X_0 = 1$, $T = 0.5$, $\mathrm{strike} = 1$



Log-log plot of the weak error.

Literature on Weak Schemes

- Mikulevicius $\&$ Platen (1991) $\Rightarrow$ jump-adapted order $\beta \in \{1, 2 \ldots\}$ weak schemes

- Liu $\&$ Li (2000) $\Rightarrow$ order $\beta \in \{1, 2 \ldots\}$ weak Taylor, extrapolation and simplified schemes

- Glassermann $\&$ Merener (2000) and Kubilius $\&$ Platen (2002) $\Rightarrow$ jump-adapted schemes with weaker assumptions on coefficients

- Bruti-Liberati $\&$ Platen (2005) $\Rightarrow$ jump-adapted order $\beta \in \{1, 2 \ldots\}$ derivative free, implicit and predictor-corrector schemes

Weak Approximations

- compute the functional

$$E(g(X_T^{t,x})) := u(t,x)$$

- $u(t,x)$ satisfies the PIDE

$$\begin{cases} \frac{\partial u}{\partial t} + \mathcal{L}\, u = 0 \\ u(T,x) = g(x) \end{cases}$$

with

$$\mathcal{L} = \frac{b(x)^2}{2}\frac{\partial^2 u}{\partial x^2} + a(x)\frac{\partial u}{\partial x} + \int_{\mathcal{E}} \Big(u(t, x + c(x,v)) - u(t,x) \Big) \phi(dv)$$

Euler Scheme

- **SDE**

$$dX_t = a(X_t)dt + b(X_t)dW_t + c(X_{t-})\, dN_t$$

- **Euler scheme**

$$Y_{n+1} = Y_n + a(Y_n)\Delta + b(Y_n)\Delta W_n + c(Y_n)\Delta p_n$$

where

$$\Delta W_n \sim \mathcal{N}(0, \Delta) \qquad \text{and} \qquad \Delta p_n = N_{t_{n+1}} - N_{t_n} \sim Poiss(\lambda\Delta)$$

- $\beta = 1.$

Euler Scheme

- **SDE**

$$dX_t = a(X_t)dt + b(X_t)dW_t + c(X_{t-})\,dJ_t$$

- **Euler scheme**

$$Y_{n+1} = Y_n + a(Y_n)\Delta + b(Y_n)\Delta W_n + c(Y_n)\Delta J_n$$

where

$$\Delta W_n \sim \mathcal{N}(0, \Delta) \qquad \text{and} \qquad \Delta J_n = \sum_{i=N_{t_n}+1}^{N_{t_{n+1}}} (\xi_i - 1)$$

- $\beta = 1.$

Order 2.0 Weak Taylor Scheme

$$
\begin{aligned}
Y_{n+1} \quad = \quad & Y_n + a\Delta + b\Delta W_n + c\Delta p_n + \left(a\,a' + \frac{1}{2}a''\,b^2\right) I_{(0,0)} + a'\,b\,I_{(1,0)} \\
& + \left(a\,b' + \frac{1}{2}b''\,b^2\right) I_{(0,1)} + b\,b'\,I_{(1,1)} + b\,c'\,I_{(1,-1)} \\
& + L^{(-1)} b\,I_{(-1,1)} + L^{(-1)} c\,I_{(-1,-1)} \\
& + \left(a\,c' + \frac{1}{2}c''\,b^2\right) I_{(0,-1)} + L^{(-1)} a\,I_{(-1,0)},
\end{aligned}
$$

with $I_{(i,j)}$ multiple stochastic integrals

Order 2.0 Weak Taylor Scheme

$$I_{(1,-1)} = \sum_{i=N(t_n)+1}^{N(t_{n+1})} W_{\tau_i} - \Delta p_n \, W_{t_n}, \quad I_{(-1,1)} = \Delta p_n \, \Delta W_n - I_{(1,-1)}$$

- simulation jump times τ_i : $W_{\tau_i} \Longrightarrow I_{(1,-1)}$ and $I_{(-1,1)}$
- Computational effort heavily dependent on intensity λ

Jump-Adapted Approximations

jump-adapted time discretisation

$$\Downarrow$$

jump times included in time discretisation

- **jump-adapted Euler scheme**

$$Y_{t_{n+1}-} = Y_{t_n} + a(Y_{t_n})\Delta_{t_n} + b(Y_{t_n})\Delta W_{t_n} \tag{1}$$

and

$$Y_{t_{n+1}} = Y_{t_{n+1}-} + c(Y_{t_{n+1}-})\big(J(t_{n+1}) - J(t_{n+1}-)\big) \tag{2}$$

- $\beta = 1$

Jump-Adapted Simplified Schemes

- **jump-adapted simplified Euler scheme**

$$Y_{t_{n+1}-} = Y_{t_n} + a(Y_{t_n})\Delta_{t_n} + b(Y_{t_n})\Delta\widehat{W}_n^2$$

  and

$$Y_{t_{n+1}} = Y_{t_{n+1}-} + c(Y_{t_{n+1}-})\left(J(t_{n+1}) - J(t_{n+1}-)\right)$$

- if $\Delta\widehat{W}_n^2$ match first 3 moments of $\Delta W_n \Longrightarrow \beta = 1$

- $$P(\Delta\widehat{W}_n^2 = \pm\sqrt{\Delta}) = \frac{1}{2}$$

Jump-Adapted Taylor Approximations

- **jump-adapted order 2 weak Taylor scheme**

$$
\begin{aligned}
Y_{t_{n+1}-} \quad = \quad & Y_{t_n} + a\Delta_{t_n} + b\Delta W_{t_n} + \frac{b\,b'}{2}\Big((\Delta W_{t_n})^2 - \Delta_{t_n}\Big) + a'\,b\,\Delta Z_{t_n} \\
& + \frac{1}{2}\left(a\,a' + \frac{1}{2}a''b^2\right)\Delta_{t_n}^2 + \left(a\,b' + \frac{1}{2}b''\,b^2\right)\{\Delta W_{t_n}\,\Delta_{t_n} - \Delta Z_{t_n}\}
\end{aligned}
$$

and

$$
Y_{t_{n+1}} = Y_{t_{n+1}-} + c(Y_{t_{n+1}-})\,\big(J(t_{n+1}) - J(t_{n+1}-)\big)
$$

- $\beta = 2$

Implicit Schemes

- implicit schemes $\Rightarrow$ wide stability regions

- **jump-adapted implicit Euler scheme**

$$Y_{t_{n+1-}} = Y_{t_n} + \Big\{\theta a(Y_{t_{n+1-}}) + (1-\theta)a\Big\}\Delta_{t_n} + b\Delta W_{t_n}$$

  with $\theta \in [0,1]$

- $\beta = 1$

- additional algebraic equation

Predictor-Corrector Schemes

- predictor-corrector $\Rightarrow$ stability and efficiency

- **jump-adapted predictor-corrector Euler scheme**

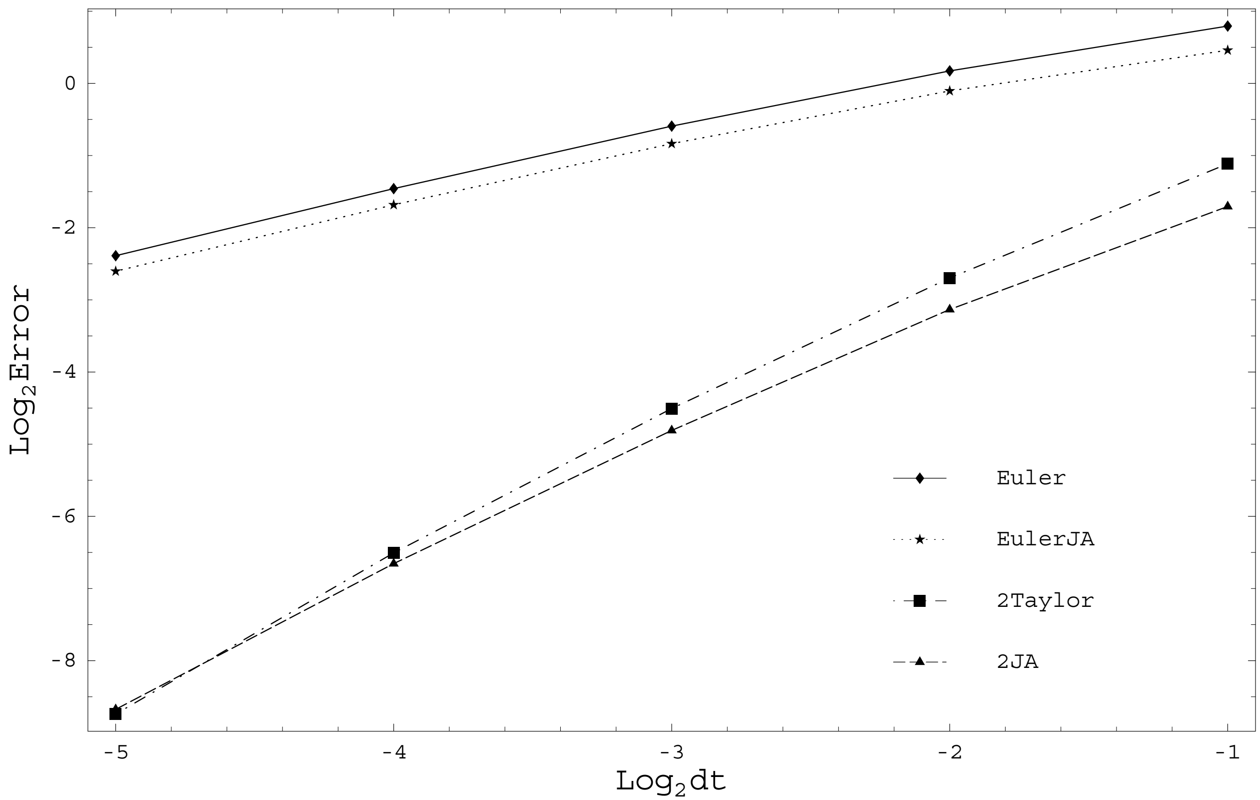
$$Y_{t_{n+1-}} = Y_{t_n} + \frac{1}{2}\Big\{a(\bar{Y}_{t_{n+1-}}) + a\Big\}\Delta_{t_n} + b\Delta W_{t_n}$$

  with predictor

$$\bar{Y}_{t_{n+1-}} = Y_{t_n} + a\Delta_{t_n} + b\Delta W_{t_n}$$

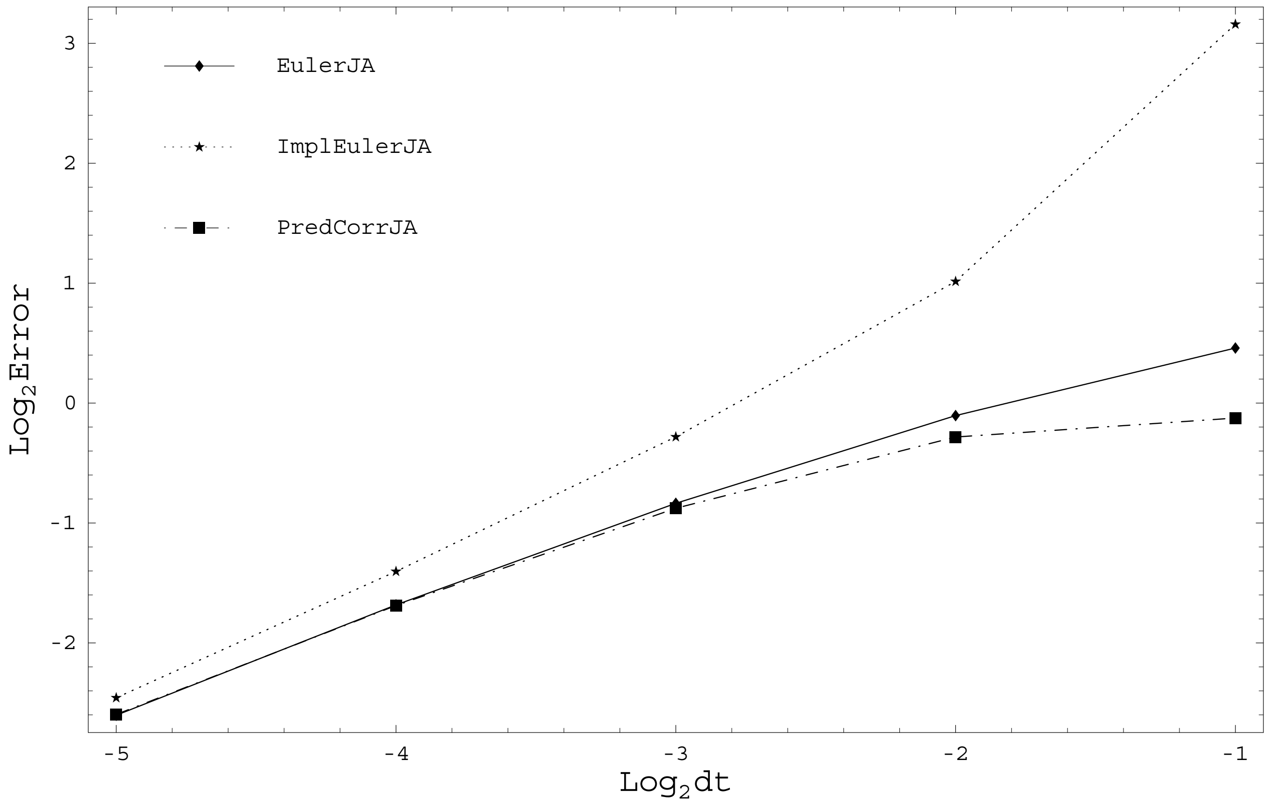
- $\beta = 1$

Numerical Results



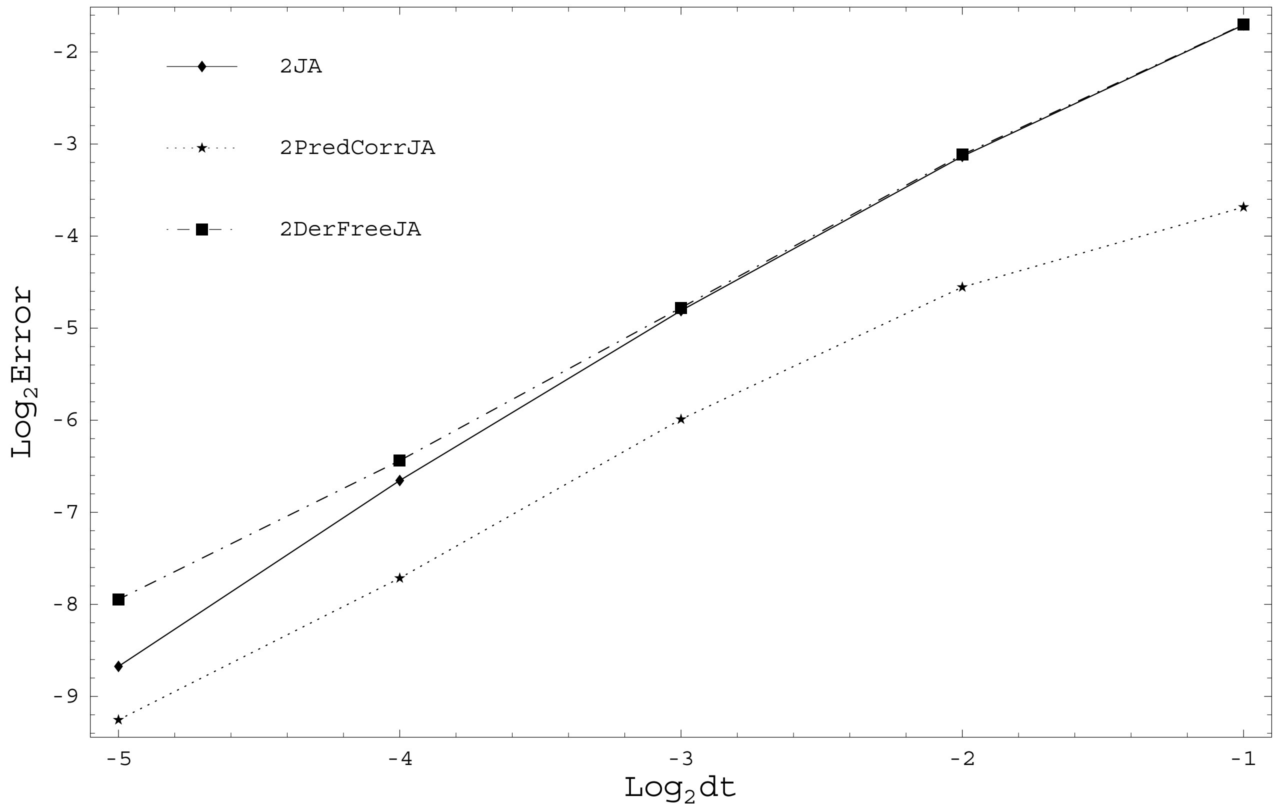
Log-log plot of weak error versus time step size.

Numerical Results



Log-log plot of weak error versus time step size.

Numerical Results



Log-log plot of weak error versus time step size.

Weak Taylor Approximations

- higher order schemes : time, Wiener and Poisson multiple integrals
- higher order schemes: computational effort dependent on intensity
- random jump size difficult to handle

Jump-Adapted Weak Approximations

- tractable higher order schemes: only time and Wiener multiple integrals

- pure diffusion approximations $\Rightarrow$ jump-adapted approximations

- random jump size easy to handle

- simplified schemes

- computational effort dependent on the intensity