

The JKW Algorithm

In this section we'll focus on the JKW algorithm for additive stochastic noise (Ref.~\cite{JenKloWin2011}).

Our basic equation is

$$dX_t = (\nu \partial_x^2 X_t + F(X_t))dt + df_t$$

F is a nonlinear function of X_t and $f_t = BdW_t$, where B is an operator which might create a spatial correlation structure on dW_t , but its time correlation is still white-noise.

In Fourier space we have

$$d\hat{X}_t = \left(-4\pi^2\nu k^2 \hat{X}_t + \hat{F}(\hat{X}_t)\right)dt + d\hat{f}_t$$

with the change of variables

$$\hat{Y}_t = e^{4\pi^2\nu k^2 t} \hat{X}_t$$

we have

$$d\hat{Y}_t = e^{4\pi^2\nu k^2 t} \hat{F}\left(e^{-4\pi^2\nu k^2 t} \hat{Y}_t\right)dt + e^{4\pi^2\nu k^2 t} d\hat{f}_t$$

We integrate this over an interval (t_n, t_{n+1})

$$\hat{Y}(t_{n+1}) = \hat{Y}(t_n) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 s} \hat{F}\left(e^{-4\pi^2\nu k^2 s} \hat{Y}(s)\right) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 s} d\hat{f}_s$$

Let us go back to the original variables

$$e^{4\pi^2\nu k^2 t_{n+1}} \hat{X}(t_{n+1}) = e^{4\pi^2\nu k^2 t_n} \hat{X}(t_n) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 s} \hat{F}\left(\hat{X}(s)\right) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 s} d\hat{f}_s$$

and now, with $\delta t = t_{n+1} - t_n$

$$\hat{X}(t_{n+1}) = e^{-4\pi^2\nu k^2 \delta t} \hat{X}(t_n) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 (s-t_{n+1})} \hat{F}\left(\hat{X}(s)\right) + \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 (s-t_{n+1})} d\hat{f}_s$$

We approximate the first integral, which involves the nonlinear term, by its simplest approximation

$$\int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 (s-t_{n+1})} \hat{F}\left(\hat{X}(s)\right) \approx \delta t e^{-4\pi^2\nu k^2 \delta t} \hat{F}\left(\hat{X}(t_n)\right)$$

This is a first order IF method, it approximates the nonlinear integrand by a polynomial (this is explained in Yang Grooms Julien 2021)

ETD methods, on the other hand, approximate only the nonlinear term by a polynomial,

$$\begin{aligned} \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 (s-t_{n+1})} \hat{F}\left(\hat{X}(s)\right) &\approx \hat{F}\left(\hat{X}(t_n)\right) \int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2 (s-t_{n+1})} \\ &= \hat{F}\left(\hat{X}(t_n)\right) \frac{1 - e^{-4\pi^2\nu k^2 \delta t}}{4\pi^2\nu k^2} \end{aligned}$$

In higher order IF methods we'd have

$$\phi(s) = \widehat{F}(\widehat{X}(s)) \approx \phi(t_n) + \phi'(t_n)s + \frac{1}{2}\phi''(t_n)s^2 + \dots$$

What I still don't understand is how to extend these results through higher order numerical integration methods. But the paper is written in the deterministic setup, where it is not that hard to implement higher order methods. The JKW algorithm is a stochastic version of their method. As they argue, one could use IFRK or ETD RK, increasing the accuracy of the solution. Ref.

\citep{YangGroomsJulien2021} shows that IF methods outperform ETD methods in the simulation of the MMT equation, but they have implemented IFRK3 method. I still have to understand how to do that. And that is for the deterministic setup. Even if X is smoother than the forcing (as we expected from a PDE, but I don't have any justification behind that), can we still use standard high order RK on the nonlinear term?

The stochastic forcing, the last term in the PDE, is a Gaussian random variable. Given that

$$\mathbb{E}[\widehat{f}(t, k)\widehat{f}^*(t', k')] = \delta(t - t')\delta(k - k')\widehat{C}(k)$$

so we have a Gaussian random variable of variance

$$\text{Var}\left[\int_{t_n}^{t_{n+1}} ds e^{4\pi^2\nu k^2(s-t_{n+1})} d\widehat{f}_s\right] = \widehat{C}(k) \frac{1 - e^{-8\pi^2\nu k^2\delta t}}{8\pi^2\nu k^2}$$

so we have the numerical scheme

$$\widehat{X}_{j,n+1} = e^{-4\pi^2\nu k_j^2\delta t} \widehat{X}_{j,n} + \delta t e^{-4\pi^2\nu k_j^2\delta t} \mathcal{F}[F(X_{\cdot,n})]_j + \sqrt{\widehat{C}(k_j) \frac{1 - e^{-8\pi^2\nu k_j^2\delta t}}{8\pi^2\nu k_j^2}} \chi_{j,n}$$

j is an index for the Fourier mode, n is an index in discretized time and $\chi_{j,n}$ are standard normal Gaussian random variables, and independent.

Notice that the zero mode diverges, as we have $k_0 = 0$. We obtain an update equation for the zero mode through the limit in $j \rightarrow 0$, which gives us

$$\widehat{X}_{0,n+1} = \widehat{X}_{0,n} + \delta t \mathcal{F}[F(X_{\cdot,n})]_0 + \sqrt{\delta t \widehat{C}(k_0)} \chi_{0,n}$$

At this point, I'm still not totally sure about the forcing correlation function. Let's see.

$$f_t = B \xi_t$$

Where η_t is a white noise

$$\begin{aligned}\mathbb{E}[\xi_t] &= 0 \\ \mathbb{E}[\xi_t \xi_s] &= \delta(t - s)\end{aligned}$$

White noise can be written as a KL expansion (source: Martin Hairer on SE, <https://mathoverflow.net/a/233914>)

$$\xi = \sum_{n=1}^{\infty} e_n \xi_n \rightarrow \int_{\mathbb{R}} dk e^{2\pi i k x} \widehat{W}(k) \rightarrow \text{RANDN} \sqrt{\delta x}$$

where ξ_n are iid Gaussians and e_n are an orthonormal basis of $L^2([0, 1])$

The right arrow means the continuous limit in a very physicist sense, since the Fourier transform of white noise makes no sense. So yes, the basis e_n does depend on x , and on k (which is n in the discrete representation).

Let's test one thing

$$\mathbb{E}[\xi(x)\xi(y)] = \sum_{n,m=1}^{\infty} e_n(x)e_m(y)\mathbb{E}[\xi_n\xi_m] = \sum_{n,m=1}^{\infty} e_n(x)e_m(y)\delta_{n,m} = \sum_n e_n(x)e_n(y)$$

So we'll just assume that, in our case

$$\sum_n e_n(x)e_n(y) \rightarrow \delta(x-y)$$

JKW define (p. 9)

$$e_n(x) = \sqrt{2} \sin(n\pi x) \rightarrow e^{2\pi i k_n x}$$

and (p. 11)

$$Bv = \sum_{n=1}^{\infty} b_n \langle e_n, v \rangle e_n$$

such that

$$f_t = B\xi_t = \sum_{n=1}^{\infty} b_n \langle e_n, \xi_t \rangle e_n \rightarrow \int_{\mathbb{R}} dk e^{2\pi i k x} b(k) \widehat{W}(k) \rightarrow \text{IDFT} \left[\sqrt{\widehat{C}_f(k)} \text{DFT} [\text{RANDN} \sqrt{\delta x}] \right]$$

So $\langle e_n, \xi_t \rangle$ is selecting the component n in Fourier space of the random field ξ_t , makes total sense.

such that

$$\begin{aligned} \mathbb{E}[f(t_1, x)f(t_2, y)] &= \mathbb{E} \sum_{n,m=1}^{\infty} b_n b_m \langle e_n, \xi \rangle \langle e_m, \xi \rangle e_n(x) e_m(y) \\ &= \mathbb{E} \sum_{n,m,p,q=1}^{\infty} b_n b_m \langle e_n, e_p \rangle \xi_p \langle e_m, e_q \rangle \xi_q e_n(x) e_m(y) \\ &= \mathbb{E} \sum_{n,m,p,q=1}^{\infty} b_n b_m \delta_{n,p} \xi_p \delta_{m,q} \xi_q e_n(x) e_m(y) \\ &= \mathbb{E} \sum_{n,m,p,q=1}^{\infty} b_n b_m \xi_n \xi_m e_n(x) e_m(y) \\ &= \sum_{n,m,p,q=1}^{\infty} b_n b_m \delta_{n,m} e_n(x) e_m(y) \\ &= \sum_{n=1}^{\infty} b_n b_n e_n(x) e_n(y) \end{aligned}$$

Fuck, what is that? Certainly some kind of convolution with white noise. But I have no idea.

In the continuous limit, we'd write

$$f(x) = \int_{\mathbb{R}} dk e^{2\pi i k x} g(k) \widehat{W}(k)$$

From which

$$\begin{aligned}
\mathbb{E}[f(x)\overline{f(y)}] &= \mathbb{E} \int_{\mathbb{R}} dk_1 e^{2\pi i k_1 x} g(k_1) \widehat{W}(k_1) \int_{\mathbb{R}} dk_2 e^{-2\pi i k_2 y} \overline{g(k_2)} \overline{\widehat{W}(k_2)} \\
&= \int_{\mathbb{R}} dk_1 e^{2\pi i k_1 x} g(k_1) \int_{\mathbb{R}} dk_2 e^{-2\pi i k_2 y} \overline{g(k_2)} \delta(k_1 - k_2) \\
&= \int_{\mathbb{R}} dk e^{2\pi i k(x-y)} g(k) \overline{g(k)} \\
&= C_f(x - y)
\end{aligned}$$

The Noise Term in JKW Algorithm

I guess the simplest way to understand this is:

I want to build a discretized version of the random field f with spectrum:

$$\mathbb{E} \left[\widehat{f}(k_1) \overline{\widehat{f}(k_2)} \right] = \delta(k_1 - k_2) \widehat{C}_f(k_1)$$

And I know how to do this, just build

$$\widehat{f}(k_n) = \sqrt{\widehat{C}_f(k_n)} \text{DFT} \left[\xi \sqrt{\delta x} \right]_n$$

where ξ is a vector of i.i.d. standard Gaussian random variables.

Now, for the integrated noise term, in the continuum limit, I want to build a random field with spectrum

$$\mathbb{E} \left[\int_{t_n}^{t_{n+1}} ds e^{4\pi^2 \nu k_1^2 (s-t_{n+1})} d\widehat{f}_s \int_{t_n}^{t_{n+1}} ds e^{4\pi^2 \nu k_2^2 (s-t_{n+1})} d\widehat{f}_s \right] = \delta(k_1 - k_2) \widehat{C}(k_1) \frac{1 - e^{-8\pi^2 \nu k_1^2 \delta t}}{8\pi^2 \nu k_1^2}$$

A discretized version of this is

$$\widehat{g}(k_n) = \sqrt{\widehat{C}(k_n) \frac{1 - e^{-8\pi^2 \nu k_n^2 \delta t}}{8\pi^2 \nu k_n^2}} \text{DFT} \left[\xi \sqrt{\delta x} \right]_n$$

The Numerical Scheme

So we summarize this as

We want a discretized representation of the PDE

$$d\widehat{X}_t = \left(-4\pi^2 \nu k^2 \widehat{X}_t + \widehat{F}(\widehat{X}_t) \right) dt + d\widehat{f}_t$$

Which is

$$\widehat{X}_{j,n+1} = e^{-4\pi^2 \nu k_j^2 \delta t} \widehat{X}_{j,n} + \delta t e^{-4\pi^2 \nu k_j^2 \delta t} \text{DFT}[F(X_{\cdot,n})]_j + \sqrt{\widehat{C}(k_j) \frac{1 - e^{-8\pi^2 \nu k_j^2 \delta t}}{8\pi^2 \nu k_j^2}} \text{DFT} \left[\xi \sqrt{\delta x} \right]_{j,n}$$

But if the forcing f_t has been built already, this can be written as

$$\widehat{X}_{j,n+1} = e^{-4\pi^2 \nu k_j^2 \delta t} \widehat{X}_{j,n} + \delta t e^{-4\pi^2 \nu k_j^2 \delta t} \text{DFT}[F(X_{\cdot,n})]_j + \sqrt{\frac{1 - e^{-8\pi^2 \nu k_j^2 \delta t}}{8\pi^2 \nu k_j^2}} \widehat{f}_{j,n}$$

Looking at the noise term in the equations we see that large k correspond to smaller time scales $T = 1/(\alpha k^2)$ if α is the diffusion coefficient. Hence large k require a very small dt , while small k correspond to very large typical times and we can use a bigger dt . This matches the observations that convergence is better for smaller k .

Why does the zero mode keep growing?

It is governed by

$$\hat{X}_{0,n+1} = \hat{X}_{0,n} + \delta t \text{DFT}[F(X_{\cdot,n})]_0 + \sqrt{\delta t} \hat{f}_{0,n}$$

In the specific case of

Analytical Tests

We can test the algorithm with a linear equation

$$\partial_t X_t = \nu \partial_x^2 X_t - \alpha X_t + f_t$$

which in Fourier space is written

$$\partial_t \hat{X}_t = -4\pi^2 \nu k^2 \hat{X}_t - \alpha \hat{X}_t + \hat{f}_t$$

its solution is

$$\hat{X}(t, k) = \int_0^t ds e^{-(4\pi^2 \nu k^2 + \alpha)(t-s)} \hat{f}(s, k)$$

with a spectrum

$$\mathbb{E} [\hat{X}(t, k) \hat{X}^*(t, k')] = \hat{C}_f(k) \delta(k - k') \int_0^t ds e^{-(8\pi^2 \nu k^2 + \alpha)(t-s)}$$

or

$$\mathbb{E} [\hat{X}(t, k) \hat{X}^*(t, k')] = \hat{C}_f(k) \delta(k - k') \frac{1 - e^{-(8\pi^2 \nu k^2 + \alpha)t}}{8\pi^2 \nu k^2 + \alpha}$$

and the large time / k limit

$$\lim_{t \rightarrow \infty} \mathbb{E} [\hat{X}(t, k) \hat{X}^*(t, k')] \underset{k \rightarrow \infty}{\sim} \hat{C}_f(k) \delta(k - k') \frac{1}{(8\pi^2 \nu k^2 + \alpha)}$$

The variance, given a Gaussian correlation function, is

$$\int_{\mathbb{R}} dk E(t, k) = \frac{\sqrt{\pi} L e^{\frac{\alpha L^2}{4\nu}} \left(\text{Erfc} \left(\frac{1}{2} L \sqrt{\frac{\alpha}{\nu}} \right) - \text{Erfc} \left(\frac{1}{2} \sqrt{\frac{\alpha(L^2 + 4\nu t)}{\nu}} \right) \right)}{2\sqrt{\alpha\nu}}$$

HOWTO

1. Define simulation prefix in file `run.sh` and parameters in `params.py`
2. send data to remote computer with `make send`
- 3.

TODO

1. Dealiasing
2. Other analytical tests (Kloeden Platen p. 118)
3. Understand difference IF and ETD, Higher order

SIMULATIONSYang Grooms Julien 2021)

Estelas

0. EM, $\alpha=0.1$, $dt=0.2 dx^2$
1. JKW, $\alpha=0.1$, $dt=0.2 dx^2$
2. JKW, $\alpha=0.1$, $dt=2 dx^2$

Tabua

- 0: Euler-Maruyama, $\alpha=0$, $dt=2 dx^2$, works
- 1: JKW, $\alpha=0$, $dt=2dx^2$
- 2: *JKW*, $\alpha=0$, $dt=0.2dx^2$