Compare Katzenberger's formulas with my analysis of stochastic ODEs

A.J. Roberts

12 Dec 2022 – December 23, 2022

Contents

Discussion

Katzenberger (1991) argued a geometric basis to modelling the slow dynamics in a slow-fast stochastic differential equation. It is great that this argument accounts for nonlinear noise-noise interactions, and the drift effects they cause. This modelling is Markovian.

In contrast, stochastic slow manifold theory developing from Boxler (1989) leads to a proof that non-Markovian effects are generally required (Chao and Roberts 1996; Roberts 2008). However, some additional *weak solution* analysis may justify a Markovian model (Chao and Roberts 1996; Roberts 2006), albeit only emergent from algebraic decay of transients.

Sections 1 to 3 explore three simple SDE systems to understand the relation between the modelling of Katzenberger (1991) and that of systematic stochastic slow manifolds (Roberts 2008; Roberts 2009–2022).

Section 4 codes the procedure of Parsons and Rogers (2015) and Parsons and Rogers (2017), in the computer algebra package Reduce (Hearn and Schöpf 2018), to evaluate the SDE modelling of Katzenberger (1991). Choose a system to analyse with the following parameter, in $\{1,2,3\}$:

```
sys:=3;
```

Comparison and questions

- Katzenberger (1991) analysis is based upon heuristic arguments about the geometry near a deterministic critical manifold. The stochastic slow manifolds (SSM) are based upon exact algebra of the system for all states in a finite domain, to a chosen error (e.g., Roberts 2008).
- Katzenberger (1991) models are Markovian, whereas SSM theory proves that non-Markovian effects generally occur—as seen in all three examples. The missed non-Markovian effects are the same order of magnitude as the drift resolved by Katzenberger (1991).
- Katzenberger (1991) assumes one can choose a deterministically defined slow variable. Whereas I prove that in general, to avoid non-Markovian *linear* effects, that the true slow variable is stochastically related to state space variables (e.g., Roberts 2008).
- The drifts predicted by the two approaches agree in the first two examples, but quantitatively disagree in the third example. Since the SSM result is based upon exact algebra, *either Katzenberger (1991) is wrong, or Parsons and Rogers (2015) and Parsons and Rogers (2017)*

wrongly expresses his results, or my code wrongly implements their expressions. Which? What happens in the example of Section 3?

- But even when the drifts agree, the SSM proves that the drifts only emerge from transients that decay algebraically in time, whereas Katzenberger (1991) gives no hint of such required long time spans.
- Questions about Parsons and Rogers (2015) and Parsons and Rogers (2017): is (18) really the required pseudo-inverse? and (19) has multiple solutions, so what is "the" solution?
- Equations (22) and (23) in Parsons and Rogers (2017) look wrong to me: for example, $J = \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}$ then (22) gives the pseudo-inverse is $J^+ := \frac{1}{\lambda}J = -J = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, whereas (18) gives $J^+ := +J = \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}$. Instead, do they really mean $J^+ := \frac{1}{\lambda^2}J$?

  And from the given (22), (23) should surely be $P = I - \frac{1}{\lambda}J^2$?

1 Irreducible noise more-or-less matches

This first system (`sys` = 1) is (nearly) the canonical irreducible system identified by Chao and Roberts (1996)—irreducible because its stochastic slow manifold is the same algebraic form as the system itself! In Ito form (and with $\sqrt{\mu} = \sigma$) the system is

$$\#1 \text{ Ito}, \qquad dx_1 = \sigma x_2 dW, \qquad dx_2 = -x_2 dt + \sigma dW. \tag{1.1}$$

In the Stratonovich form, with white noise $\eta = dW/dt$, this is

$$\#1 \text{ Strat}, \qquad \dot{x}_1 = -\tfrac{1}{2}\sigma^2 + \sigma x_2 \eta\,, \qquad \dot{x}_2 = -x_2 + \sigma\eta\,. \tag{1.2}$$

For this and the other two examples, the x_1-axis, $x_2 = 0$, is the deterministic ($\sigma = 0$) critical manifold. Deterministically the isochrons, the stable foliation, are just parallel lines $x_1 =$ constant.

The code of Section 4 here generates the output of Section 4.1. It asserts that for $z = x_1$ the slow model is simply

$$\dot{z} = 0, \tag{1.3}$$

that is, nothing at all.

For comparison, my web service (Roberts 2009–2022) constructs the coordinate transform to XY-variables, where $y_1 = x_2$ and $w_1 = \eta$,

$$x_1 = X_1 - \sigma e^{t}\star w_1\, Y_1 + O\big(\sigma^2\big)$$
$$y_1 = Y_1 + \sigma e^{-1t}\star w_1 + O\big(\sigma^2\big) \qquad (1.4)$$

And in these new variables the (Stratonovich) state space evolution is (the parameter ε in my analysis (Roberts 2009–2022) is a robust *ordering parameter* to more-or-less ensure convergence to a valid model—usually set my $\varepsilon = 1$)

$$\dot X_1 = -1/2\sigma^2\varepsilon + \sigma^2 e^{-1t}\star w_1\, w_1 + O\big(\sigma^3\big)$$
$$\dot Y_1 = -Y_1 + O\big(\sigma^3\big) \qquad (1.5)$$

In the new XY-coordinate system: $Y_1 \to 0$ exponentially quickly; leaving X_1 to be the *true slow variable* for all time; as throughout X_1 evolves as above via the multiplicative noise-noise interactions—the system itself is linear in dynamical variables.

The slow X_1 evolution is non-Markovian through the dependence upon past history of the noise, but also the coordinate transform and hence the interpretation of X_1 is non-Markovian in its generic dependence upon the near-future of the noise via the term $\sigma e^{t}\star w_1\, Y_1$, albeit zero on the stochastic slow manifold.

Chao and Roberts (1996) and Roberts (2006) argued that one could *weakly model* the X_1 evolution via analysis of its Fokker–Planck PDE. The result is that on long times, with errors decaying algebraically in time, one may be justified in using the *weak* model $\dot X_1 = \sigma^2 \frac{1}{2}\eta'$ for some new white noise η'. The zero drift agrees with (1.3) of Katzenberger (1991), but (1.3) misses the 'small' fluctuations, and misses the long-time required for justification.

2 Parabolic isochrons more-or-less matches

This second system (`sys` = 2) has a drift simply generated by excursions off the deterministic slow manifold. In Ito form the system is

$$\#2\text{ Ito,}\qquad dx_1 = x_2^2 dt\,,\qquad dx_2 = -x_2 dt + \sigma dW. \tag{2.1}$$

In the Stratonovich form, with white noise $\eta = dW/dt$, this is

$$\#2\text{ Strat,}\qquad \dot x_1 = x_2^2\,,\qquad \dot x_2 = -x_2 + \sigma\eta\,. \tag{2.2}$$

Deterministically the isochrons are parabolas $x_1 = c - \frac{1}{2}x_2^2$ for each c.

The code of Section 4 here generates the output of Section 4.2. It asserts that for $z = x_1$ the slow model is simply

$$\dot z = \tfrac{1}{2}\sigma^2, \tag{2.3}$$

that is, a slow drift without any fluctuations.

For comparison, my web service (Roberts 2009–2022) constructs the coordinate transform to XY-variables, where $y_1 = x_2$ and $w_1 = \eta$,

$$\begin{aligned} x_1 &= X_1 + \sigma\varepsilon\big(- \mathrm{e}^{t}\star w_1\, Y_1 - \mathrm{e}^{-1t}\star w_1\, Y_1\big) - 1/2\varepsilon Y_1^2 + O\big(\sigma^2\big) \\ y_1 &= Y_1 + \sigma \mathrm{e}^{-1t}\star w_1\, + O\big(\sigma^2\big) \end{aligned} \tag{2.4}$$

And in these new variables the (Stratonovich) state space evolution is

$$\begin{aligned} \dot X_1 &= \sigma^2\varepsilon \mathrm{e}^{-1t}\star w_1\, w_1 + O\big(\sigma^3\big) \\ \dot Y_1 &= -Y_1 + O\big(\sigma^3\big) \end{aligned} \tag{2.5}$$

In the new XY-coordinate system: $Y_1 \to 0$ exponentially quickly; and throughout X_1 evolves as above via the nonlinear noise-noise interactions.

Chao and Roberts (1996) and Roberts (2006) showed that one could *weakly* model the X_1 evolution via analysis of its Fokker–Planck PDE. The result is that on long times, with errors decaying algebraically in time, one may be justified in using the *weak* model $\dot X_1 = \sigma^2(\frac{1}{2} + \frac{1}{2}\eta')$ for some new white noise η'. The drift agrees with (2.3) of Katzenberger (1991), but (2.3) misses the 'small' fluctuations, and misses the long-time required for justification.

3 Variously sloping isochrons appear problematic

This third system (`sys` = 3) has a drift generated by excursions off the deterministic slow manifold, but now due to variously sloping isochrons. In Ito form the system is

$$\#3 \text{ Ito}, \qquad dx_1 = x_1 x_2\, dt\,, \qquad dx_2 = -x_2 dt + \sigma dW. \tag{3.1}$$

In the Stratonovich form, with white noise $\eta = dW/dt$, this is

$$\#3 \text{ Strat}, \qquad \dot x_1 = x_1 x_2\,, \qquad \dot x_2 = -x_2 + \sigma\eta\,. \tag{3.2}$$

Deterministically the isochrons are the curves $x_1 = ce^{-x_2}$ for each c: these not only curve but also have varying slant.

The code of Section 4 here generates the output of Section 4.3. It asserts that for $z = x_1$ the slow model is

$$\dot z = \tfrac{1}{2}\sigma^2(2 + u_1 + u_2 z)z + \sigma z\eta, \tag{3.3}$$

in terms of two undetermined constants u_i arising in the non-unique solutions for matrices $X_1 := \begin{bmatrix} (u_1-1)/z & u_1 \\ u_1 & u_1 z \end{bmatrix}$ and $X_2 := \begin{bmatrix} u_2/z & u_2 \\ u_2 & u_2 z \end{bmatrix}$. *I presume the desired unique solution* is the non-simgular one without any division by variable z, in which case $u_1 := 1$ and $u_2 := 0$ leading to

$$\dot z = \tfrac{3}{2}\sigma^2 z + \sigma z\eta, \tag{3.4}$$

that is, an exponentially unstable drift with multiplicative fluctuations. (Do the fluctuations stabilise the drift?)

For comparison, my web service (Roberts 2009–2022) constructs the coordinate transform to XY-variables, where $y_1 = x_2$ and $w_1 = \eta$,

$$\begin{aligned} x_1 &= X_1 - \varepsilon X_1 Y_1 - \sigma\varepsilon \mathrm{e}^{-1t}\star w_1\, X_1 \quad + 1/2\varepsilon^2 X_1 Y_1^2 + \sigma\varepsilon^2 \mathrm{e}^{-1t}\star w_1\, X_1 Y_1 \\ &\qquad - 1/6\varepsilon^3 X_1 Y_1^3 - 1/2\sigma\varepsilon^3 \mathrm{e}^{-1t}\star w_1\, X_1 Y_1^2 + O\big(\varepsilon^4, \sigma^2\big) \\ y_1 &= Y_1 + \sigma \mathrm{e}^{-1t}\star w_1 \; + O\big(\varepsilon^4, \sigma^2\big) \end{aligned} \tag{3.5}$$

And in these new variables the (Stratonovich) state space evolution is

$$\begin{aligned}\dot X_1 &= \sigma\varepsilon w_1 X_1 + O\big(\varepsilon^5,\sigma^3\big)\\ \dot Y_1 &= -Y_1 + O\big(\varepsilon^5,\sigma^3\big)\end{aligned}\tag{3.6}$$

In the new XY-coordinate system: $Y_1 \to 0$ exponentially quickly; and throughout X_1 evolves as above via the nonlinear noise-noise interactions. Here the noise is multiplicative so we need to consider the Ito form in order to compare to (3.4) of Katzenberger (1991): applying the transform rule (e.g., Roberts 2015, Thm. 20.9) gives

$$dX_1 = +\tfrac{1}{2}\sigma^2 X_1\,dt + \sigma\varepsilon X_1\,dW + O\big(\sigma^3\big)\tag{3.7}$$

The drift differs from that of (3.4) which is apparently the prediction of Katzenberger (1991). (These can only be brought into line by choosing $u_1 = -1$ to give the weird result that $X_1 := \begin{bmatrix}-2/z & -1\\ -1 & -z\end{bmatrix}$.)

Question: *is there a mistake in my coding? in the interpretation of X_i? in Parsons and Rogers (2015) and Parsons and Rogers (2017)? or in Katzenberger (1991)?*

4 Katzenberger modelling

Dynamical variables are `x(i)`, and white noises `eta(i)`. For the systems, set the Ito interpretation which Parsons and Rogers (2015) and Parsons and Rogers (2017) address in their SDE form

$$\dot{\vec x} = \vec f(\vec x) + \epsilon\vec h(\vec x) + \sigma G(\vec x)\vec\eta(t).\tag{4.1}$$

```
operator x,eta;
f :=tp mat(( (if sys=2 then x(2)^2 else 0)
            +(if sys=3 then x(1)*x(2) else 0)
            , -x(2) ));
gg:=tp mat(( (if sys=1 then x(2) else 0) , 1 ));
h :=tp mat(( 0,0 ));
etav:=tp mat((eta(1)));
```

By design, the systems have critical manifold $x_2 = 0$ so define `man` for when we substitute this, parameterised by `z`.

```
man:={x(1)=>z,x(2)=>0};
```

Mostly, the following is general Get the dimensionality of the system, and define corresponding zero and identity matrices.

```
n:=part(length(f),1);
matrix Zero(n,n);
Id:=Zero$ for i:=1:n do Id(i,i):=1;
```

Form Jacobian of f, evaluated on the critical manifold, and its transpose.

```
tmp:=Zero$
for i:=1:n do for j:=1:n do tmp(i,j):=df(f(i,1),x(j));
Jac:=sub(man,tmp);
JacT:=tp Jac;
```

Since here eigenvalues of J are only 0 and -1, then "the pseudo-inverse" according to equation (18) of Parsons and Rogers (2015) and Parsons and Rogers (2017) is simply J itself:

```
JacPI:=Jac;
```

But (18) is only one possible pseudo-inverse because, for example, the definition of the Moore–Penrose pseudo-inverse uses the SVD ((18) and the Moore–Penrose are the same only for symmetric matrices). Perhaps some pseudo-inverse proportional to J^T would be better?

Useful proc to access a given element in an array of matrices

```
procedure el(a,i,j)$ a(i,j)$
```

Via the equation following (18) of Parsons and Rogers (2015) and Parsons and Rogers (2017), compute the Hessians evaluated on the critical manifold.

```
array Hes(n);
for i:=1:n do begin
  tmp:=Zero;
  for j:=1:n do for k:=1:n do
      tmp(j,k):=df(f(i,1),x(j),x(k));
  write Hes(i):=sub(man,tmp);
end;
```

Solve the Lyapunov equation (19) of Parsons and Rogers (2015) and Parsons and Rogers (2017)—*but there are multiple solutions* (I do not see that Parsons and Rogers (2015) and Parsons and Rogers (2017) identify what to do with the multiple solutions). For the moment parametrise them all with `u#`.

```
array xx(n);
operator u;
us:= for i:=1:n join for j:=1:n collect u(i,j)$
for l:=1:n do begin
  tmp:=Zero$
  for i:=1:n do for j:=1:n do tmp(i,j):=u(i,j);
  res19:=JacT*tmp+tmp*Jac+Hes(l);
  eqns:= for i:=1:n join for j:=1:n collect res19(i,j);
  soln:=( solve(eqns,us) where arbcomplex(~i)=>mkid(u,i) );
  write xx(l):=sub(soln,tmp);
end;
```

Check the general solution is OK.

```
for i:=1:n do if JacT*xx(i)+xx(i)*Jac+Hes(i) neq Zero
    then rederr("**** Failure in solving a Lyapunov system");
```

Compute the projection matrix $P =$ `pp` via equation (20) of Parsons and Rogers (2015) and Parsons and Rogers (2017):

```
pp:=Id-JacPI*Jac;
```

Then compute $Q =$ `qq` via equation (21) of Parsons and Rogers (2015) and Parsons and Rogers (2017), first overwriting `xx` to store complicated sub-expressions.

```
array qq(n),php(n);
for l:=1:n do begin
  php(l):=(tp pp)*Hes(l)*pp;
  xx(l):=xx(l)-(tp JacPI)*Hes(l)*pp-(tp pp)*Hes(l)*JacPI;
end;
```

Second, evaluate equation (21) to get matrix Q_i.

```
for i:=1:n do begin
  tmp:=Zero;
  for j:=1:n do for k:=1:n do
      tmp(j,k):=for l:=1:n sum
          -JacPI(i,l)*el(php(l),j,k)+pp(i,l)*el(xx(l),j,k);
  write qq(i):=tmp;
end;
```

Plug into formula (8) of Parsons and Rogers (2015) [p.8], equivalently into (7) of Parsons and Rogers (2017).

```
array gqg(n);
factor sigma,epsilon;
tmp:=0*f$
for i:=1:n do begin
  write gqg(i):=(tp gg)*qq(i)*gg;
  tmp(i,1):=trace(gqg(i));
end;
dzdtdrift:=sub(man, epsilon*pp*h+sigma^2/2*tmp );
dzdtnoise:=sub(man, sigma*pp*gg*etav );
```

Finish the reduce script.

```
end;%script
```

4.1 Algebra for system (1.1)

```
1: in_tex "katzComparison.tex"$
sys := 1

      [   0   ]
f := [        ]
      [ - x(2)]

       [x(2)]
gg := [    ]
       [ 1  ]

      [0]
h := [ ]
      [0]

etav := [eta(1)]

man := {x(1) => z,x(2) => 0}
n := 2

        [0   0 ]
jac := [      ]
        [0   -1]

         [0   0 ]
jact := [      ]
         [0   -1]

          [0   0 ]
jacpi := [      ]
          [0   -1]

           [0   0]
```

```
hes(1) := [    ]
          [0  0]

          [0  0]
hes(2) := [    ]
          [0  0]

         [u1  0]
xx(1) := [     ]
         [0   0]

         [u2  0]
xx(2) := [     ]
         [0   0]

      [1  0]
pp := [    ]
      [0  0]

         [u1  0]
qq(1) := [     ]
         [0   0]

         [0  0]
qq(2) := [    ]
         [0  0]

          [    2   ]
gqg(1) := [x(2) *u1]

gqg(2) := [0]

             [0]
dzdtdrift := [ ]
             [0]
```

```

             [0]
dzdtnoise := [ ]
             [0]
```

4.2 Algebra for system (2.1)

```
1: in_tex "katzComparison.tex"$
sys := 2

     [     2 ]
     [ x(2)  ]
f := [       ]
     [ - x(2)]

      [0]
gg := [ ]
      [1]

     [0]
h := [ ]
     [0]

etav := [eta(1)]

man := {x(1) => z,x(2) => 0}
n := 2

       [0  0 ]
jac := [     ]
       [0  -1]

        [0  0 ]
jact := [     ]
        [0  -1]
```

```

              [0   0 ]
jacpi := [         ]
              [0   -1]

               [0   0]
hes(1) := [       ]
               [0   2]

               [0   0]
hes(2) := [       ]
               [0   0]

              [u1   0]
xx(1) := [         ]
              [0    1]

              [u2   0]
xx(2) := [         ]
              [0    0]

           [1   0]
pp := [       ]
           [0   0]

              [u1   0]
qq(1) := [         ]
              [0    1]

              [0   0]
qq(2) := [       ]
              [0   0]

gqg(1) := [1]

```

```
gqg(2) := [0]

                [       2 ]
                [ sigma   ]
                [---------]
dzdtdrift :=    [    2    ]
                [         ]
                [    0    ]

                [0]
dzdtnoise :=    [ ]
                [0]
```

4.3 Algebra for system (3.1)

```
1: in_tex "katzComparison.tex"$
sys := 3

     [x(2)*x(1)]
f := [         ]
     [  - x(2) ]

      [0]
gg := [ ]
      [1]

     [0]
h := [ ]
     [0]

etav := [eta(1)]

man := {x(1) => z,x(2) => 0}
n := 2

```

```
       [0  z ]
jac := [     ]
       [0  -1]

        [0  0 ]
jact := [     ]
        [z  -1]

         [0  z ]
jacpi := [     ]
         [0  -1]

          [0  1]
hes(1) := [    ]
          [1  0]

          [0  0]
hes(2) := [    ]
          [0  0]

         [ u1 - 1       ]
         [--------   u1 ]
xx(1) := [   z          ]
         [              ]
         [   u1      u1*z]

         [ u2       ]
         [----   u2 ]
xx(2) := [ z        ]
         [          ]
         [ u2   u2*z]

      [1  z]
pp := [    ]
      [0  0]
```

```

             [ u1 + u2*z - 1                       ]
             [---------------    u1 + u2*z + 1   ]
qq(1) := [        z                               ]
             [                                      ]
             [ u1 + u2*z + 1   z*(u1 + u2*z + 2)]

             [0  0]
qq(2) := [     ]
             [0  0]

gqg(1) := [z*(u1 + u2*z + 2)]

gqg(2) := [0]

                 [      2                     ]
                 [ sigma *z*(u1 + u2*z + 2) ]
                 [--------------------------]
dzdtdrift := [            2               ]
                 [                            ]
                 [            0               ]

                 [sigma*eta(1)*z]
dzdtnoise := [              ]
                 [      0       ]
```

References

Boxler, P. (1989). "A Stochastic Version Of The Centre Manifold Theorem". In: *Probab. Th. Rel. Fields* 83, pp. 509–545. DOI: 10.1007/BF01845701 (cit. on p. 2).

Chao, Xu and A. J. Roberts (1996). "On the low-dimensional modelling of Stratonovich stochastic differential equations". In: *Physica A* 225, pp. 62–80. DOI: 10.1016/0378-4371(95)00387-8 (cit. on pp. 2–5).

Hearn, Anthony C. and Rainer Schöpf (2018). *Reduce User's Manual*. Tech. rep. http://reduce-algebra.sourceforge.net/ (cit. on p. 2).

Katzenberger, G.S. (1991). "Solutions of a stochastic differential equation forced onto a manifold by a large drift". In: *Annals of Probability* 19.4, pp. 1587–1628. DOI: 10.1214/aop/1176990225 (cit. on pp. 1–5, 7).

Parsons, Todd L. and Tim Rogers (2015). *Dimension reduction via timescale separation in stochastic dynamical systems*. Tech. rep. [http://arXiv.org/abs/1510.07031] (cit. on pp. 2, 3, 7–10).

– (Sept. 2017). "Dimension reduction for stochastic dynamical systems forced onto a manifold by large drift: a constructive approach with examples from theoretical biology". In: *Journal of Physics A: Mathematical and Theoretical* 50.41, pp. 1–33. ISSN: 1751-8121. DOI: 10.1088/1751-8121/aa86c7 (cit. on pp. 2, 3, 7–10).

Roberts, A. J. (2006). "Resolving the multitude of microscale interactions accurately models stochastic partial differential equations". In: *LMS J. Computation and Maths* 9, pp. 193–221. DOI: 10.1112/S146115700000125X (cit. on pp. 2, 4, 5).

– (2008). "Normal form transforms separate slow and fast modes in stochastic dynamical systems". In: *Physica A* 387, pp. 12–38. DOI: 10.1016/j.physa.2007.08.023 (cit. on p. 2).

– (2009–2022). *Derive invariant manifolds and/or normal forms of general stochastic or non-autonomous, multiscale, differential equations*. Tech. rep. http://tuck.adelaide.edu.au/sdenf.php [December 23, 2022] (cit. on pp. 2, 4–6).

– (2015). *Model emergent dynamics in complex systems*. SIAM, Philadelphia. ISBN: 9781611973556. http://bookstore.siam.org/mm20/ (cit. on p. 7).