Projective integration of expensive multiscale stochastic simulation

Xiaopeng Chen[1] A. J. Roberts[2]
Ioannis G. Kevrekidis[3]

(Received 10 December 2010; revised 16 June 2011)

Abstract

Detailed microscale simulation is typically too computationally expensive for the long time simulations necessary to explore macroscale dynamics. Projective integration uses bursts of the microscale simulator, on microscale time steps, and then computes an approximation to the system over a macroscale time step by extrapolation. Projective integration has the potential to be an effective method to compute the long time dynamic behaviour of multiscale systems. However, many multiscale systems are significantly influenced by noise. By a maximum likelihood estimation, we fit a linear stochastic differential equation to short bursts of data. The analytic solution of the linear stochastic differential equation then estimates the solution over a macroscale, projective integration, time step. We explore how the noise affects the

http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/3764 gives this article, © Austral. Mathematical Soc. 2011. Published August 10, 2011. ISSN 1446-8735.

projective integration in two different methods. Monte Carlo simulation suggests design parameters offering stability and accuracy for the algorithms. The algorithms developed here may be applied to compute the long time dynamic behaviour of multiscale systems with noise and to exploit parallel computation.

Contents

1 Introduction

Projective integration is used to extrapolate the long time dynamic behaviour of mutiscale systems from small time steps [9, 7]. Many applications of projective integration are to systems based upon microscopic simulators with inherently stochastic or chaotic dynamics [4, e.g.]. Thus in many applications the projective integration is influenced by apparent noise [11, 5, e.g.]. Calderon [4] and Givon et al. [8] considered aspects of projective integration for such systems. Here we focus on and explore just the one aspect of how the noise affects the projective integration.

Two methods are investigated by Monte Carlo simulation. The methods explored here may be applied to compute the long time dynamic behaviour of multiscale systems with noise. Future considerations will include developing methods suitable for reducing simulation times with parallel computing [8].

We contribute to the understanding of projective integration by applying it to the class of stochastic processes governed by statistically homogeneous stochastic differential equations of the form

$$dX = a(X)\,dt + b(X)\,dW\,, \tag{1}$$

where the drift $a(X)$ and the volatility $b(X)$ are smooth deterministic functions, and where all noises can be gathered into the one Wiener process $W(t,\omega)$. Section 2 introduces basic projective integration for such SDEs. The great simplification of this exploration is that we suppose there are no hidden fast dynamics: the process $X(t,\omega)$ is the only dynamical variable. Thus we do not address any issues associated with 'lifting' and 'restriction' of fast variables in a fast-slow stochastic system [9, e.g.]. Instead we address, on their own, the issues of accuracy and stability of projective integration methods applied to the many multiple paths of a stochastic process.

The first result of Section 2 is that the basic projective integration scheme often has significant bad predictions. Consequently, Section 3 introduces a predictor-corrector version of projective integration. Example simulations provide evidence that such a predictor-corrector version may be an effective method for macroscale simulations.

2 A stochastic model underlies projective integration

We explore the projective integration of the general stochastic differential equation (1). Suppose there is some 'unknowable' microscale system that generates realisations of a stochastic process $X(t,\omega)$ that is both of interest and observable: for example, the stochastic process $X(t,\omega)$ could be the population count in a complicated particle/agent simulation. Crucially, also assume that the realisations are very expensive to compute (or, potentially, to perform experimentally). Thus to simulate over long times we use projective integration: execute a burst of the microscale system to generate a short time series of the process $X(t,\omega)$; extrapolate from this short burst over a macroscale time step; then restart the simulator at the latter time and repeat. The challenge for a stochastic system is that the extrapolation is strongly sensitive to the noise in the microscopic burst.

In overview, a projective integration method for the stochastic system is the following recursive algorithm. Choose a burst time interval δt and a projective integration time step Δt.

- Suppose the system is at state X_n at time t_n; that is, $X(t_n,\omega) = X_n$.
- Execute the expensive microscale simulator to generate a particular realisation $X(t,\omega)$ for a burst $t_n \leqslant t \leqslant t_n' = t_n + \delta t$ starting from $X(t_n,\omega) = X_n$ and ending at $X_n' = X(t_n',\omega)$.
- We fit the burst with the linear Itô SDE model

  $$dX = (a_0 + a_1 X)\,dt + (b_0 + b_1 X)\,dW\,, \tag{2}$$

  for constants a_i and b_i. Then, from the burst data, the maximum likelihood estimator of Ait-Sahalia et al. [1, 2, 3] for the SDE (2) [private communication] estimates the four drift and volatility coefficients in the SDE (2).
- Kloeden and Platen [10] give the analytic solution of the linear SDE (2) as the following: given the initial condition $X(t_n',\omega) = X_n'$,

  $$\begin{aligned} \text{let } \varphi(t,\omega) &= \exp\left[(a_1 - b_1^2/2)t + b_1 W(t,\omega)\right], \\ \text{then } X^{(2)}(t,\omega) &= \varphi \times \left[\frac{X_n'}{\varphi(t_n')} + (a_0 - b_0 b_1)\int_{t_n'}^{t} \frac{dt}{\varphi} + b_0 \int_{t_n'}^{t} \frac{dW}{\varphi}\right]. \end{aligned} \tag{3}$$

Using the maximum likelihood estimated coefficients from the burst, the solution (3) then steps across a macroscale time to the numerical estimate

$$X_{n+1}(\omega) = X^{(2)}(t_{n+1}, \omega) \quad \text{at time } t_{n+1} = t_n + \delta t + \Delta t\,. \tag{4}$$

- Repeat the above for further macroscale time steps.

Others, such as Siettos et al. [11, §II.C] and Erban et al. [5, §5], fit such a burst with the near trivial Ito SDE model

$$dX = a_0\, dt + b_0\, dW\,, \quad \text{for constant } a_0 \text{ and } b_0, \tag{5}$$

and so take a projective integration time step of

$$\hat{X}_{n+1} = X_n' + a_0\, \Delta t + b_0\, \Delta W \quad \text{for some } \Delta W \sim N(0, \Delta t); \tag{6}$$

They generally set $\Delta W = 0$ (its maximum likelihood value). In an analogue of fitting the model (2), Gear et al. [6, §3] fitted polynomials to the signal $X(t)$ to project in time, but they did not explore errors induced by stochastic fluctuations. In simulating stochastic chemical reactions, Calderon [4] also used Ait-Sahalia's maximum likelihood estimator to fit the SDE (2) to bursts of data. However, in contrast to our approach, Calderon used only the deterministic part of (2) to predict the future dynamics.

The above, Euler-like, projective integration method, and the predictor-corrector method in Section 3, were applied to simulations of the Itô SDE

$$dX = -X\, dt + 0.03(1 + X)\, dW\,, \quad X(0, \omega) = 1\,. \tag{7}$$

This specific SDE is used throughout this article as an example. The only error is due to the nature of fitting a model to the limited amount of information available in the stochastic bursts. Although to be precise, the stochastic process to which we apply the projective integration methods is an $\mathcal{O}(\partial t)$ Milstein scheme that approximates the SDE (7) using a micro-time step ∂t,

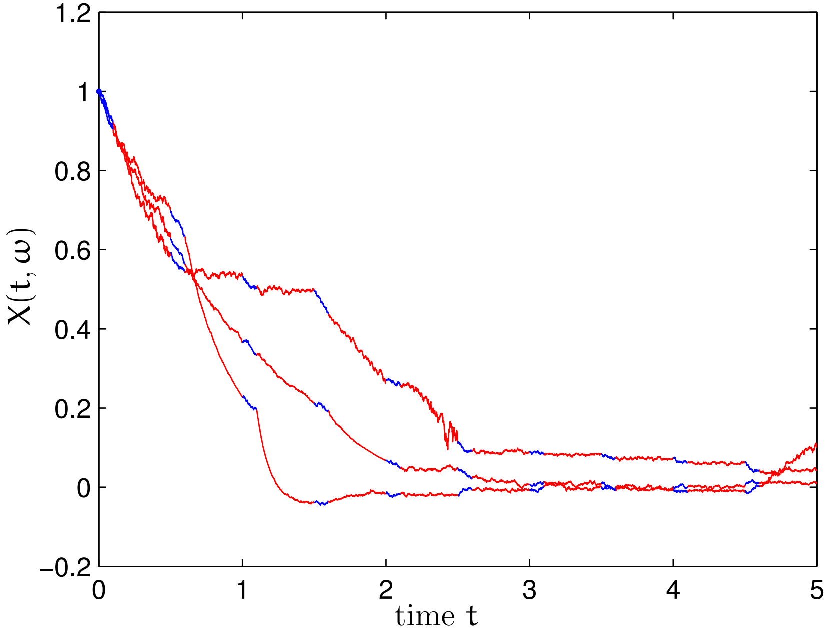


Figure 1: three realisations of the Euler-like projective integration of Section 2 applied to the microscale SDE (7) simulated with micro-time step $\partial t = 0.001$. In the projective integration: bursts (blue) lasted $\delta t = 0.1$; and using (3) the macroscale time step (red) in (4) was $\Delta t = 0.4$.

typically we selected the micro-step $\partial t = 0.001$. When we refer to the SDE (7), we actually mean the Milstein process. The difference should be negligible for our purposes.

Figure 1 shows three realisations of the projective integration applied to the SDE (7). They appear to be evolving towards the stochastic equilibrium about $X = 0$. However, the algorithm uncomfortably often 'jumps' in the projective step due to poor estimation, from the burst, of the coefficients in the SDE (2).

Such jumps induced by poor estimation constitute a severe limitation. Figure 2

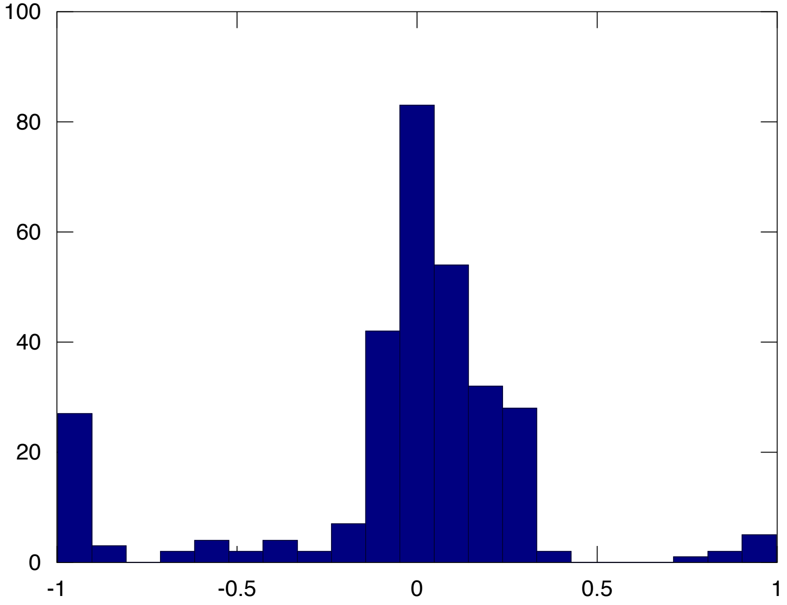
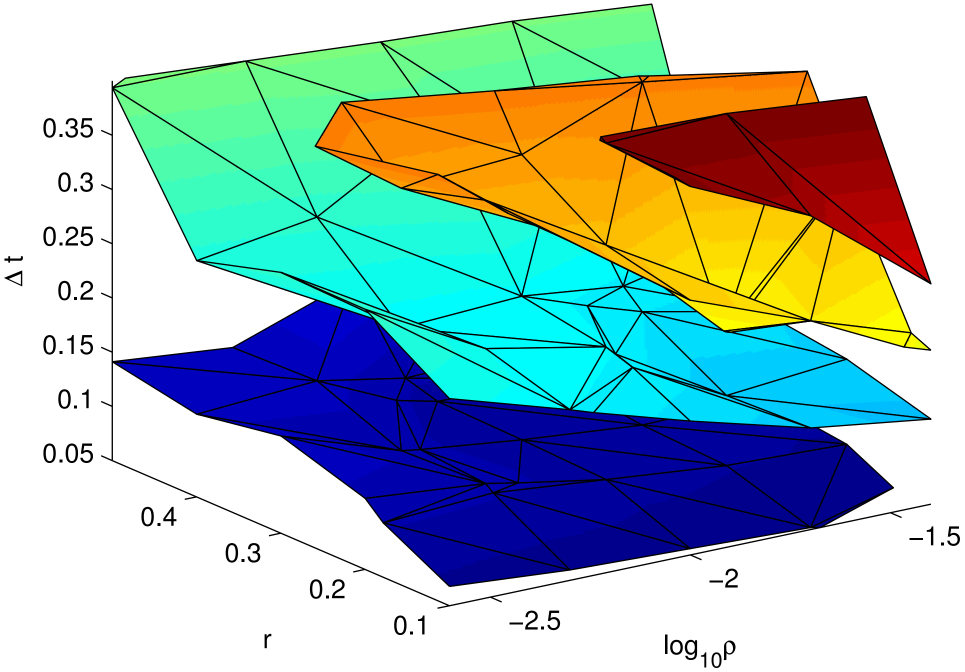


Figure 2: histogram of errors in one step of the projective integration (4) of Section 2 (the extreme bins contain all larger and smaller errors): the mean $= -0.76$, the standard deviation $= 10.34$, the median $= 0.02$ and the inter-quartile range $= 0.19$. The microscale SDE (7) was simulated as in Figure 1.

plots a histogram of the errors after one step of the method applied to the SDE (7). In 300 realisations, a significant number had large errors showing up as fat tails in the histogram as shown by the errors grouped into the extreme leftmost and rightmost bins.

We systematically explored a range of parameter values by changing the macroscale step Δt, the length of the bursts $\delta t = r\,\Delta t$, and the microscale sampling interval $\partial t = \rho\,\delta t$. Then 300 realisations for each of 100 parameter combinations leads to Figure 3: plotted are isosurfaces of the median absolute error, MAE $=$ median$\left\{|X_1^{(4)}(\omega) - X^{(7)}(t_1,\omega)|\right\}$, where $X_1^{(4)}(\omega)$ and $X^{(7)}(t_1,\omega)$



(errorsEulerB03err.u3d)

Figure 3: isosurfaces of a one step error in projective Euler-like integration (4): the surfaces are for median absolute error MAE $= 0.03, 0.06, 0.09, 0.12$ (in order blue to red). Roughly, the error fits MAE $\approx 0.39\rho^{0.08}(1-r)^{1.1}(\Delta t)^{0.73}$.

are respectively the one step numerical estimate and analytic solution of the SDE (7). This error appears to behave acceptably. However, our choice of median error hides any fat tails in errors: such fat tails are generally seen in this simple implementation of projective integration.

3 Predictor-corrector stochastic projective integration

Figure 1 suggests that the most significantly sensitive part of the projective algorithm of Section 2 is the estimation of the drift coefficients a_i in the presence of noise. The volatility coefficients b_i seem less problematic. One way to get better estimates of the drift, inspired by predictor-corrector methods, is to do the following predictor-corrector scheme. The method is analogous to that proposed by Gear et al. [6, 7, e.g.]. The difference is in our use of an SDE model for the coarse time step. Here

- execute a first burst of the microscale simulator (blue in Figure 4);
- take a simple tentative predictor projective step using (6);
- execute a second burst of the microscale simulator starting from $\hat{X}_{n+1}$ at time t_{n+1} (green in Figure 4);
- use Ait-Sahalia's [1, 2, 3] maximum likelihood estimator for the SDE (2) [private communication] to fit the SDE (2) to the composite data of both bursts together;
- then take a corrected projective step using the solution (3) (red in Figure 4).

Figure 4 shows three realisations of one example. They appear to be much more satisfactory than the Euler-like scheme of the previous section. Figure 5 supports the marked increase in accuracy of the predictor-corrector method: the histogram of errors shows an overall smaller error and no fat tails in

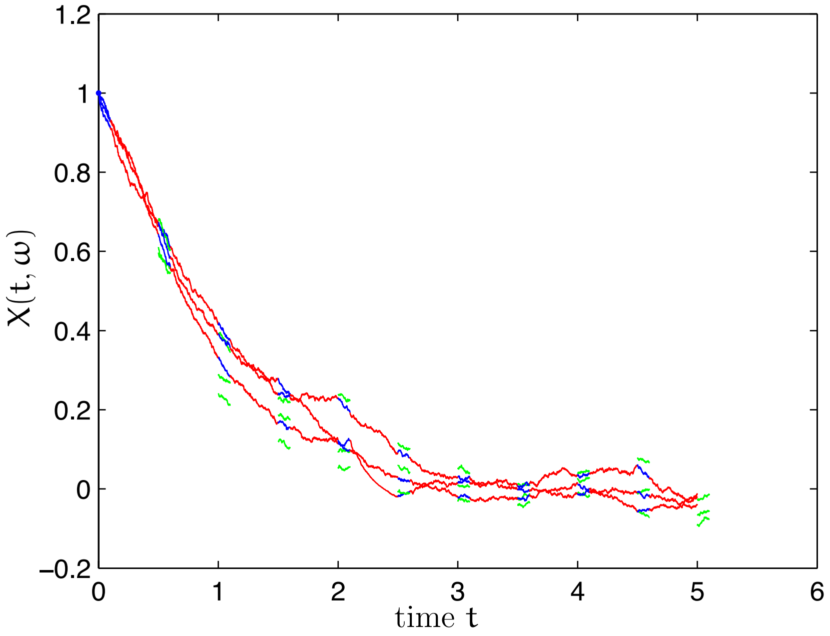


Figure 4: three realisations of the 'predictor-corrector' projective integration of Section 3 applied to the microscale SDE (7) with micro-time step $\partial t = 0.001$. In the projective integration: initial bursts (blue) lasted $\delta t = 0.1$; second bursts (green) were started following the simple prediction (6); and lastly using (3) the macroscale time step (red) in (4) was $\Delta t = 0.4$.

the distribution of errors. The lack of fat tails suggests that this predictor-corrector method should be much more reliable.

How does the error of the method vary with design parameters? Figure 6 plots isosurfaces of the median absolute errors for a range of parameters. A reasonable fit to the data in this domain is

$$\text{MAE} \approx 0.067\rho^{0.03}(1-r)^{2.1}(\Delta t)^{0.27}. \tag{8}$$

The almost absent dependence upon the micro-scale time steps, via ρ, indicates that the resolution of the data sampling within a burst in not particularly

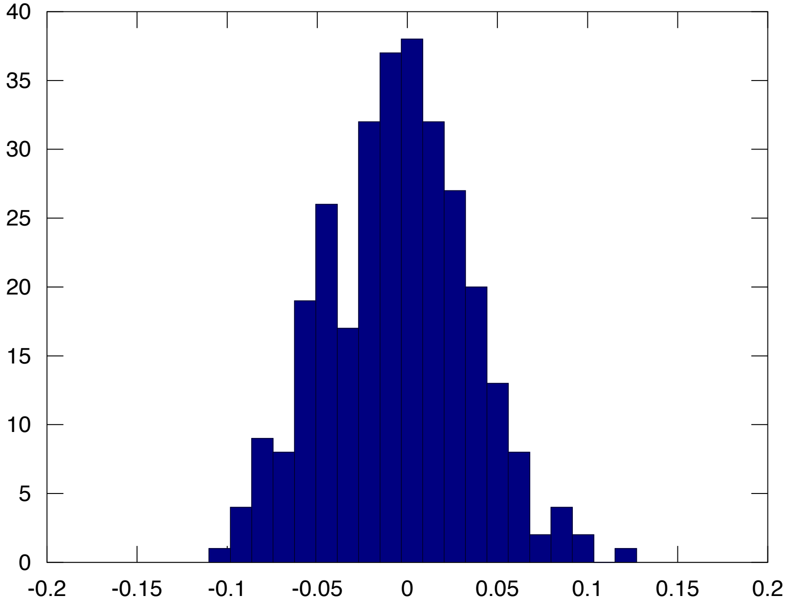
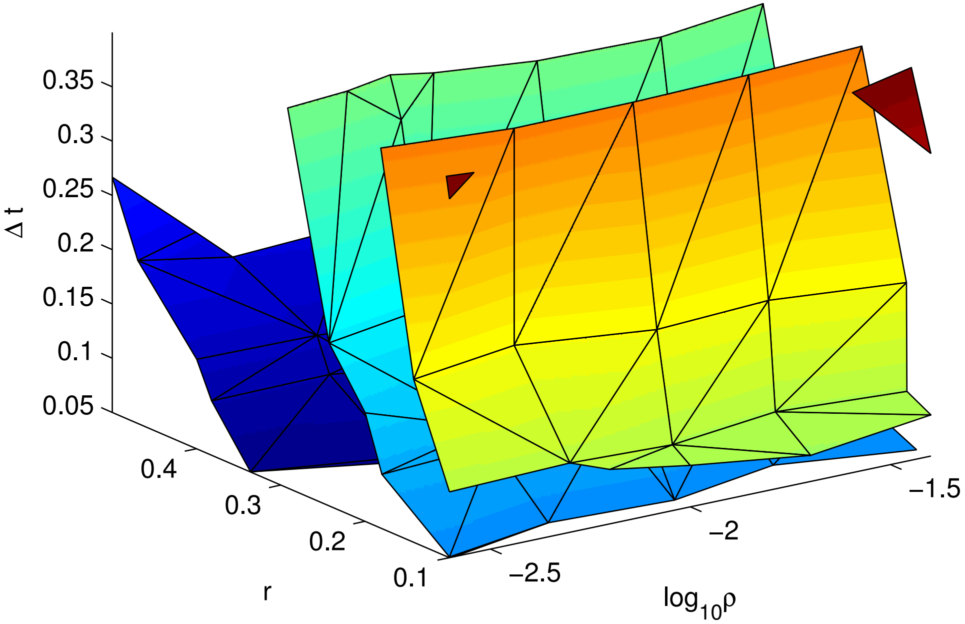


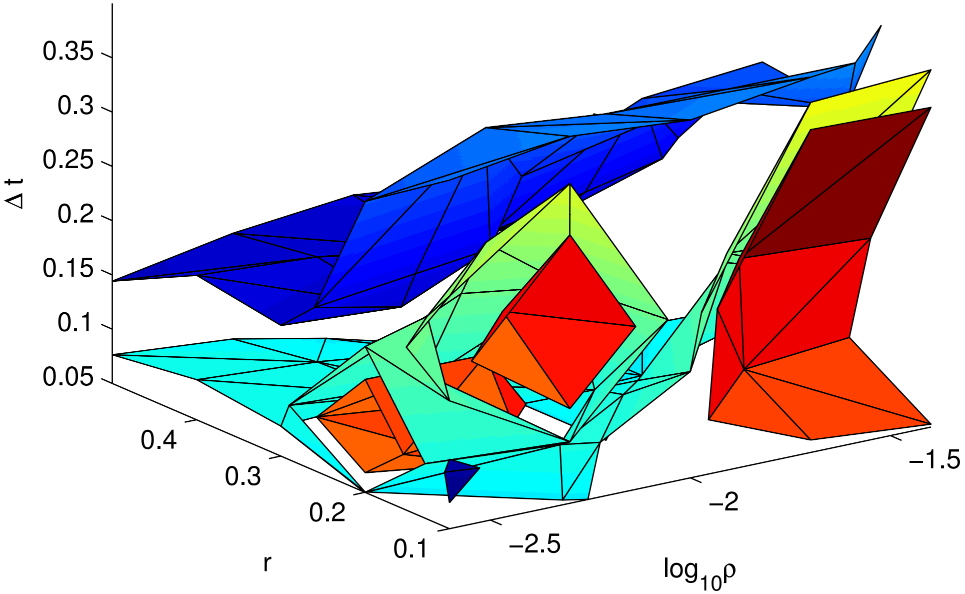
Figure 5: histogram of errors in one step of the 'predictor-corrector' projective integration of Section 3: the mean $= -0.005$, the standard deviation $= 0.040$, the median $= -0.004$ and the interquartile range $= 0.052$. The microscale SDE (7) was simulated as in Figure 4.

important. The $(1-r)^2$ factor reaffirms that the smaller the fraction of the projection step the better—not surprisingly. The error also depends on the time step Δt. But recall that the SDE here is in the class fitted exactly by the model SDE (2); thus in principle there need be no error in an integration step over an arbitrarily large interval. In general applications of the algorithm, the SDE of interest will not be in the class (2)—the class will be a local approximation—and so there will be additional error. This article does not explore the error caused by such local modelling. Here we focus on errors induced by the stochastic nature of the process.



(errorsHeunB03err.u3d)

Figure 6: isosurfaces of a one step error in projective predictor-corrector integration: the surfaces are for median absolute error MAE $= 0.01, 0.02, 0.03, 0.04$ (in order blue to red, obtained from 300 realisations). Roughly, the error fits MAE $\approx 0.067\rho^{0.03}(1-r)^{2.1}(\Delta t)^{0.27}$.



(errorsHeunB03smr.u3d)

Figure 7: isosurfaces of the, one step, error ratio RMS / MAE in projective predictor-corrector integration: the surfaces are for ratios $2, 6, 20$ (in order blue to red).

Figure 6 continues to use the median absolute error as a measure of error. The reason is that this predictor-corrector algorithm is not always reliable. For some design parameters the predictions do have fat tails, although apparently not as obese as those for the simple Euler-like projective method. Figure 7 plots isosurfaces that indicate parameter regimes of good and poor behaviour. The isosurfaces are those of the ratio between the root mean square error (RMS) of the 300 realisations of the error to the median absolute error (MAE). Recall that for a normal distribution the ratio $\sigma/\text{MAE} = 1.4826$. Thus we identify roughly several parameter regimes: above the blue surface in Figure 7, the regime of $\text{RMS}/\text{MAE} < 2$, the predictions should be reliable as we expect the distribution of errors to be like those in Figure 5; below the blue surface and above the cyan surface, $2 < \text{RMS}/\text{MAE} < 6$, there will be increasing numbers of poor predictions; below the cyan surface $6 < \text{RMS}/\text{MAE}$ and thus we expect fat tails, tending to obese tails below the red surface. Avoiding bad outlier predictions is the major concern of this article.

Recall that Figure 6 and its approximation (8) suggest accuracy improves when reducing the time step Δt. However, Figure 7 indicates that we should not take too small a time step Δt as otherwise rare bad predictions become significantly common: the distribution of errors possesses 'fat tails' for small Δt.

Figure 7 reaffirms that there is little point in recording more data from the micro-scale as there is only a weak trend favouring small ρ.

4 Conclusion

This research focussed upon some design parameters of the two numerical algorithms in this empirical exploration. We kept the noise level fixed at volatility circa 0.03–0.06 because this level of noise is significant, without being dominant. By applying the algorithms to the SDE (7), which is in the class of the model SDE (2), we focus on errors induced by the stochastic nature of the processes—there is no error in the functional form of the SDE underlying the projective integration.

The predictor-corrector algorithm appears to be reasonably accurate and reasonably robust to the presence of noise when compared with Euler-like projective integration. However, Section 3 highlights that stochastic projection suffers from 'fat tailed' errors if one uses macroscale time steps that are too small. In applications to general SDEs (1), the time step cannot be too large either as the error in approximating the SDE by the model (2) will be large. In general applications we will have to be careful about the time steps for stochastic projective integration.

The next step is to confirm these empirical Monte Carlo simulation results by stochastic Taylor expansions.

Acknowledgements The Australian Research Council supported this research with grants DP0774311 and DP0988738. The work of I.G.K. was partially supported by the US Department of Energy (DE-SC0002097). We thank Prof. Ait-Sahalia for his maximum likelihood code for SDE (2).

References

[1] Yacine Ait-Sahalia. Maximum likelihood estimation of discretely sampled diffusions: a closed-form approximation approach. *Econometrica*, 70(1):223–262, 2002. C664, C669

[2] Yacine Ait-Sahalia. Likelihood inference for diffusions: a survey. In Jianqing Fan and Hira L. Koul, editors, *Frontiers in Statistics: in Honor of Peter J. Bickel's 65th Birthday*, chapter 17, pages 369–405. Imperial College Press, 2006. C664, C669

[3] Yacine Ait-Sahalia and Robert Kimmel. Maximum likelihood estimation of stochastic volatility models. *Journal of Financial Economics*, 83(2):413–452, 2007. doi:10.1016/j.jfineco.2005.10.006 C664, C669

[4] C. P. Calderon. Local diffusion models for stochastic reacting systems: estimation issues in equation-free numerics. *Molecular Simulation*, 33(9–10):713–731, August 2007. doi:10.1080/08927020701344740 C662, C665

[5] Radek Erban, Ioannis G. Kevrekidis, and Hans G. Othmer. An equation-free computational approach for extracting population-level behavior from individual-based models of biological dispersal. *Physica D: Nonlinear Phenomena*, 215(1):1–24, 2006. doi:10.1016/j.physd.2006.01.008 C662, C665

[6] C. W. Gear, I. G. Kevrekidis, and C. Theodoropoulos. 'Coarse' integration/bifurcation analysis via microscopic simulators: micro-Galerkin methods. *Computers and Chemical Engrg*, 26:941–963, 2002. C665, C669

[7] C. W. Gear and Ioannis G. Kevrekidis. Projective methods for stiff differential equations: Problems with gaps in their eigenvalue spectrum. *SIAM Journal on Scientific Computing*, 24(4):1091–1106, 2003. doi:10.1137/S1064827501388157 C662, C669

[8] Dror Givon and Ioannis G. Kevrekidis. Coarse-grained projective schemes for certain singularly perturbed stochastic problems. Technical report, Princeton University, 2008. C662, C663

[9] Ioannis G. Kevrekidis and Giovanni Samaey. Equation-free multiscale computation: Algorithms and applications. *Annu. Rev. Phys. Chem.*, 60:321–44, 2009. doi:10.1146/annurev.physchem.59.032607.093610 C662, C663

[10] P. E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations*, volume 23 of *Applications of Mathematics*. Springer–Verlag, 1992. C664

[11] C. I. Siettos, M. D. Graham, and I. G. Kevrekidis. Coarse brownian dynamics for nematic liquid crystals: Bifurcation, projective integration,

and control via stochastic simulation. *J. Chemical Physics*, 118(22):10149–10156, 2003. doi:10.1063/1.1572456 C662, C665

Author addresses

1. **Xiaopeng Chen**, School of Mathematical Sciences, University of Adelaide, South Australia.
   mailto:xioapeng.chen@adelaide.edu.au
2. **A. J. Roberts**, School of Mathematical Sciences, University of Adelaide, South Australia.
   mailto:anthony.roberts@adelaide.edu.au
3. **Ioannis G. Kevrekidis**, Department of Chemical and Biological Engineering and Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544, USA.
   mailto:yannis@princeton.edu