

# Lecture 1: Stirring & Mixing

Stirring: mechanical action (cause)  
 Mixing: homogenization of a scalar (effect)

$\theta(\underline{x}, t)$  = concentration,  $\underline{u}(\underline{x}, t)$  given

Advection-Diffusion eq.  $\frac{\partial \theta}{\partial t} + \underline{u} \cdot \nabla \theta = \kappa \nabla^2 \theta$ ,  $\nabla \cdot \underline{u} = 0$  in  $\Omega$

(AD) Boundary conditions:  $\hat{n} \cdot \nabla \theta = 0$   
 $\hat{n} \cdot \underline{u} = 0$  } on boundary  $\partial \Omega$

Let  $\langle \cdot \rangle = \frac{1}{|\Omega|} \int_{\Omega} \cdot dV$   $|\Omega|$  = volume of  $\Omega$   
 "volume average"

Multiply (AD) by  $\theta^{m-1}$ , integrate:

$$\langle \theta^{m-1} \partial_t \theta \rangle = \partial_t \langle \theta^m \rangle$$

$$\langle \theta^{m-1} \underline{u} \cdot \nabla \theta \rangle = \langle \underline{u} \cdot \nabla \theta^m \rangle = \langle \nabla \cdot (\underline{u} \theta^m) \rangle$$

$$= \int_{\partial \Omega} \theta^m \underbrace{\underline{u} \cdot \hat{n}}_0 dS = 0$$

$\nabla \cdot \underline{u} = 0$

$$\langle \theta^{m-1} \kappa \nabla^2 \theta \rangle = \kappa m \langle \nabla \cdot (\theta^{m-1} \nabla \theta) - \nabla \theta^{m-1} \cdot \nabla \theta \rangle$$

$$= \kappa m \int_{\partial \Omega} \theta^{m-1} \underbrace{\nabla \theta \cdot \hat{n}}_0 dS - \kappa m(m-1) \langle \theta^{m-2} |\nabla \theta|^2 \rangle$$

$$\partial_t \langle \theta^m \rangle = -\kappa m(m-1) \langle \theta^{m-2} |\nabla \theta|^2 \rangle$$

$m=0$  is trivial

$m=1$ :  $\partial_t \langle \theta \rangle = 0$  Total amount of  $\theta$  is conserved

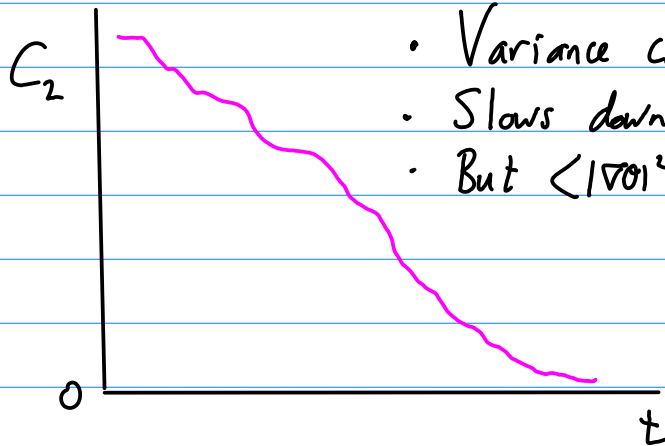
$m=2$ :  $\partial_t \langle \theta^2 \rangle = -2\kappa \langle |\nabla \theta|^2 \rangle$   $\langle \theta^2 \rangle$  non-increasing!

Let variance  $\text{Var} = C_2 = \langle \theta^2 \rangle - \langle \theta \rangle^2$

$$\partial_t C_2 = -2\kappa \langle |\nabla \theta|^2 \rangle$$

↑  
constant

Scenario:



- Variance can only decrease.
- Slows down as  $\langle |\nabla \theta|^2 \rangle \rightarrow 0$
- But  $\langle |\nabla \theta|^2 \rangle = 0$  iff  $\theta = \text{constant}$ .

↑  
in some sense

Hence the system is "driven" towards a homogeneous state where

$$\theta(\underline{x}, t) = \langle \theta \rangle = \text{constant}. \quad (C_2 = 0, \langle \theta^2 \rangle = \langle \theta \rangle^2)$$

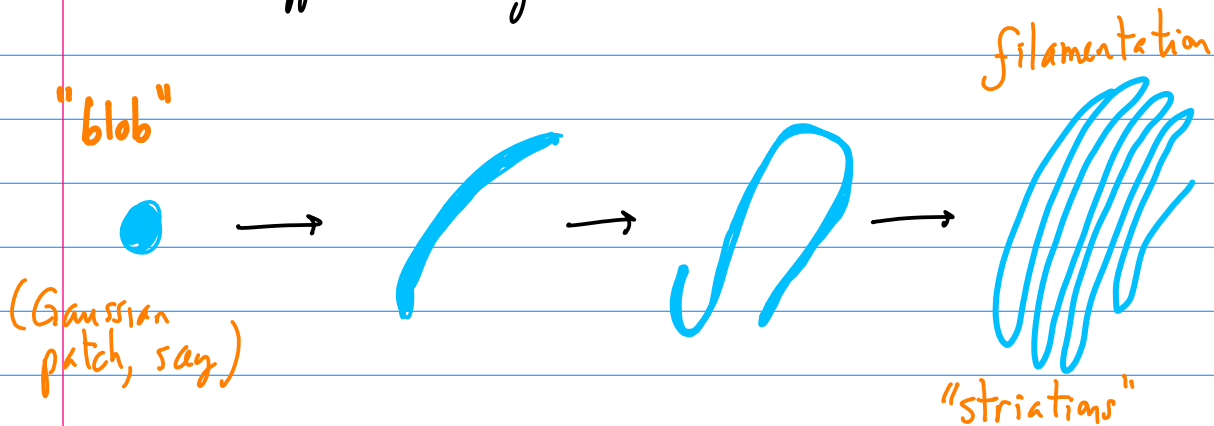
Assume  
 $\langle \theta \rangle = 0$   
WLOG

No fluctuations from the mean! When  $C_2$  is small "enough", we say the system is mixed.

Big Q: Where is  $\underline{u}(\underline{x}, t)$ ! ? (stirring)

It doesn't appear in the variance equation!

But of course the variance equation is not closed: it depends on  $\nabla\theta$ .  
What happens when you stir?



This hints at the answer: stirring increases  $\nabla\theta$

$$\partial_t \langle \theta^2 \rangle = -2\kappa \langle |\nabla\theta|^2 \rangle$$

*this becomes larger as we stir*

By how much are gradients increased? After all, if  $|\nabla\theta|$  becomes too large, then  $\langle \theta^2 \rangle \rightarrow 0$ , so there are no gradients anymore!

Answer: for "good" stirring, the system is driven to a state where

$$\kappa \langle |\nabla\theta|^2 \rangle \rightarrow \text{independent of } \kappa$$

Hence,  $\nabla\theta \sim \kappa^{-1/2}$

This is the chaotic/turbulent mixing scenario:

$$\frac{\partial \langle \theta^2 \rangle}{\partial t} \text{ becomes independent of } \kappa \text{ after a "short" transient}$$

*(How short? Typically  $\sim \log \kappa$ )*

*This is the Platonic ideal of mixing*

Furthermore, the smallest scales visible in the concentration field  $\theta(x, t)$  have size  $\sim \sqrt{\kappa}$ . (missing a dimensional factor  $\rightarrow$  see later)

Note that  $\partial_t \langle \theta^2 \rangle$  independent of  $\kappa$  is crucial: in most applications,  $\kappa$  is tiny!

Heat:  $\kappa = 2.2160 \times 10^{-5} \text{ m}^2/\text{s}$  at 300K

10 m room: diffusion time  $\sim \frac{L^2}{\kappa} = \frac{(10\text{m})^2}{(2 \times 10^{-5} \text{ m}^2/\text{s})} \sim 4.5 \times 10^6 \text{ sec}$

$\sim 1300 \text{ hours}$

So we better stir!  
Even thermal convection  $\leftarrow$   
is often enough.

$\sim 53 \text{ days!}$



## Lecture 2: Filament model

Last time:  $\frac{\partial \theta}{\partial t} + u \cdot \nabla \theta = \kappa \nabla^2 \theta$ ,  $\nabla \cdot u = 0$   
(AD)

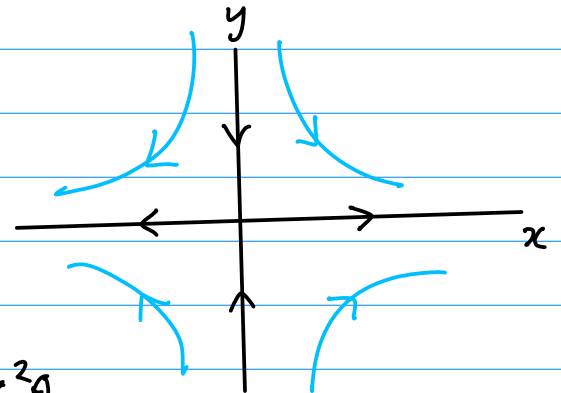
$u$  is stirring (advection)  
 $\kappa$  is mixing (diffusion) ← small at first

Let's look at a simple exact solution that illustrates important features.

Example of a good mixer:

$$\underline{u}(x, t) = (\lambda x, -\lambda y)$$

"hyperbolic point"



AD:  $\partial_t \theta + \lambda x \partial_x \theta - \lambda y \partial_y \theta = \kappa \nabla^2 \theta$

Can solve this exactly (we'll say more next time), but let's do the simplest thing: look for an  $x$ -independent solution of the form:

$$\theta(x, t) = e^{-\lambda t} f(y)$$

$$-\lambda f - \lambda y f' = \kappa f''$$

Boundary condition:

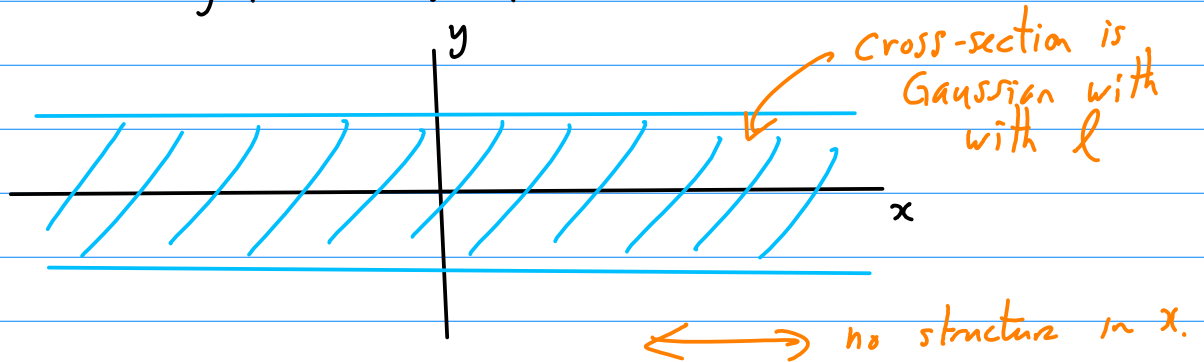
$$f \rightarrow 0 \text{ as } y \rightarrow \pm \infty.$$

Solution is:  $f(y) = e^{-y^2/2l^2}$ , where  $l^2 = \frac{\kappa}{\lambda}$

Hence,

$$\theta(x, t) \sim e^{-\lambda t} e^{-y^2/2l^2}$$

This is the "filament" solution:



In fact, this solution tells us about the ultimate state of any compactly-supported initial condition:

"blob"



"filament"



"intensity fading" as  $e^{-\lambda t}$

central part

$\sim$  Gaussian cross-section

For this case, we know the length scale of "striations";

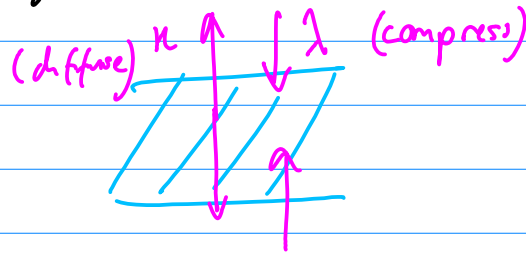
$$l = \sqrt{\frac{\kappa}{\lambda}}$$

Batchelor length


Note  $l \sim \sqrt{\kappa}$ , as necessary to make decay rate indep. of  $\kappa$ !

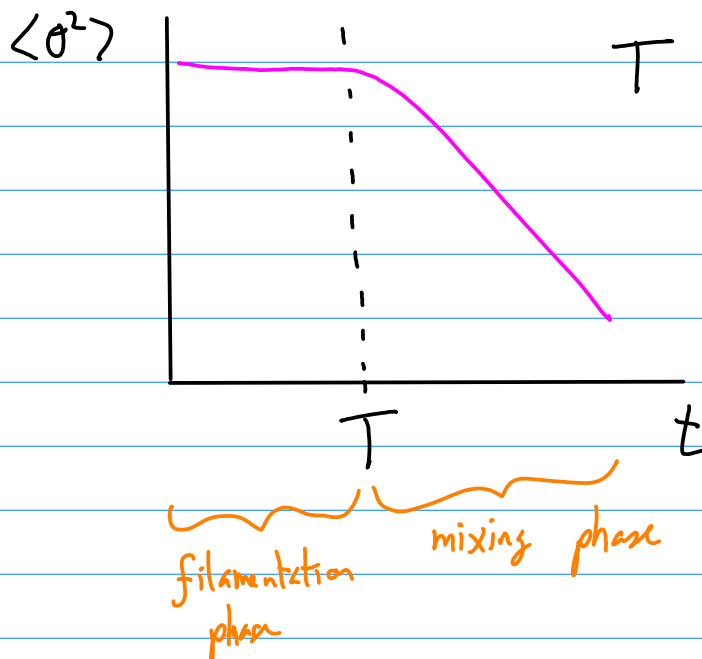
In practical applications,  $\lambda$  is often taken to be the local rate of strain.

$l$  is set by a balance between compression and diffusion



Summary: how mixing proceeds

- A blob is stirred 
- For a while,  $\langle \theta^2 \rangle$  is  $\sim$  constant, since  $\kappa$  is small
- When  $\nabla \theta$  reaches scales of order  $l$ , diffusion takes over
- After that,  $\langle \theta^2 \rangle$  decays at a  $\kappa$ -independent rate



$$T \text{ given by: } e^{-1} T \sim \sqrt{\kappa}$$

$$\underline{T \sim \lambda^{-1} \log \kappa}$$

A more general approach:

$$\frac{\partial \theta}{\partial t} + u \cdot \nabla \theta = k \Delta \theta$$

Assume smooth  $u(x, t)$ , choose a reference trajectory:

$$x_0(t), \quad \dot{x}_0 = u(x_0(t), t)$$

Now let  $x = x_0(t) + r$ .

Change coordinates from  $x$  to  $r$ .

$$\theta(x, t) = \tilde{\theta}(r, t)$$

$$\frac{\partial \theta}{\partial t} = \frac{\partial}{\partial t} \Big|_x \theta(x, t) = \frac{\partial}{\partial t} \Big|_x \tilde{\theta}(r, t)$$

$$\overset{\text{constant } r}{=} = \frac{\partial}{\partial t} \Big|_r \tilde{\theta}(r, t) + \underbrace{\frac{\partial r}{\partial t} \Big|_x}_{-\dot{x}_0} \cdot \nabla_r \tilde{\theta}$$

Als.,  $u \cdot \nabla \theta = u \cdot \nabla_r \theta$ , so  $-\dot{x}_0 = -u(x_0, t)$

$$\frac{\partial \theta}{\partial t} \Big|_x + u \cdot \nabla \theta = \left( \frac{\partial \tilde{\theta}}{\partial t} \Big|_r - u(x_0, t) \cdot \nabla_r \tilde{\theta} \right) + u(x, t) \cdot \nabla_r \tilde{\theta}$$

$$= \frac{\partial \tilde{\theta}}{\partial t} + \underbrace{\{u(x_0 + r, t) - u(x_0, t)\}}_{\text{velocity difference between } x \text{ and } x_0} \cdot \nabla_r \tilde{\theta}$$

For smooth  $u$ ,

$$u(x_0 + r, t) - u(x_0, t) = r \cdot \nabla u(x_0, t) + O(r^2)$$

$$\text{So } \frac{\partial \tilde{\theta}}{\partial t} + r \cdot \nabla u(x_0, t) \cdot \nabla_r \tilde{\theta} = \kappa \Delta_r \tilde{\theta} + O(r^2)$$

This is an advection-diffusion equation valid in the neighborhood of a fluid trajectory  $x_0(t)$ .

Drop  $\sim$  and  $o$ 's:

$$\frac{\partial \theta}{\partial t} + x \cdot A^T(t) \cdot \nabla \theta = \kappa \Delta \theta$$

AD for  
linearized  
velocity  
field

where  $A^T(t) = \nabla u(x_0(t), t)$  is the velocity gradient matrix near  $x_0(t)$ .

Note that  $\nabla \cdot u = 0$  implies

$$\nabla \cdot (x \cdot A^T) = \sum_{ij} \partial_i (x_j A_{ij}) = \sum_i A_{ii} = \text{tr } A$$

$$\text{trace } A = 0$$

Next we will solve this in general.

## Lecture 3: Lagrangian Coordinates

Goal: show convergence to "filament solution"

$$\frac{\partial \theta}{\partial t} + u \cdot \nabla \theta = \kappa \Delta \theta$$

(last time)  
 $\frac{\partial x}{\partial t} \Big|_X$

Particle trajectories:  $\dot{x} = u(x(t), t)$ ,  $x(0) = X$

Use the  $X$  themselves as coordinate.

$x$  — Eulerian or spatial coordinates

$X$  — Lagrangian or material coordinates

$$\theta(x, t) = \Theta(X, t)$$

$$\frac{\partial}{\partial t} \Big|_X \Theta(X, t) = \frac{\partial}{\partial t} \Big|_X \theta(x, t) = \frac{\partial}{\partial t} \Big|_x \theta(x, t) + \underbrace{\nabla \theta \cdot \frac{\partial x}{\partial t}}_{u(x, t)} \Big|_X$$

$$= \frac{\partial \theta}{\partial t} \Big|_x + u \cdot \nabla \theta$$

Hence,  $\frac{\partial \Theta}{\partial t} \Big|_X = \kappa \Delta \Theta$

write  $x$ , not  $X$ !

So now transform  $\Delta \circlearrowleft$  to  $\Delta_X \circlearrowleft$ .

Indices:  $i, j, h, l$  Eulerian,  $p, q, r, s$  Lagrangian

$$\frac{\partial \circlearrowleft}{\partial x^i} = \frac{\partial \circlearrowleft}{\partial X^p} \frac{\partial X^p}{\partial x^i} \quad \text{Einstein sum convention (for Benodok)}$$

Define the vectors:  $(e^p)_i := \frac{\partial X^p}{\partial x^i}$ .

$$(e_q)_i := \frac{\partial x^i}{\partial X^q}$$

Note  $e^p \cdot e_q = \frac{\partial X^p}{\partial x^i} \frac{\partial x^i}{\partial X^q} = \delta^p_q$

Define also the metric tensor  $g_{pq} := e_p \cdot e_q$

and its inverse  $g^{pq} = e^p \cdot e^q$

Check:  $g^{pq} g_{qr} = \underbrace{e^p \cdot e^q}_{\text{II}} e_q \cdot e_r = e^p \cdot \text{II} \cdot e_q = e^p \cdot e_q = \delta^p_q$

Finally, let  $g := \det [g_{pq}]$

Cauchy-Green deformation tensor  
right

left

$$\text{Claim: } \Delta^{\oplus} = \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^p} \left( \sqrt{g} g^{pq} \frac{\partial \oplus}{\partial x^q} \right)$$

$$\begin{aligned} \Delta^{\oplus} &= \frac{\partial}{\partial x^i} \left( \frac{\partial \oplus}{\partial x^i} \right) = \frac{\partial}{\partial x^p} \left( \underbrace{\frac{\partial x^q}{\partial x^i}}_{(e^q)_i} \frac{\partial \oplus}{\partial x^q} \right) \frac{\partial x^p}{\partial x^i} \\ &= \frac{\partial}{\partial x^p} \left( \underbrace{e^p \cdot e^q}_{g^{pq}} \frac{\partial \oplus}{\partial x^q} \right) - \frac{\partial \oplus}{\partial x^q} \underbrace{e^q \cdot \frac{\partial e^p}{\partial x^p}}_{\boxed{\phantom{e^q \cdot \frac{\partial e^p}{\partial x^p}}}} \end{aligned}$$

The Christoffel symbols (or connections) are

$$\Gamma_{pq}^r = e^r \cdot \frac{\partial e_p}{\partial x^q}$$

$$\left( = e^r \cdot \frac{\partial^2 x}{\partial x^q \partial x^p} \right)$$

$$\text{Since } e^r \cdot e_p = \delta_{pq}^r,$$

$$\text{so } \Gamma_{pq}^r = \Gamma_{qp}^r$$

$$\Gamma_{pq}^r = -e_p \cdot \frac{\partial e^r}{\partial x^q}$$

So the boxed term above is

$$\begin{aligned} e^q \cdot \frac{\partial e^p}{\partial x^r} &= e^q \cdot \mathbb{I} \cdot \frac{\partial e^p}{\partial x^r} = \underbrace{e^q \cdot e^s}_{g^{qs}} e_s \cdot \frac{\partial e^p}{\partial x^r} \\ &= -g^{qs} \Gamma_{sp}^p \end{aligned}$$



Claim:  $\Gamma_{sp}^r = \frac{\partial}{\partial X^s} \log \sqrt{g}$  ↙ metric determinant

First, derive well-known form for  $\Gamma$ :

$$\Gamma_{pq}^r = e^r \cdot \frac{\partial e_p}{\partial X^q} = e^r \cdot e^s e_s \cdot \frac{\partial e_p}{\partial X^q}$$

$$= g^{rs} e_s \cdot \frac{\partial e_p}{\partial X^q}$$

since  $\frac{\partial e_p}{\partial X^q} = \frac{\partial e_p}{\partial X^r}$

$$= \frac{1}{2} g^{rs} \left( e_s \cdot \frac{\partial e_p}{\partial X^q} + e_s \cdot \frac{\partial e_q}{\partial X^r} \right)$$

$$= \frac{1}{2} g^{rs} \left( \frac{\partial}{\partial X^q} (e_s \cdot e_p) + \frac{\partial}{\partial X^r} (e_s \cdot e_q) - e_p \cdot \frac{\partial e_s}{\partial X^q} - e_q \cdot \frac{\partial e_s}{\partial X^r} \right)$$

↖  $\frac{\partial e_q}{\partial X^s}$  ↗  $\frac{\partial e_p}{\partial X^s}$

$$= \frac{1}{2} g^{rs} \left( \frac{\partial g_{sp}}{\partial X^q} + \frac{\partial g_{sq}}{\partial X^r} - \frac{\partial g_{pq}}{\partial X^s} \right)$$

$$\Gamma_{pq}^r = \frac{1}{2} g^{rs} \left( \frac{\partial g_{sp}}{\partial X^q} + \frac{\partial g_{sq}}{\partial X^r} - \frac{\partial g_{pq}}{\partial X^s} \right)$$

Now contract  $r$  and  $q$ :

$$\Gamma_{pr}^r = \frac{1}{2} g^{rs} \left( \cancel{\frac{\partial g_{sr}}{\partial x^r}} + \frac{\partial g_{sr}}{\partial x^p} - \cancel{\frac{\partial g_{pr}}{\partial x^s}} \right)$$

these cancel after relabeling

$$= \frac{1}{2} g^{rs} \frac{\partial g_{rs}}{\partial x^p}$$

Great! Now to show that this equals  $\log \sqrt{g}$ .

The determinant can be written

$$g = g_{rs} C^{rs} \quad (\text{no sum over } r)$$

where  $C^{rs}$  is the cofactor matrix of  $g_{rs}$ .

$$\frac{\partial g}{\partial g_{rs}} = C^{rs}, \quad \text{since the cofactor } C^{rs} \text{ does not contain } g_{rs}.$$

$$\text{The inverse of } [g_{rs}] \text{ is } g^{rs} = \frac{C^{rs}}{g} = \frac{1}{g} \frac{\partial g}{\partial g_{rs}}$$

Hence,

$$\Gamma_{pr}^r = \frac{1}{2} g^{rs} \frac{\partial g_{rs}}{\partial x^p} = \frac{1}{2g} \frac{\partial g}{\partial g_{rs}} \frac{\partial g_{rs}}{\partial x^p} = \frac{1}{2g} \frac{\partial g}{\partial x^p}$$

$$\therefore \Gamma_{pr}^r = \frac{\partial}{\partial x^p} \log \sqrt{g}.$$

So now back to our Laplacian:

$$\begin{aligned} \Delta \Phi &= \frac{\partial}{\partial x^p} \left( g^{pq} \frac{\partial \Phi}{\partial x^q} \right) + \frac{\partial \Phi}{\partial x^q} g^{pq} \frac{\partial \log \sqrt{g}}{\partial x^p} \\ &= \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^p} \left( \sqrt{g} g^{pq} \frac{\partial \Phi}{\partial x^q} \right) \end{aligned}$$

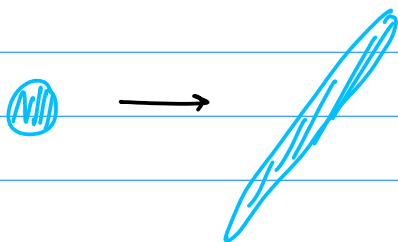
However,  $g = 1$  when  $\nabla \cdot u = 0$  (show this soon),  
so we get

$$\frac{\partial \Phi}{\partial t} = \kappa \frac{\partial}{\partial x^p} \left( g^{pq} \frac{\partial \Phi}{\partial x^q} \right)$$

AD eq'n  
in Lagrangian  
coordinates

This holds for any differentiable velocity field!

Seems very simple, but  $g^{pq}(X, t)$  can be atrocious!



$g^{pq}$  records how a sphere  
is locally deformed to an  
ellipsoid.

## Lecture 4: Solving the Lagrangian equation

Last time: convert the AD eq'n to Lagrangian coordinates:

$$\partial_t \Theta = \kappa \nabla_x \cdot (g^{-1} \nabla_x \Theta)$$

where  $\dot{x} = u(x, t)$ ,  $x(0) = X$

For instance, start with linearized velocity equation:  
("Filament" lecture)  $\rightarrow$  transpose

$$\partial_t \theta + x \cdot A^T(t) \cdot \nabla_x \theta = \kappa \Delta_x \theta$$

with  $u(x) = x \cdot A^T(t)$ ,  $\text{trace } A = 0$ .

Trajectories satisfy:  $\dot{x} = x \cdot A^T(t)$ ,  $x(0) = X$ ,

Try the solution: (or  $\dot{x} = A(t) \cdot x$ )

$$x(t) = \exp\left(\int_0^t A(\tau) d\tau\right) \cdot X$$

The matrix exponential is defined from its Taylor series,

$$e^B = I + B + \frac{1}{2} B^2 + \dots$$

Check: with  $B(t) = \int_0^t A(\tau) d\tau$ ,  $\dot{B} = A(t)$ ,

$$\begin{aligned} \frac{d}{dt} \exp B(t) &= \frac{d}{dt} \left( I + B + \frac{1}{2} B^2 + \dots \right) \\ &= \dot{B} + \frac{1}{2} (\dot{B}B + B\dot{B}) + \dots \end{aligned}$$

If the commutator

$$[B, \dot{B}] = \int_0^t [A(t), A(\tau)] d\tau$$

vanishes  $\forall t \geq 0$ , i.e.,

$$(1) \quad [A(t_1), A(t_2)] = 0, \quad \forall t_1, t_2 \geq 0$$

then we're good, and

$$\frac{d}{dt} \exp B(t) = e^B \cdot \dot{B} = e^B \cdot A$$

$$x^i = (e^B)^i_j x^j$$

so the solution is indeed  $x = e^{B(t)} \cdot X$ .

Also we then have

$$X = e^{-B(t)} \cdot x$$

With indices:  $X^p = [e^{-B}]^p_i x^i$

← rows

↖ columns

This immediately gives the deformation gradient,

$$\frac{\partial X}{\partial x} = e^{-B(t)}, \quad \left( \frac{\partial X^p}{\partial x^i} = [e^{-B}]^p_i \right)$$

and the metric  $g^{pq} = [e^{-B}]^p_i [e^{-B}]^q_i$

$$[e^{-B}]^p_i [e^B]^i_q = \delta^p_q \quad \text{or} \quad [g^{-1}]^{pq} = ([e^{-B}][e^{-B}]^T)^{pq} \\ = [e^{-B} \ e^{-B^T}]^{pq}$$

It is not a given that

$$e^{-B} e^{-B^T} = e^{-(B+B^T)}$$

This requires  $[B(t), B^T(t)] = 0$

The matrix  $B$  is then called normal.

$$[B(t), B^T(t)] = \int_0^t [A(\tau), A^T(\sigma)] d\tau d\sigma,$$

$$(2) \text{ so } [A(t_1), A^T(t_2)] = 0 \quad \forall t_1, t_2 \geq 0$$

$\Rightarrow B(t)$  normal.

In general we assume (1) (so we can use the matrix exponential), but not (2).

Progress is made by using the singular value decomposition (SVD) of  $e^B$ .

$$e^B = UDV^T, \quad U, V \text{ orthogonal}$$

$$e^{-B} = UD^{-1}V^T, \quad D \text{ diagonal } > 0$$

$$[g_{pq}] = e^{B^T} e^B = VDU^T UDV^T = VD^2V^T$$

$$[g^{pq}] = VD^{-2}V^T \quad D = \begin{pmatrix} \Lambda_1 & & \\ & \ddots & \\ & & \Lambda_n \end{pmatrix}$$

We can also write this in dyad notation:

$$g^{pq} = \sum_{\sigma} V^p_{\sigma} \Lambda_{\sigma}^{-2} V^q_{\sigma}$$

$$= \Lambda_u^{-2} u^p u^q + \Lambda_m^{-2} m^p m^q + \Lambda_s^{-2} s^p s^q$$

where  $\Lambda_u \geq \Lambda_m \geq \Lambda_s$  (drop  $\Lambda_m$  in 2D)

Note that  $\det \sqrt{g} = \Lambda_u \Lambda_m \Lambda_s = 1$

In 2D,  $\Lambda_u = 1, \Lambda_s = \Lambda_u^{-1}$ .

We call the directions  $u$  *unstable*  
 $m$  *middle*  
 $s$  *stable*

Insert into the AD equation:

$$\begin{aligned} \partial_t \Theta &= \kappa \nabla_X \cdot (g^{-1} \nabla_X \Theta) \\ &= \kappa \nabla_X \cdot \left( (\Lambda_u^{-2} uu + \Lambda_m^{-2} mm + \Lambda_s^{-2} ss) \cdot \nabla_X \Theta \right) \\ &= \left[ \Lambda_u^{-2} (u \cdot \nabla_X)^2 + \Lambda_m^{-2} (m \cdot \nabla_X)^2 + \Lambda_s^{-2} (s \cdot \nabla_X)^2 \right] \Theta \end{aligned}$$

Use the orthogonal directions  $u, s$  to define coordinates  $(X, Y)$

$$u \cdot \nabla_X = \frac{\partial}{\partial X}, \quad s \cdot \nabla_X = \frac{\partial}{\partial Y}$$

In the most typical case,  $\Lambda_u$  and  $\Lambda_s$  are exponential in time.

$$\Lambda_s = e^{\lambda_s t} = e^{-\lambda t}, \quad \Lambda_u = e^{\lambda_u t} = e^{\lambda t}$$

(2D) (2D)



So we can make the approximation

$$\partial_t \Theta = n e^{2|\lambda_s|t} \frac{\partial^2 \Theta}{\partial Y^2} + O(e^{-2\lambda_m t})$$

$t \rightarrow \infty$

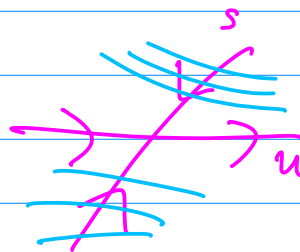
The equation has become one-dimensional

Now we can solve it exactly!

Note that diffusion starts to play a role when

$$n e^{2|\lambda_s|t} = 1 \Rightarrow t = \frac{1}{|\lambda_s|} \log n^{-1}$$

as claimed before.



gradients are amplified in the contracting direction.

Note: the " $t \rightarrow \infty$ " above should be taken with a grain of salt. It will often be the case that  $\partial_Y \Theta \rightarrow 0$  rapidly, allowing the other terms to take over.

## Lecture 5: Solving the Lagrangian equation (part 2)

Last time:  $e^B = UDV^T$  or  $[e^B]_{ij} = U_{i\sigma} D_{\sigma\sigma} V_{\sigma j}$

$$\frac{\partial \mathbb{H}}{\partial t} = \kappa e^{2|\lambda_s|t} \frac{\partial^2 \mathbb{H}}{\partial Y^2} + O(e^{-2|\lambda_m|t}), \quad t \rightarrow \infty$$

$\lambda_s < 0$

Initial condition:  $\mathbb{H}(\underline{X}, 0) = \mathbb{H}_0(\underline{X})$

$$\underline{x} = e^B \cdot \underline{X} = UDV^T \cdot \underline{X}$$

$$(U^T \cdot \underline{x}) = D(V^T \cdot \underline{X}) \leftarrow \text{two frames: } U^T \text{ \& } V^T$$

So let  $\underline{x} = U \cdot \tilde{\underline{x}}$ ,  $\underline{X} = V \cdot \tilde{\underline{X}}$

The  $\sim$  coordinates are now aligned with the singular eigenvectors:

$$\tilde{\underline{x}} = D \tilde{\underline{X}} \quad (\text{drop tildes})$$

Redefine  $(x, y, z)$  and  $(X, Y, Z)$  to be aligned with  $\underline{u}, \underline{s}, \underline{m}$ , the unstable ( $\lambda > 0$ ), stable, and middle directions

(Note: since  $\lambda_u + \lambda_m + \lambda_s = 0$ ,  $\lambda_u \geq \lambda_m \geq \lambda_s$ , we know  $\lambda_u > 0$  and  $\lambda_s < 0$ .)

The coordinates agree at  $t=0$ , and we have

$$\theta_0(\underline{x}) = \Theta_0(\underline{X}).$$

with  $x = X e^{\lambda_u t}$ ,  $y = Y e^{-|\lambda_s| t}$ ,  $z = Z e^{\lambda_m t}$

To solve the advection equation above, define a new time:

$$\kappa e^{2|\lambda_s| t} dt = dT$$

$$T = \frac{\kappa}{2|\lambda_s|} (e^{2|\lambda_s| t} - 1)$$

$$\Rightarrow \frac{\partial \Theta}{\partial T} = \frac{\partial^2 \Theta}{\partial Y^2} + O(e^{-2(\lambda_m + |\lambda_s|)t})$$

Let's assume  $\lambda_m + |\lambda_s| > 0$ . Then we can neglect the  $O(\cdot)$  term for  $t$  large, and solve

$$\partial_T \Theta \simeq \partial_Y^2 \Theta.$$

This of course is a simple heat equation, with Green's function

$$G(Y, T; Y_0, 0) = \frac{1}{\sqrt{4\pi T}} e^{-\frac{(Y - Y_0)^2}{4T}}$$

The full solution is then

$$\Theta(X, Y, Z, T) = \int_{-\infty}^{\infty} \Theta_0(X, Y_0, Z) \frac{e^{-(Y-Y_0)^2/4T}}{\sqrt{4\pi T}} dY_0$$

$$T = \frac{\kappa}{2|\lambda_s|} (e^{2|\lambda_s|T} - 1) = \frac{1}{2} l^2 e^{2|\lambda_s|t} + (\text{neglect})$$

↳ Batchelor length  $l^2 = \kappa/|\lambda_s|$

Recall that  $\theta(x, y, z, t) = \Theta(X, Y, Z, t)$ , so

$$\theta(x, y, z, t) = \frac{1}{\sqrt{2\pi} l} e^{-|\lambda_s|t} \int_{-\infty}^{\infty} \theta_0(x e^{-\lambda_u t}, Y_0, z e^{-\lambda_m t}) \times \exp \left\{ -\frac{(y e^{|\lambda_s|t} - Y_0)^2}{2l^2 e^{2|\lambda_s|t}} \right\} dY_0$$

We evaluate the LHS at fixed  $x, y, z$ ,  $t \rightarrow \infty$ .

Hence,  $x e^{-\lambda_u t} \rightarrow 0$ .

Now assume  $\theta_0$  has compact support. Then  $\theta_0(\cdot, Y_0, \cdot)$  vanishes for  $|Y_0|$  large enough.

In that case, for  $t$  large enough, ok to replace

$$\exp \left\{ -\frac{(y e^{|\lambda_s|t} - Y_0)^2}{2l^2 e^{2|\lambda_s|t}} \right\} \text{ by } \exp \left\{ -\frac{(y e^{|\lambda_s|t})^2}{2l^2 e^{2|\lambda_s|t}} \right\}$$

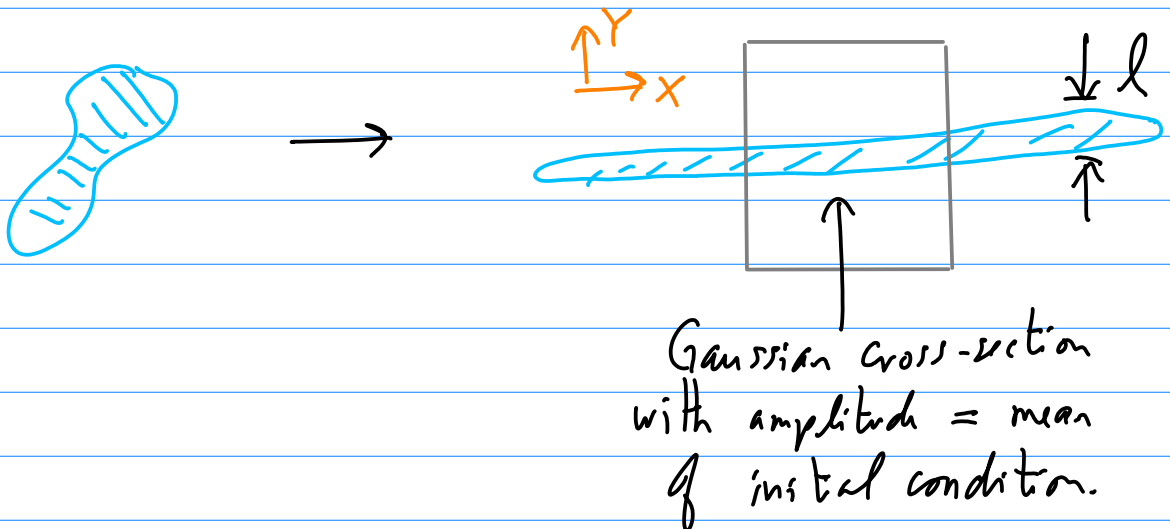
which is equal to  $e^{-y^2/2l^2}$ . Hence,

$$\theta(x, y, z, t) = e^{-\lambda_s |t|} \frac{e^{-y^2/2l^2}}{\sqrt{2\pi} l^2} \int_{-\infty}^{\infty} \theta_0(0, Y_0, z e^{-\lambda_m t}) dY_0$$

In 2D, we can drop the 3rd argument of  $\theta_0$ :

$$\theta(x, y, z, t) = e^{-\lambda t} \frac{e^{-y^2/2l^2}}{\sqrt{2\pi} l^2} \int_{-\infty}^{\infty} \theta_0(0, Y_0) dY_0$$

Recall our filament solution  $e^{-\lambda t} e^{-y^2/2l^2}$ :  
 we have just shown that (assuming our neglect of terms was justified) any compactly-supported initial  $\theta_0$  will converge to the filament solution locally.



Note that  $\partial_x^2 \theta \sim e^{-3\lambda t}$ ,  $\partial_y^2 \theta \sim e^{-\lambda t}$ ,

so  $\frac{\partial_x^2 \theta}{\partial_y^2 \theta} \sim e^{-2\lambda t} \rightarrow 0$  is always negligible

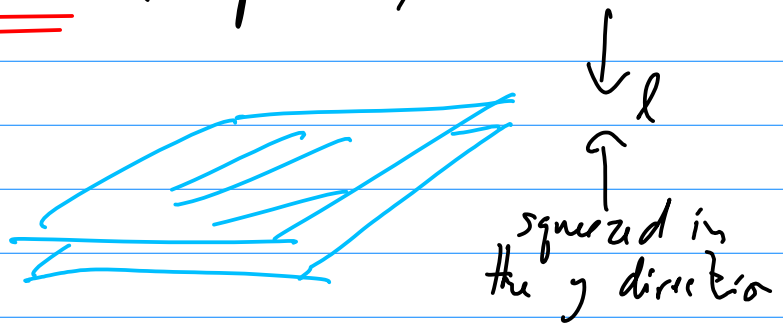
What about 3D? We already assumed  $\lambda_m + |\lambda_s| > 0$ .  
But  $\lambda_m$  can have either sign (or 0).

If  $\lambda_m > 0$ , then the  $z$  coordinate is stretched  
just like the  $x$  coordinate, and so  $ze^{-\lambda_m t} \rightarrow 0$ :

$$\theta(x, y, z, t) = e^{-|\lambda_s|t} \frac{e^{-y^2/2\ell^2}}{\sqrt{2\pi} \ell^2} \int_{-\infty}^{\infty} \theta_0(0, y_0, 0) dy_0$$

3D,  $\lambda_m > 0$

The filament is a sheet (or pancake):



If  $\underline{-|\lambda_s| < \lambda_m < 0}$ , then the  $z$  direction  
behaves like the  $y$  direction.

$$\partial_x^2 \theta \sim e^{-(|\lambda_s| + 2\lambda_u)t}$$

$$\partial_y^2 \theta \sim e^{-|\lambda_s|t}$$

$$\partial_z^2 \theta \sim e^{-(|\lambda_s| - 2|\lambda_m|)t}$$

Notice that  $\frac{\partial_z^2 \theta}{\partial_y^2 \theta} \sim e^{2|\lambda_m|t} \rightarrow \infty$  !

So we are not justified in neglecting the  $\partial_z^2 \theta$  term as  $t \rightarrow \infty$ . Diffusion destroys gradients in the  $y$  (fastest contracting) direction, but that leaves gradients in the other contracting direction. (Z)

We can then show:

$$\theta(x, y, z, t) = e^{-(|\lambda_s| + |\lambda_m|)t} \frac{e^{-\frac{1}{2}\left(\frac{y^2}{l_s^2} + \frac{z^2}{l_m^2}\right)}}{2\pi l_s l_m}$$

$$\times \int_{-\infty}^{\infty} \theta(0, Y_0, Z_0) dY_0 dZ_0$$

This is a filament with an elliptical cross-section

$$\left( l_s = \sqrt{\frac{\kappa}{|\lambda_s|}}, l_m = \sqrt{\frac{\kappa}{|\lambda_m|}} \right)$$

## Lecture 6: Shear flows

$$\partial_t \theta + x \cdot A^T \cdot \partial_x \theta = \kappa \Delta \theta, \quad \text{tr } A(t) = 0.$$

$$B(t) = \int_0^t A(\tau) d\tau$$

Consider the two-dimensional case.

Eigenvalues of  $A = 1$  and  $-1$ .

What about  $\lambda = 0$ ? Then either  $A \equiv 0$   
or  $A$  is non-normal:

$$A^2 = 0, \quad A \neq 0$$

Any matrix  $A$  with  $\text{tr } A = \det A = 0$  will satisfy  $A^2 = 0$ .

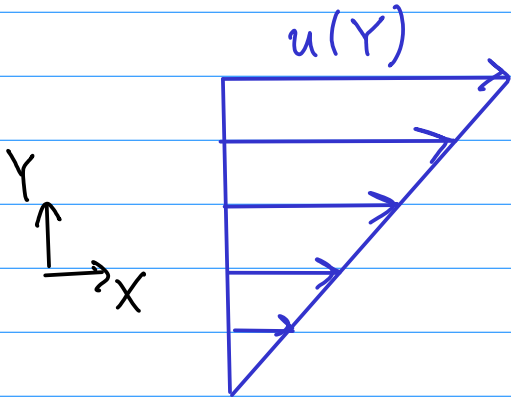
Take, for instance,  $A = \begin{pmatrix} 0 & \alpha \\ 0 & 0 \end{pmatrix}$ .

$$\begin{aligned} e^{B} &= e^{At} = I + At + \underbrace{\frac{1}{2}(At)^2}_{0} + \dots \\ &= I + At \end{aligned}$$



So particle trajectories are

$$\underline{x} = e^B \underline{X} = \begin{pmatrix} 1 & \alpha t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} X + \alpha t Y \\ Y \end{pmatrix}$$



This is called a shear flow.

These are flows that only vary  $\perp$  to their direction,

What is the metric?

$$g = (e^B)^T (e^B) = \begin{pmatrix} 1 & 0 \\ \alpha t & 1 \end{pmatrix} \begin{pmatrix} 1 & \alpha t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \alpha t \\ \alpha t & 1 + (\alpha t)^2 \end{pmatrix}$$

$$g^{-1} = (e^{-B})(e^{-B})^T = \begin{pmatrix} 1 & -\alpha t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -\alpha t & 1 \end{pmatrix} = \begin{pmatrix} 1 + (\alpha t)^2 & -\alpha t \\ -\alpha t & 1 \end{pmatrix}$$

Hence,

$$\nabla_{\underline{x}} \cdot (g^{-1} \cdot \nabla_{\underline{x}} \textcircled{H}) = (1 + (\alpha t)^2) \frac{\partial^2 \textcircled{H}}{\partial X^2} - 2\alpha t \frac{\partial^2 \textcircled{H}}{\partial X \partial Y} + \frac{\partial^2 \textcircled{H}}{\partial Y^2}$$

$$\text{let } \hat{\Psi}(k_x, k_y, t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Psi(X, Y, t) e^{-i(k_x X + k_y Y)} dX dY$$

(Fourier transform)

Then:

$$\partial_t \hat{\Psi} = -n \left( (1 + (\alpha t)^2) k_x^2 - 2 \alpha t k_x k_y + k_y^2 \right) \hat{\Psi}$$

with solution

$$\hat{\Psi}(k_x, k_y, t) = \hat{\Psi}_0(k_x, k_y) \times \exp \left\{ -n \left( \left( t + \frac{1}{3} \alpha^2 t^3 \right) k_x^2 - \alpha t^2 k_x k_y + t k_y^2 \right) \right\}$$

$$\Psi(X, Y, t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{\Psi}_0(k_x, k_y) \frac{e^{i(k_x X + k_y Y)}}{(2\pi)^2} \exp \left\{ -n \left( \left( t + \frac{1}{3} \alpha^2 t^3 \right) k_x^2 - \alpha t^2 k_x k_y + t k_y^2 \right) \right\} dk_x dk_y$$

The biggest term in the exponential is  $t^3$ .

$$e^{-\frac{1}{3} n \alpha^2 t^3 k_x^2}$$

For large  $t$  this kills the integral, unless

$$k_x \sim t^{-3/2}$$

So small wavenumbers get selected in  $X$ .

In  $Y$ , the dominant term is

$$e^{-\kappa \alpha t^2 K_X K_Y} \sim e^{-\kappa \alpha t^2 t^{-3/2} K_Y}$$

$t^{1/2}$

So  $K_Y \sim t^{-1/2}$ .

Given these scalings, can neglect  $t K_Y^2 \sim t^{-2}$ ,  
but NOT  $t K_X^2 \sim 1$ .

Now rescale: let  $K_X = \xi(\alpha t)^{-3/2}$ ,  $K_Y = \eta(\alpha t)^{-1/2}$ :

$$\Theta(X, Y, t) = \int_{-\infty}^{\infty} \hat{\Theta}_0(\xi(\alpha t)^{-3/2}, \eta(\alpha t)^{-1/2}) \frac{e^{i(\xi(\alpha t)^{-3/2} X + \eta(\alpha t)^{-1/2} Y)}}{(2\pi)^2} \\ \times \exp\left\{-\frac{\kappa}{\alpha} \left(\frac{1}{3} \xi^2 - \xi \eta + \eta^2\right)\right\} (\alpha t)^{-2} d\xi d\eta$$

If we assume  $\hat{\Theta}_0(\underline{k}) e^{-\frac{\kappa}{\alpha}(\cdot)}$  decays  
for large  $|\underline{k}|$  (true even if "rough"), can approximate

$$\hat{\Theta}_0(\xi(\alpha t)^{-3/2}, \eta(\alpha t)^{-1/2}) \rightarrow \hat{\Theta}_0(0, 0)$$

*this is the average  
of the initial condition*

Then we can explicitly do the integrals:

$$\Theta(X, Y, t) = \frac{2\pi\sqrt{3}}{(2\pi)^2} \frac{(\alpha t)^{-2}}{\chi^2} \hat{\Theta}_0(0, 0)$$

$$\times \exp \left\{ -\frac{1}{\chi^2 t^3} \left( 3X^2 + 3\alpha t XY + (\alpha t)^2 Y^2 \right) \right\}$$

where  $\chi = \sqrt{\frac{\kappa}{\alpha}}$  (length scale)

Now note that  $X = x - \alpha t$ ,  $Y = y$ , so

$$3X^2 + 3\alpha t XY + (\alpha t)^2 Y^2 = 3x^2 - 3(\alpha t)xy + (\alpha t)^2 y^2$$

Hence we can write the exponential as

$$e^{-\frac{1}{2} \underline{x} \cdot Q \cdot \underline{x}}$$

$$\text{where } Q = \frac{1}{\chi^2 t^3} \begin{pmatrix} 3 & -\frac{3}{2}(\alpha t) \\ -\frac{3}{2}(\alpha t) & (\alpha t)^2 \end{pmatrix}$$

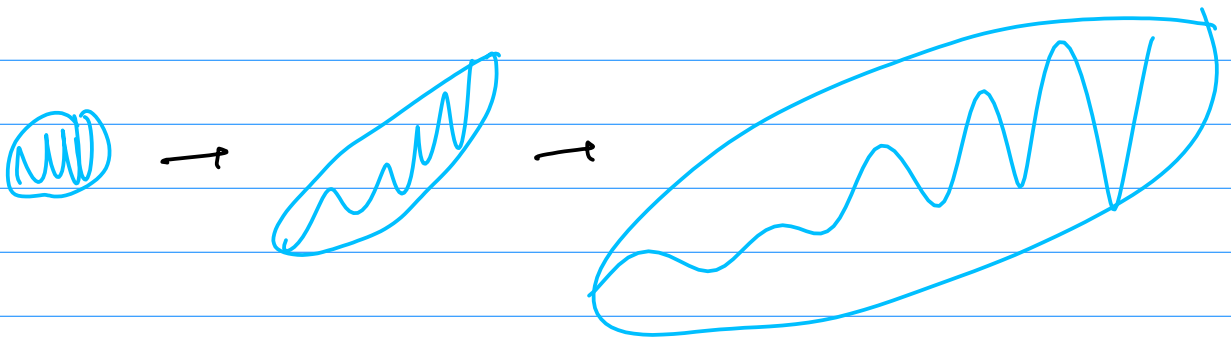
$Q(t)$  has eigenvalues that go as

$$\frac{3}{4\chi^2(\alpha t)^3} \quad \text{and} \quad \frac{1}{\chi^2(\alpha t)} \quad \text{as } \alpha t \rightarrow \infty$$

These correspond to the axes of an ellipsoid:

$$a \sim \frac{2}{\sqrt{3}} \chi(\alpha t)^{3/2}, \quad b \sim \chi(\alpha t)^{1/2}$$

So what happens? A blob tilts in the shear

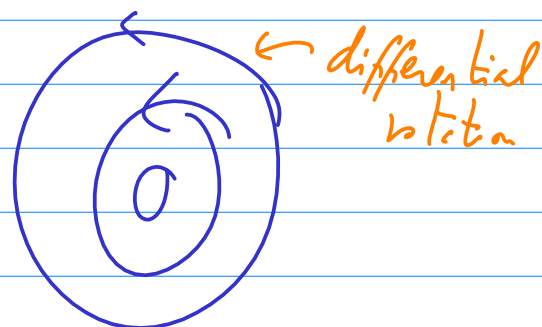


But unlike the exponential case neither direction achieves a constant width. Both axes keep growing, though one at  $t^{3/2}$ , the other  $t^{1/2}$ .

This means the area  $\sim t^{3/2} t^{1/2} = t^2$ ,

so by conservation of  $\theta$  we expect  $\theta \sim t^{-2}$ , as is indeed the case.

Shear flows are quite common near boundaries and in vortices.



Side note: What is the SVD of  $B$ ?

$$e^B = U D V^T \quad D = \begin{pmatrix} \Lambda & 0 \\ 0 & \Lambda^{-1} \end{pmatrix} \quad \leftarrow \text{eigenvalues of } g$$

$$\Lambda = \frac{1}{2} (|\alpha t| + \sqrt{4 + (\alpha t)^2}) > 1$$

$U, V$  are now time-dependent...

Write as before  $\hat{u}, \hat{s}$ :  $g\hat{u} = \Lambda^2 \hat{u}$ ,  $g\hat{s} = \Lambda^{-2} \hat{s}$

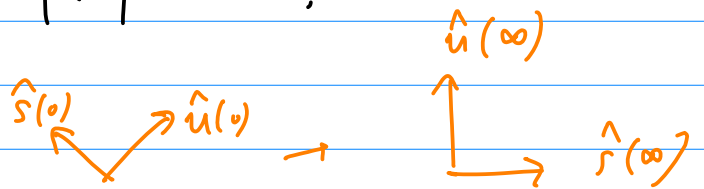
$$\hat{u} = \frac{1}{\sqrt{1 + \Lambda^{-2}}} \begin{pmatrix} \Lambda^{-1} \\ 1 \end{pmatrix}, \quad \hat{s} = \frac{1}{\sqrt{1 + \Lambda^{-2}}} \begin{pmatrix} 1 \\ -\Lambda^{-1} \end{pmatrix}$$

For large  $t$ ,

$$\Lambda \sim (\alpha t)^2, \quad \left. \begin{array}{l} \hat{u} \sim \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \hat{s} \sim \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{array} \right\} |\alpha t| \rightarrow \infty.$$

$$\hat{u} \sim \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

$$\hat{s} \sim \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$



The eigenvectors turn by  $\pi/4$  as  $t$  goes from 0 to  $\infty$

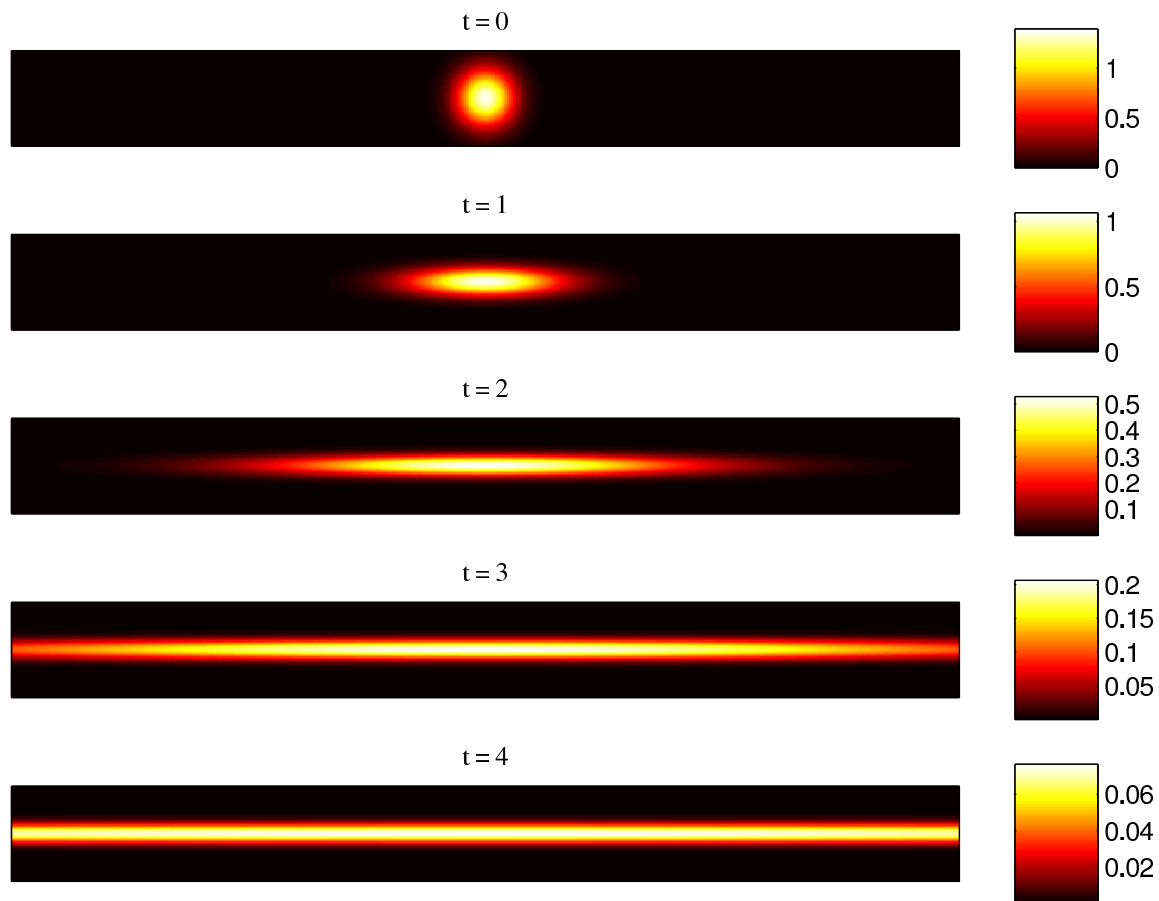


Figure 1: A patch of dye in a uniform straining flow. The amplitude of the concentration field decreases exponentially with time. The length of the filament increases exponentially, whilst its width is stabilised at  $\ell = \sqrt{\kappa/\lambda}$ . (From J.-L. THIFFEAULT, *Scalar decay in chaotic mixing*, in *Transport and Mixing in Geophysical Flows*, J. B. Weiss and A. Provenzale, eds., vol. 744 of *Lecture Notes in Physics*, Berlin, 2008, Springer, pp. 3–35.)

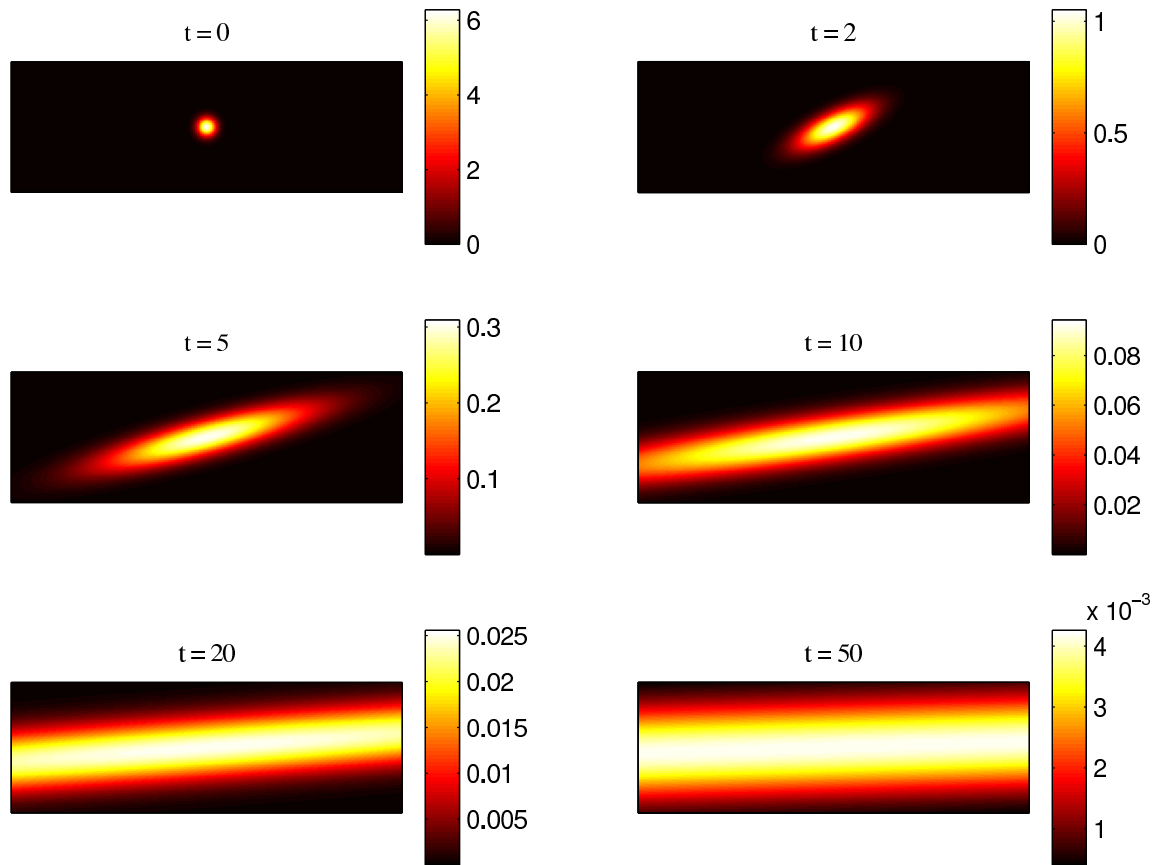


Figure 2: A patch of dye in a uniform shearing flow. The amplitude of the concentration field decreases algebraically with time as  $t^{-2}$ . The length of the filament increases as  $t^{3/2}$ , whilst its width increases as  $t^{1/2}$ . (From J.-L. THIFFEAULT, *Scalar decay in chaotic mixing*, in *Transport and Mixing in Geophysical Flows*, J. B. Weiss and A. Provenzale, eds., vol. 744 of *Lecture Notes in Physics*, Berlin, 2008, Springer, pp. 3–35.)



## Lecture 7: The tangent map

For typical cases thus far,  $\theta \sim e^{-\lambda t}$ .

$\lambda$  comes from the eigenvalues of  $A$  in  $\dot{u} = A \cdot x$ .

$$A = (\nabla u)^T$$

How do we generalize  $\lambda$  for flows where  $\nabla u$  is not a spatially-constant matrix?

Go back to trajectories:

$$\dot{x} = u(x(t), t), \quad x(0) = X$$

Write solutions as  $x = \varphi_t(X)$ .

The tangent map  $D\varphi_t(X) := \frac{\partial \varphi_t}{\partial X}(X)$

(This is the same as  $\frac{\partial x^i}{\partial X^j}$  earlier.)

Recall  $\dot{x}$  means  $\frac{\partial}{\partial t} \Big|_x x$ , so  $\frac{\partial}{\partial t} \Big|_x$  commutes w.  $\frac{\partial}{\partial X}$ .

holding  $x$  constant  $\nearrow$

$$\text{Hence, } \frac{\partial \dot{x}^i}{\partial X^a} = \frac{\partial D\varphi_t}{\partial t} = \frac{\partial}{\partial X^a} u(x(t), t) \\ = \frac{\partial u}{\partial x^i} \frac{\partial x^i}{\partial X^a}$$

Hence,

$$\frac{\partial D\varphi_t}{\partial t} = (\nabla u)^T \cdot D\varphi_t$$

$(D\varphi_t)^{-1}$

This ODE has initial condition  $D\varphi_0 = \mathbb{I}$  (identity)

(This equation must be solved together with  $\dot{x} = u$ .)

For a fixed vector  $\nu$ ,

$$\frac{\partial}{\partial t} (D\varphi_t \cdot \nu) =$$

$$(D\varphi_t \cdot \nu) \cdot \frac{\partial}{\partial t} (D\varphi_t \cdot \nu) = (D\varphi_t \cdot \nu) \cdot (\nabla u)^T \cdot (D\varphi_t \cdot \nu)$$

$$\frac{1}{2} \frac{\partial}{\partial t} \|D\varphi_t \cdot \nu\|^2 = (D\varphi_t \cdot \nu) \cdot e \cdot (D\varphi_t \cdot \nu)$$

$$\text{where } e = \frac{1}{2} (\nabla u + (\nabla u)^T)$$

Rate-of-strain tensor

Note that

$$\underbrace{(D\varphi_t \cdot v)}_v \cdot e \cdot (D\varphi_t \cdot v) = v \cdot (e \cdot v) \leq \|v\| \|e \cdot v\|$$

Cauchy-Schwartz

In  $L^2$ , we have  $\|e \cdot v\| \leq \|e\| \|v\|$

where  $\|e\|$  is the Frobenius norm of  $e$ :

$$\|e\|_F^2 = \sum_{i,j} |e_{ij}|^2.$$

or any matrix norm compatible with vector  $L^2$ , such as the induced norm

$$\|e\| = \max_{w \neq 0} \frac{\|e \cdot w\|}{\|w\|} = \text{spectral radius of } e.$$

Hence,

$$\frac{1}{2} \partial_t \|D\varphi_t v\|^2 \leq \|e\| \|D\varphi_t v\|^2$$

$$\partial_t \|D\varphi_t v\| \leq \|e\| \|D\varphi_t v\|$$

Grönwall's inequality

$$\Rightarrow \|D\varphi_t v\| \leq e^{\int_0^t \|e\| d\tau} \|v\|$$

$$\text{since } \|D\varphi_0 v\| = \|v\|$$

$$\text{or } \frac{1}{t} \log \left( \frac{\|D\varphi_t v\|}{\|v\|} \right) \leq \frac{1}{t} \int_0^t \|e\| d\tau \leq \sup_t \|e\|$$

This at least tells us that for smooth-enough flows, things can't go too crazy.

(Rough  $\nabla u$  can lead to finite-time blowup of  $D\varphi_t$ !)

over time.  
Replace by space/time.

Claim:  $\partial_t \det D\varphi_t = (\nabla \cdot u) \det D\varphi_t.$

Write  $M = D\varphi_t$  to lighten notation,  $M$  is  $d \times d$   
( $\dot{M} = AM$ )

$$\det M = \sum_{i_1 \dots i_d} \sum_{j_1 \dots j_d} \frac{1}{d!} \varepsilon_{i_1 \dots i_d} \varepsilon_{j_1 \dots j_d} M_{i_1 j_1} \dots M_{i_d j_d}$$

$$\partial_t \det M = \sum_{i_1 \dots i_d} \sum_{j_1 \dots j_d} \frac{1}{(d-1)!} \varepsilon_{i_1 \dots i_d} \varepsilon_{j_1 \dots j_d} \dot{M}_{i_1 j_1} M_{i_2 j_2} \dots M_{i_d j_d}$$

← time deriv

AM, with  $A = (\nabla u)^T$

$$\begin{aligned}
\partial_t \det M &= \sum_{i_1 \dots i_d} \sum_{j_1 \dots j_d} \sum_l \frac{1}{(d-1)!} \varepsilon_{i_1 \dots i_d} \varepsilon_{j_1 \dots j_d} \\
&\quad A_{i_1 l} M_{l j_1} M_{i_2 j_2} \dots M_{i_d j_d} \\
&= \sum_{i_1, j_1, l} A_{i_1 l} M_{l j_1} \left( \sum_{i_2 \dots i_d} \sum_{j_2 \dots j_d} \frac{1}{(d-1)!} \varepsilon_{i_1 \dots i_d} \varepsilon_{j_1 \dots j_d} \right. \\
&\quad \left. M_{i_2 j_2} \dots M_{i_d j_d} \right) \\
&\quad \underbrace{\hspace{10em}}_{\text{cofactor matrix}} \\
&\quad = \det M (M^{-1})_{j_1 i_1} \\
&= \sum_{i_1 l} A_{i_1 l} \delta_{l i_1} \det M = (\text{trace } A) \det M.
\end{aligned}$$

Thus, we've finally proved that if  $\text{trace } A = 0$  ( $\nabla \cdot u = 0$ ) then  $\det D\varphi_t = \text{constant} = 1$ .

This also makes it clear why  $\nabla \cdot u$  controls the "compressibility" of the flow.

## Lecture 8: Lyapunov exponents

$$\dot{x} = u(x(t), t), \quad x(0) = X$$

$$\Rightarrow x(t) = \varphi_t(X).$$

For  $\nabla \cdot u = 0$ ,  $\varphi$  preserves volume, since

$$d^3 x = \underbrace{\left| \frac{\partial x}{\partial X} \right|}_{1 \text{ if } \nabla \cdot u = 0} d^3 X$$

Hence  $\varphi$  preserves Lebesgue measure.

Theorem (Oseledec Multiplicative Ergodic Theorem)

Let  $D\varphi_t$  be the tangent map of a measure-preserving flow  $\varphi_t$  on a compact manifold  $M$ .

Then there is a  $\Gamma \subseteq M$ ,  $\mu(M) = 1$ , such that  $\varphi_t \Gamma \subseteq \Gamma$ ,  $t \geq 0$ , and for all  $X \in \Gamma$ :

(1)  $\Lambda_X := \lim_{t \rightarrow \infty} (D\varphi_t^\top(X) D\varphi_t(X))^{1/2t}$  exists

(2) Let  $\exp \lambda_X^{(1)} < \exp \lambda_X^{(2)} < \dots < \exp \lambda_X^{(s)}$

$$s = s(X)$$

be the distinct eigenvalues of  $A_X$ ,  $\lambda_X^{(r)} \in \mathbb{R}$ .  
 (we allow  $\lambda_X^{(1)} = -\infty$ )  
 (since  $A_X^T = A_X$ )

Let  $U_X^{(1)}, \dots, U_X^{(s)}$  the corresponding eigenspaces.

$$m_X^{(r)} = \dim U_X^{(r)}$$

Let  $V_X^{(0)} = \{0\}$  and  $V_X^{(r)} = U_X^{(1)} \oplus \dots \oplus U_X^{(r)}$ ,  $r=1, \dots, s$

Then for  $v \in V_X^{(r)} \setminus V_X^{(r-1)}$ ,  $1 \leq r \leq s$ :

$$\lim_{t \rightarrow \infty} \frac{1}{t} \log \|D\varphi_t(X)v\| = \lambda_X^{(r)}$$

The  $\lambda_X^{(r)}$  are called Lyapunov exponents

or characteristic exponents.

They are  $\varphi_t$ -invariant:  $\lambda_{\varphi_t(X)}^{(r)} = \lambda_X^{(r)}$ .

The subspaces  $\{V_X^{(r)}\}_{r=0}^s$  are nested:

$$\{0\} = V_X^{(0)} \subset \dots \subseteq V_X^{(s)} = \mathbb{R}^d$$

← dimension  
filtration

and are also  $\varphi_t$ -invariant:  $D\varphi_t(X)V_X^{(r)} = V_{\varphi_t(X)}^{(r)}$

Definition:  $\varphi$  is ergodic if the only measurable sets that are mapped to themselves ( $\varphi_t(E) = E, t > 0$ ) have  $\mu(E) = 0$  or  $\mu(E) = 1$ .

If  $\varphi$  is ergodic, then  $\lambda_x^{(r)}$  and  $s(x)$  are constant almost everywhere. (Then drop the  $x$ )

So how do we interpret the theorem?

First note that it is a statement of mean exponential growth:

$$\|D\varphi_t v\| \sim e^{\lambda^{(s)} t}, \text{ for almost all } v. \quad t \rightarrow \infty$$

Why almost all  $v$ ? Because  $V_x^{(s)} = \mathbb{R}^d$ .

So essentially all vectors in the tangent space grow exponentially at a rate  $\lambda^{(s)}$ .

So which vectors grow at  $\lambda^{(s-1)}$ ? The ones that have no projection onto this fastest-growing eigenspace:

$$V_x^{(s-1)} = \mathbb{R}^d \setminus V_x^{(s)}.$$

This is a well-known phenomenon which can be illustrated with a constant matrix:

in space



$$D\varphi_t = \begin{pmatrix} e^{\lambda^{(1)}t} & \\ & e^{\lambda^{(2)}t} \end{pmatrix}, \quad \lambda^{(1)} < \lambda^{(2)}$$

$$\text{Then } D\varphi_t \begin{pmatrix} u \\ v \end{pmatrix} = u e^{\lambda^{(1)}t} + v e^{\lambda^{(2)}t} \\ \rightarrow u e^{\lambda^{(1)}t} \text{ as } t \rightarrow \infty,$$

unless  $u = 0$ . Then  $\rightarrow v e^{\lambda^{(2)}t}$  as  $t \rightarrow \infty$ .

The power of Oseledec's thm is that it extends this simple idea to much greater generality.

There is a discrete version of the theorem, which applies to measure-preserving maps.

$$x_n = \varphi_n(x), \quad D\varphi_n(x) \text{ tangent map.}$$

$$\text{Then } \lambda_x^{(r)} = \lim_{n \rightarrow \infty} \frac{1}{n} \log \| D\varphi_n(x)v \| \\ v \in V_x^{(r)} \setminus V_x^{(r-1)}$$

When the map is autonomous, write:

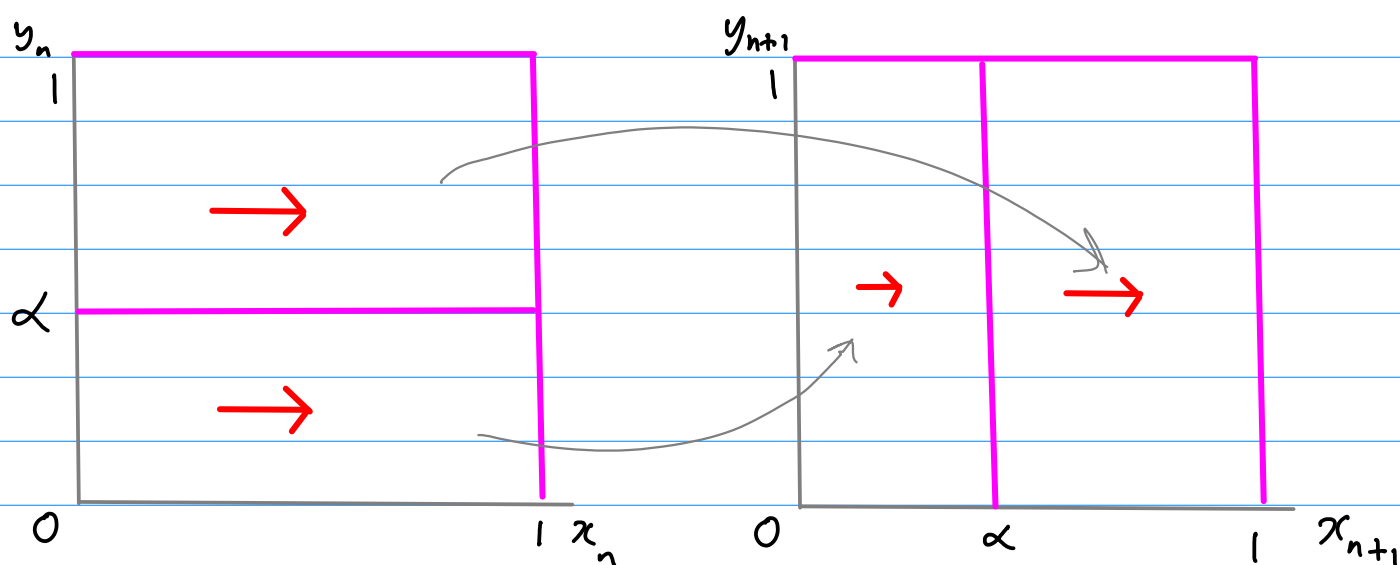
$$\varphi_n(x) = \varphi(\varphi(\varphi \dots \varphi(x))) = \varphi^{(n)}(x)$$

Example: Baker's map

$$X = (x, y) \in [0, 1]^2. \quad X_{n+1} = \varphi(X_n)$$

$$x_{n+1} = \begin{cases} \alpha x_n & y_n < \alpha \\ \alpha + \beta x_n & y_n > \alpha \end{cases} \quad \alpha + \beta = 1$$

$$y_{n+1} = \begin{cases} \alpha^{-1} y_n & y_n < \alpha \\ \beta^{-1} (y_n - \alpha) & y_n > \alpha \end{cases}$$



$$D\varphi = \begin{cases} \begin{pmatrix} \alpha & \\ & \alpha^{-1} \end{pmatrix}, & y < \alpha \\ \begin{pmatrix} \beta & \\ & \beta^{-1} \end{pmatrix}, & y > \alpha \end{cases} \quad \text{area-preserving}$$

Can be shown that this is ergodic

Note that a vector  $v = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$  always shrinks.

All other vectors expand:  $V_x^{(2)} = \mathbb{R}^2$ ,  $V_x^{(1)} = t \begin{pmatrix} 1 \\ 0 \end{pmatrix}$   
 $t \in \mathbb{R}$

Now compute Lyapunov exponent:

$$\begin{aligned} \lambda^{(1)} &= \lim_{n \rightarrow \infty} \frac{1}{n} \log \|D\varphi_n(x)v\|, \quad v = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= \lim_{n \rightarrow \infty} \frac{1}{n} \log \|D\varphi(\varphi_{n-1}x) \cdots D\varphi(\varphi_1x) D\varphi(x)v\| \end{aligned}$$

(example:  $D\varphi_2(x) = D(\varphi(\varphi(x))) = D\varphi(\varphi(x)) D\varphi(x)$ )

Each of the  $D\varphi(\varphi_k x)$  is either  $\begin{pmatrix} \alpha^{-1} \\ \alpha \end{pmatrix}$  or  $\begin{pmatrix} \beta^{-1} \\ \beta \end{pmatrix}$ .

So let  $\Delta_n = r(\varphi_{n-1}x) \cdots r(\varphi_1x) r(x)$

where  $r(x) = \begin{cases} \alpha & , \quad y < \alpha \\ \beta & , \quad y > \alpha \end{cases}$

$$\begin{aligned} \lambda^{(1)} &= \lim_{n \rightarrow \infty} \frac{1}{n} \log \Delta_n \\ &= \lim_{n \rightarrow \infty} \frac{1}{n} \log (r(\varphi_{n-1}x) \cdots r(\varphi_1x) r(x)) \end{aligned}$$

$$= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=0}^{n-1} \log r(\varphi_k X)$$

Now we invoke Birkhoff's ergodic theorem:

$$\lambda^{(1)} = E \log r(X) = \int \log r(x) d\mu(x)$$

Since  $\mu$  is uniform,  $r(x) = \alpha$  on a fraction  $\alpha$  of the space,  $\beta$  on a fraction  $\beta$  of the space.

$$\lambda^{(1)} = \alpha \log \alpha + \beta \log \beta < 0$$

Since  $\det D\varphi = 1$ ,  $\lambda^{(2)} = -\lambda^{(1)} > 0$ .

This is a rare case with an analytic expression for  $\lambda$ .

## Lecture 9: Finite-time Lyapunov exponents

Last time:

$$\lambda_X^{(r)} = \lim_{t \rightarrow \infty} \frac{1}{t} \log \|D\varphi_t(X)\nu\|$$

For ergodic systems,  $\lambda_X^{(r)} = \lambda^{(r)}$ .

Now consider dropping the limit:

$$\lambda_X^{(r)}(t) = \frac{1}{t} \log \|D\varphi_t(X)\nu\|$$

Now even for ergodic systems these are a function of  $X$ .

Consider the SVD of  $D\varphi_t(X)$ :

$$D\varphi_t(X) = U_t(X) D_t(X) V_t^T(X).$$

Drop the  $t$ 's and  $X$ 's

$$\begin{aligned} \|D\varphi_t(X)\nu\|^2 &= \nu \cdot D\varphi_t^T D\varphi_t \cdot \nu \\ &= \nu \cdot V D^2 V^T \cdot \nu \end{aligned}$$

Now choose  $\nu_i^{(h)} = V_{ih}$ , the  $h^{\text{th}}$  column of  $V$ .

$$\begin{aligned}
\| D\varphi_t(X) v^{(h)} \|^2 &= v_i^{(h)} V_{im} D_{ml}^2 V_{jl} v_j^{(h)} \\
&= V_{ih} V_{im} D_{ml}^2 V_{jl} V_{jk} \\
&= D_{hh}^2 \quad (\text{no sum over } h)
\end{aligned}$$

So even for finite time we can define the eigenspaces. We have also

$$D_{hh} = e^{\lambda_X^{(h)}(t) t}$$

It is convenient to derive equations for the time-evolution of the SVD.

Recall:  $\frac{d}{dt} D\varphi_t = (\nabla u)^T D\varphi_t$

Let  $M = D\varphi_t$ ,  $A = (\nabla u)^T$ .

$$\dot{M} = AM, \quad M = UDV^T$$

$$\dot{M} = \dot{U}DV^T + U\dot{D}V^T + UD\dot{V}^T = AM$$

$$(*) \quad U^T \dot{U} D + \dot{D} + D \dot{V}^T V = \underbrace{U^T A U}_{\hat{A}}$$

Note that  $U^T \dot{U}$  is antisymmetric, since

$$\frac{d}{dt} (U^T U) = 0 = \dot{U}^T U + U^T \dot{U}$$

$$(U^T \dot{U})^T = -U^T \dot{U}$$

and similarly for  $V^T \dot{V}$ .

$$(U^T \dot{U} D)_{ij} = (U^T \dot{U})_{ih} D_{hj} = (U^T \dot{U})_{ij} D_{jj}$$

$$(D \dot{V}^T V)_{ij} = D_{ih} (\dot{V}^T V)_{hj} = D_{ii} (\dot{V}^T V)_{ij}$$

No sum  
over  $j$

Hence, if we evaluate the matrix equation (\*) on its diagonal ( $i=j$ ), we find:

$$\underbrace{(U^T \dot{U})_{ii}}_0 D_{ii} + \dot{D}_{ii} + D_{ii} \underbrace{(\dot{V}^T V)_{ii}}_0 = \hat{A}_{ii} D_{ii}$$

(no sum over  $i$ )

$$\dot{D}_{ii} = \hat{A}_{ii} D_{ii}$$

(no sum over  $i$ )

Equation for  
eigenvalues of  $M$   
(depends on  $U$ )

Next we need equations for  $U$  and  $V$ .

Take (\*) again but evaluate for  $i \neq j$ :

$$(1) \quad (U^T \dot{U})_{ij} D_{jj} + \underbrace{\dot{D}_{ij}}_0 + D_{ii} (\dot{V}^T V)_{ij} = \hat{A}_{ij} D_{jj}$$

$(i \neq j, \text{ no sum over } i \text{ and } j)$

Interchange  $i, j$ :

$$(U^T \dot{U})_{ji} D_{ii} + D_{jj} (\dot{V}^T V)_{ji} = \hat{A}_{ji} D_{ii}$$

$$(2) \quad -(U^T \dot{U})_{ij} D_{ii} - D_{jj} (\dot{V}^T V)_{ij} = \hat{A}_{ji} D_{ii}$$

Now divide (1) by  $D_{jj}$  and (2) by  $D_{ii}$  and add:

$$\left( \frac{D_{ii}}{D_{jj}} - \frac{D_{jj}}{D_{ii}} \right) (\dot{V}^T V)_{ij} = \hat{A}_{ij} + \hat{A}_{ji}$$

$$\text{let } \Delta_{ij} = D_{ii} / D_{jj}.$$

$$\boxed{(\dot{V}^T V)_{ij} = \frac{\hat{A}_{ij} + \hat{A}_{ji}}{\Delta_{ij} - \Delta_{ji}}}$$

$(i \neq j)$

The RHS is antisym in  $i \leftrightarrow j$ , as required



Note that  $(\dot{V}^T V)_{ij} = 0$  for  $i=j$ .

Can turn this into an equation for  $\dot{V}$ :

$$\dot{V}_{ij} = -V_{il} (\dot{V}^T V)_{lj}$$

$$\text{Now } \Delta_{ij} = D_{ii}/D_{jj}$$

$$\text{with } D_{ii} = e^{\lambda^{(i)} t}$$

Now assume we order the exponents such that

$$\lambda^{(1)} > \lambda^{(2)} > \dots > \lambda^{(s)}$$

(Reverse from theorem, but more convenient at this point.)

$$\text{Then } \Delta_{ij} = e^{(\lambda^{(i)} - \lambda^{(j)}) t}$$

$$\text{For large } t, \Delta_{ij} \rightarrow \begin{cases} \infty, & i < j \\ -\infty, & i > j \end{cases}$$

$$\text{So } \Delta_{ij} + \Delta_{ji} \rightarrow \exp(|\lambda^{(i)} - \lambda^{(j)}| t) + O(-|\lambda^{(i)} - \lambda^{(j)}| t),$$

$t \rightarrow \infty$

$$\left[ \begin{array}{l} \text{Since:} \\ (\dot{V}^T V) V^T = \dot{V}^T \\ V (\dot{V}^T V)^T = \dot{V} \\ -V (\dot{V}^T V) = \dot{V} \end{array} \right]$$

Hence,

$$(\dot{V}V)_{ij} = e^{-|\lambda^{(i)} - \lambda^{(j)}|t} (\hat{A}_{ij} + \hat{A}_{ji}) + O(e^{-2|\lambda^{(i)} - \lambda^{(j)}|t}),$$

$t \rightarrow \infty$

Conclude:

$\dot{V} \rightarrow 0$  exponentially, at a rate given by the 'gap' between exponents.

This will happen even if the  $\lambda^{(i)}$  themselves haven't quite converged yet.

This is typical: the Lyapunov exponents often converge slowly, but the characteristic direction on a given trajectory converges very rapidly.

What about the  $U$ ?

Take  $\frac{(1)}{D_{ii}} + \frac{(2)}{D_{jj}}$ :

$$(U^T \dot{U})_{ij} \Delta_{ji} + (\dot{V}^T V)_{ij} = \hat{A}_{ij} \Delta_{ji}$$

$$-(U^T \dot{U})_{ij} \Delta_{ij} - (\dot{V}^T V)_{ij} = \hat{A}_{ji} \Delta_{ij}$$

$$\Rightarrow (U^T \dot{U})_{ij} (\Delta_{ji} - \Delta_{ij}) = \Delta_{ij} \hat{A}_{ji} + \Delta_{ji} \hat{A}_{ij}$$

$$(U^T \dot{U})_{ij} = \frac{\Delta_{ij} \hat{A}_{ji} + \Delta_{ji} \hat{A}_{ij}}{(\Delta_{ji} - \Delta_{ij})}$$

*( $i \neq j$ , no sum over  $i$  and  $j$ )*

For large  $t$ ,

$$(U^T \dot{U})_{ij} = \begin{cases} -\hat{A}_{ji}, & i < j \\ \hat{A}_{ij}, & i > j \end{cases} \quad t \rightarrow \infty$$

So  $\dot{U} \neq 0 \Rightarrow U$  doesn't converge.

## Lecture 10: FTLEs for the baker's map

Recall the (generalized) baker's map:

$$X = (x, y) \in [0, 1]^2. \quad X_{n+1} = \varphi(X_n)$$

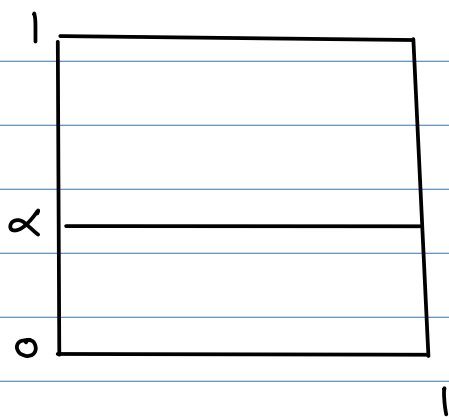
$$x_{n+1} = \begin{cases} \alpha x_n & y_n < \alpha \\ \alpha + \beta x_n & y_n > \alpha \end{cases} \quad \alpha + \beta = 1$$

$$y_{n+1} = \begin{cases} \alpha^{-1} y_n & y_n < \alpha \\ \beta^{-1} (y_n - \alpha) & y_n > \alpha \end{cases}$$

We computed Lyapunov exponents

$$\lambda^{(2)} = -\lambda^{(1)} = \lambda = \alpha \log \alpha^{-1} + \beta \log \beta^{-1}$$

We used Birkhoff's ergodic thm, which suggests iterates land in lower box with probability  $\alpha$ , upper box w. prob  $\beta$ .



Suggests writing

$$\lambda_n = \frac{1}{n} \sum_{k=1}^n h_k$$

where  $h_k = \log \alpha^{-1}$  w. prob  $\alpha$

$\log \beta^{-1}$  w. prob  $\beta$

are i.i.d.

If  $n \rightarrow \infty$  the law of large numbers says  $\lambda_n \rightarrow \lambda$ .

What about for finite  $n$ ? What is the PDF of  $\lambda_n$ ?

(prob. dist. func.)

This is a standard problem in probability.

For small deviations the central limit theorem holds, but we'll be interested in large deviations:

$$\mathbb{P} \left\{ |\lambda_n - \lambda| < \frac{c}{\sqrt{n}} \right\} \quad \text{vs} \quad \mathbb{P} \left\{ |\lambda_n - \lambda| < c \right\}$$

small deviations

large deviations

$$\langle h \rangle = \lambda = \alpha \log \alpha^{-1} + \beta \log \beta^{-1}$$

$$\begin{aligned} \sigma^2 &= \langle (h - \lambda)^2 \rangle = \alpha (\log \alpha^{-1} - \lambda)^2 + \beta (\log \beta^{-1} - \lambda)^2 \\ &= \alpha \left( \underbrace{(1-\alpha)}_{\beta} \log \alpha^{-1} - \beta \log \beta^{-1} \right)^2 + \beta \left( \underbrace{(1-\beta)}_{\alpha} \log \beta^{-1} - \alpha \log \alpha^{-1} \right)^2 \end{aligned}$$

$$= \alpha \beta^2 \left( \log (\alpha^{-1} / \beta^{-1}) \right)^2 + \beta \alpha^2 \left( \log (\beta^{-1} / \alpha^{-1}) \right)^2$$

$$= \alpha \beta (\alpha + \beta) \log^2 (\alpha / \beta)$$

$$\sigma^2 = \alpha \beta \log^2 (\alpha / \beta)$$

To find the distribution of  $\lambda_n$ , first find the generating function for  $h$ :

$$\langle e^{hs} \rangle = \alpha e^{s \log \alpha^{-1}} + \beta e^{s \log \beta^{-1}}$$

The generating function for  $\lambda_n$  is:

$$\begin{aligned} \langle e^{\lambda_n s} \rangle &= \langle e^{\frac{s}{n} \sum h_n} \rangle \\ &= \langle \prod e^{\frac{s}{n} h_n} \rangle \\ &= \langle e^{\frac{s}{n} h} \rangle^n \quad \text{since } h_n \text{ are i.i.d.} \\ &= (\alpha e^{\frac{s}{n} \log \alpha^{-1}} + \beta e^{\frac{s}{n} \log \beta^{-1}})^n \\ &= e^{n \Lambda(s/n)} \\ &= e \end{aligned}$$

where

$$\Lambda(s) = \log (\alpha e^{s \log \alpha^{-1}} + \beta e^{s \log \beta^{-1}})$$

We want to invert the two-sided Laplace transform to obtain the prob. density function of  $\lambda_n$ , for large  $n$ .

Cramér's theorem then tells us we need to find:

$$I(x) = \sup_s \{xs - \Lambda(s)\}$$

Assume  $0 < \alpha < 1/2$ , so  $\alpha < \beta$  and  $\log \alpha^{-1} > \log \beta^{-1}$ .

$$\Lambda(s) = s \log \beta^{-1} + \log (\alpha e^{s(\log \alpha^{-1} - \log \beta^{-1})} + \beta)$$

We can find  $I(x)$  using calculus:

$$\begin{aligned} \frac{d}{ds} (xs - \Lambda(s)) &= x - \Lambda'(s) \\ &= x - \log \beta^{-1} - \frac{\alpha (\log \alpha^{-1} - \log \beta^{-1}) e^{(\log \alpha^{-1} - \log \beta^{-1})s}}{\alpha e^{s(\log \alpha^{-1} - \log \beta^{-1})} + \beta} \\ &= 0 \end{aligned}$$

Solve for  $s = s_x(x)$ !

$$s_x(x) = \frac{\log \left( \frac{\beta(x - \log \beta^{-1})}{\alpha(\log \alpha^{-1} - x)} \right)}{\log \alpha^{-1} - \log \beta^{-1}}, \quad \log \beta^{-1} < x < \log \alpha^{-1}.$$

Hence,

$$I(x) = \log \left( \frac{\log \alpha^{-1} - x}{\beta(\log \alpha^{-1} - \log \beta^{-1})} \right) + \left( \frac{x - \log \beta^{-1}}{\log \alpha^{-1} - \log \beta^{-1}} \right) \log \left( \frac{\beta(x - \log \beta^{-1})}{\alpha(\log \alpha^{-1} - x)} \right)$$

This is the rate function (or Cramér function, or entropy function)

Note that  $I(x)$  has a minimum at  $x = \lambda$ ,  
and that

$$I''(\lambda) = 1/\sigma^2$$

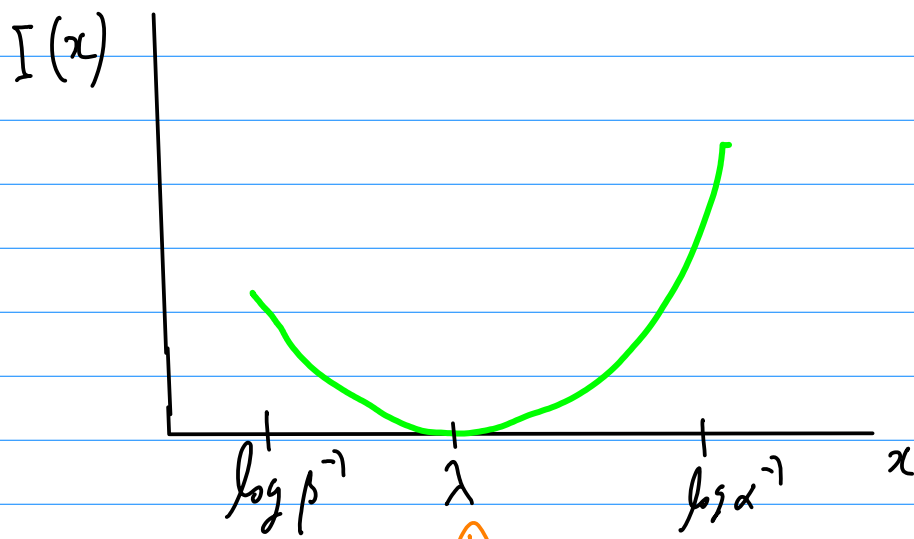
We have finally that

$$p_{\lambda_n}(h) \sim \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-n I(h))$$

$n \rightarrow \infty$

$$\log \beta^{-1} < h < \log \alpha^{-1}$$

Sketch for  $\alpha = 1/4$ ,  $\beta = 3/4$ :



$$I(x) \sim \frac{(x-\lambda)^2}{2\sigma^2}$$

Gaussian approximation



This is normalized properly, since for large  $n$ :

$$\int_{\log \beta^{-1}}^{\log \alpha^{-1}} \rho_{\lambda_n}(h) dh \sim \int_{-\infty}^{\infty} \frac{\sqrt{n}}{\sqrt{2\pi n^2}} \exp\left(-n \frac{I''(\lambda)(h-\lambda)^2}{2}\right) dh$$

$n \rightarrow \infty$

$$= 1 \quad (+ \text{exponentially small terms})$$

This PDF matches simulations very well.

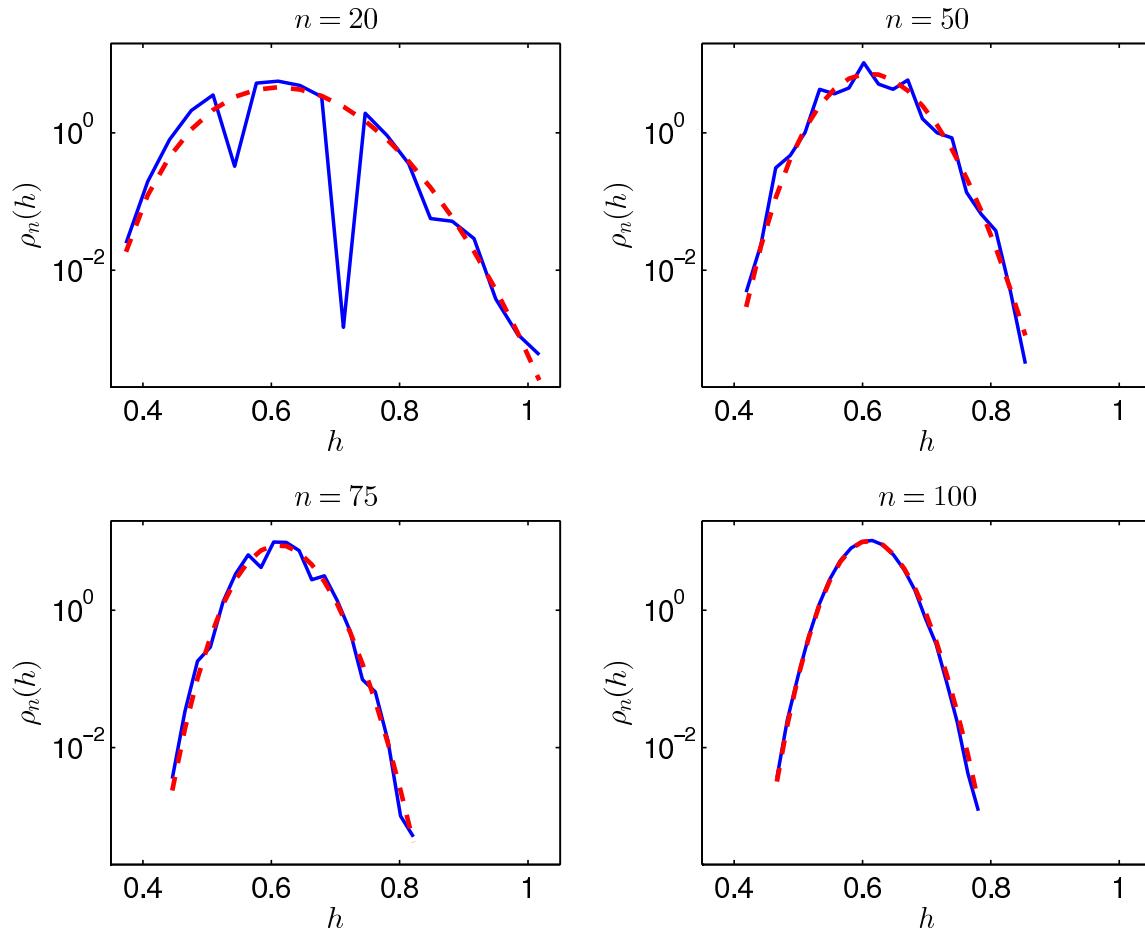


Figure 1: The probability distribution of finite-time Lyapunov exponents for the baker's map with  $\alpha = 0.3$ . As the iterate  $n$  increases, the distribution converges to the large-deviation probability density,  $\rho_n(h)$  (dashed red). The distribution was computed using  $10^5$  randomly-distributed trajectories. See Figs. 2 and 3 for the Matlab code used to generate these figures.

```

function dist_baker

Npts = 100000; Niter = 100; % number of initial points and iterates
alpha = .3; beta = 1-alpha;

% The large-deviation form of the PDF.
mh = @(h) (h + log(beta))/(log(beta/alpha));
G = @(h) mh(h).*log(mh(h)) + (1-mh(h)).*log(1-mh(h)) ...
    + mh(h)*log(beta/alpha) - log(beta);
sigma2 = alpha*beta*log(alpha/beta)^2;
Pld = @(h,n) sqrt(n/2/pi/sigma2) * exp(-n*G(h));

% Specify which 4 values of iterate n to plot.
plotn = [20 50 75 100]; plotgeom = [2 2]; nplot = 0;
% Generate random initial conditions.
rng('default'); X = rand(Npts,2);
lstr = zeros(Npts,1); % the log-stretch
for n = 1:Niter
    [X,str] = baker(X,alpha); % apply baker's map (vectorized)
    lstr = lstr + log(str);
    if any(plotn == n)
        % Plot the results.
        nplot = nplot + 1; subplot(plotgeom(1),plotgeom(2),nplot)
        % Histogram of average stretching, normalized.
        [P,bins] = hist(lstr/n,20); P = P/trapz(bins,P);
        semilogy(bins,P,'b','LineWidth',1.5), hold on
        semilogy(bins,Pld(bins,n),'r--','LineWidth',2), hold off
        xlabel('$h$', 'Interpreter','LaTeX')
        ylabel('$\rho_{n}(h)$', 'Interpreter','LaTeX')
        axis([.35 1.05 2e-4 2e1])
        title(sprintf('$n = %d$',n), 'Interpreter','LaTeX')
    end
end
print -dpdf dist_baker.pdf

```

Figure 2: The Matlab code `dist_baker.m`.

```

function [Xn,stretch] = baker(X,al)
%BAKER Baker's map.
% XN = BAKER(X) returns the image of X=(x,y) under the action of the
% baker's map. BAKER(X,ALPHA) returns the generalized baker's map, where
% 0 < ALPHA < 1. X must be in the unit square [0,1]^2. X can also be an
% array with a 2-vector on each row.
%
% [XN,STRETCH] = BAKER(X) returns a vector STRETCH that records the
% vertical stretching experienced by the particle (1/alpha or
% 1/(1-alpha)). This is used to reconstruct the tangent map.

if nargin < 2
    % Default is the uniform baker's map.
    al = 0.5;
end

if al > 1 || al < 0
    error('Baker''s map requires 0 < alpha < 1.')
end

be = 1-al;
x = X(:,1); y = X(:,2);
xn = zeros(size(x)); yn = zeros(size(y)); % Allocate arrays.

% Formula for y <= alpha.
ila = find(y <= al);
xn(ila) = al*x(ila);
yn(ila) = y(ila)/al;
% Formula for y > alpha.
iga = find(y > al);
xn(iga) = al + be*x(iga);
yn(iga) = (y(iga) - al)/be;

Xn = [xn yn];

if nargin > 1
    stretch = 1/al*ones(size(xn));
    stretch(iga) = 1/be;
end

```

Figure 3: The Matlab code baker.m.

## Lecture 11: Renovating flows

Consider a two-dimensional linear divergence-free velocity field given by

$$\mathbf{u}(\mathbf{x}) = R(\theta) A R^T(\theta) \cdot \mathbf{x} \quad (1)$$

where  $A$  is a constant traceless matrix and

$$R(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (2)$$

is a rotation matrix. The velocity gradient matrix is then

$$(\nabla \mathbf{u})^T = R(\theta) A R^T(\theta). \quad (3)$$

An infinitesimal line segment  $\delta \mathbf{x}$  obeys

$$\delta \dot{\mathbf{x}} = \delta \mathbf{x} \cdot \nabla \mathbf{u}. \quad (4)$$

Hence, as long as  $\nabla \mathbf{u}$  remains constant, the initial line segment  $\delta \mathbf{x}(0)$  is stretched after a time  $\tau$  to

$$\delta \mathbf{x}(\tau) = \exp(\tau R A R^T) \cdot \delta \mathbf{x}(0). \quad (5)$$

For any traceless matrix  $A$  with determinant  $\det A = -\zeta^2$ , we have

$$\exp A = I \cosh \zeta + A \zeta^{-1} \sinh \zeta. \quad (6)$$

Hence,

$$\delta \mathbf{x}(\tau) = (I \cosh \zeta + \tau R A R^T \zeta^{-1} \sinh \zeta) \cdot \delta \mathbf{x}(0). \quad (7)$$

Now let

$$A = \begin{pmatrix} \gamma & -\omega \\ \omega & -\gamma \end{pmatrix}, \quad (8)$$

where  $\gamma$  is the rate-of-strain of the flow, and  $\omega$  is half its vorticity ( $\nabla \times \mathbf{u} = 2\omega \hat{\mathbf{z}}$ ). The corresponding rotated matrix is

$$R A R^T = \begin{pmatrix} \gamma \cos 2\theta & \gamma \sin 2\theta - \omega \\ \gamma \sin 2\theta + \omega & -\gamma \cos 2\theta \end{pmatrix}, \quad (9)$$

and the exponential is

$$\exp(\tau R A R^T) = \begin{pmatrix} \cosh \zeta + (\gamma\tau/\zeta) \cos 2\theta \sinh \zeta & ((\gamma\tau/\zeta) \sin 2\theta - (\omega\tau/\zeta)) \sinh \zeta \\ ((\gamma\tau/\zeta) \sin 2\theta + (\omega\tau/\zeta)) \sinh \zeta & \cosh \zeta - (\gamma\tau/\zeta) \cos 2\theta \sinh \zeta \end{pmatrix} \quad (10)$$

with

$$\zeta = \sqrt{-\det(\tau R A R^T)} = \tau \sqrt{\gamma^2 - \omega^2}. \quad (11)$$

Note that this expression is valid for  $\gamma^2 < \omega^2$ , as well as for  $\gamma^2 = \omega^2$  by taking the limit. The latter case corresponds to a shear flow, since then  $A^2 = (\gamma^2 - \omega^2)I = 0$  with  $A \neq 0$ .

To simplify expressions, we let

$$\Gamma = \gamma\tau/\zeta, \quad \Omega = \omega\tau/\zeta, \quad (12)$$

whence (10) becomes

$$\exp(\tau RAR^T) = \begin{pmatrix} \cosh \zeta + \Gamma \cos 2\theta \sinh \zeta & (\Gamma \sin 2\theta - \Omega) \sinh \zeta \\ (\Gamma \sin 2\theta + \Omega) \sinh \zeta & \cosh \zeta - \Gamma \cos 2\theta \sinh \zeta \end{pmatrix}. \quad (13)$$

The matrix  $RAR^T$  can represent an arbitrary 2D linear flow: there are 3 free parameters  $(\theta, \gamma, \omega)$ , which is the same as the number of independent components of a traceless 2D matrix. Now we assume that the flow *renovates*: for fixed  $\gamma$  and  $\omega$ , we choose a uniformly-distributed random angle  $\theta \in [0, 2\pi)$ . We allow this flow to act for a time  $\tau$ , and after that period we select a new, independent random angle and start over. The random angle  $\theta$  allows us to make analytic progress, and to compute explicitly quantities such as Lyapunov exponents.

Equation (7) is linear in  $\delta\mathbf{x}$ , so the initial length of  $\delta\mathbf{x}$  is irrelevant and doesn't have to be infinitesimal. Moreover, the angle  $\theta$  is random, so we may choose for  $\delta\mathbf{x}$  a vector  $\boldsymbol{\ell} = (1 \ 0)$  that lies along the  $x$  axis with unit length. Then after one step it is transformed to the vector

$$\boldsymbol{\ell}' = \exp(\tau RAR^T) \cdot \boldsymbol{\ell} = (\cosh \zeta + \Gamma \cos 2\theta \sinh \zeta \quad (\Gamma \sin 2\theta + \Omega) \sinh \zeta) \quad (14)$$

which is just the first column of (13). The length of the transformed vector is

$$\begin{aligned} \|\boldsymbol{\ell}'\|^2 &= (\cosh \zeta + \Gamma \cos 2\theta \sinh \zeta)^2 + (\Gamma \sin 2\theta + \Omega)^2 \sinh^2 \zeta \\ &= \cosh^2 \zeta + \Gamma \cos 2\theta \sinh 2\zeta + (\Gamma^2 + \Omega^2 + 2\Gamma\Omega \sin 2\theta) \sinh^2 \zeta. \end{aligned}$$

To find the Lyapunov exponent, we need to average  $\log\|\boldsymbol{\ell}'\|$  over  $\theta$ . Write

$$\|\boldsymbol{\ell}'\|^2 = a + b \sin 2\theta + c \cos 2\theta \quad (15)$$

with

$$a = \cosh^2 \zeta + (\Gamma^2 + \Omega^2) \sinh^2 \zeta, \quad b = 2\Gamma\Omega \sinh^2 \zeta, \quad c = \Gamma \sinh 2\zeta. \quad (16)$$

The logarithm of the length is then

$$\begin{aligned} 2 \log\|\boldsymbol{\ell}'\| &= \log(a + b \sin 2\theta + c \cos 2\theta) \\ &= \log a + \log(1 + (b/a) \sin 2\theta + (c/a) \cos 2\theta) \\ &= \log a + \log(1 + \alpha \cos(2\theta + \beta)) \end{aligned} \quad (17)$$

where  $\beta$  is some phase, and

$$\alpha^2 = (b^2 + c^2)/a^2 = 1 - (\Gamma^2 \cosh 2\zeta - \Omega^2)^{-2}, \quad 0 \leq \alpha < 1. \quad (18)$$

Note that  $\alpha$  is zero if and only if  $\gamma$  is zero. Now we average over  $\theta$ :

$$2\langle \log\|\boldsymbol{\ell}'\| \rangle = \log a + \frac{1}{2\pi} \int_0^{2\pi} \log(1 + \alpha \cos(2\theta + \beta)) d\theta. \quad (19)$$

The phase  $\beta$  is inconsequential, so we drop it and evaluate the integral:

$$\begin{aligned} 2\langle \log\|\boldsymbol{\ell}'\| \rangle &= \log a + \frac{1}{\pi} \int_0^\pi \log(1 + \alpha \cos \psi) d\psi \\ &= \log a + \log\left(\frac{1}{2}(1 + \sqrt{1 - \alpha^2})\right) \\ &= \log\left(\frac{1}{2}a(1 + \sqrt{1 - \alpha^2})\right). \end{aligned}$$

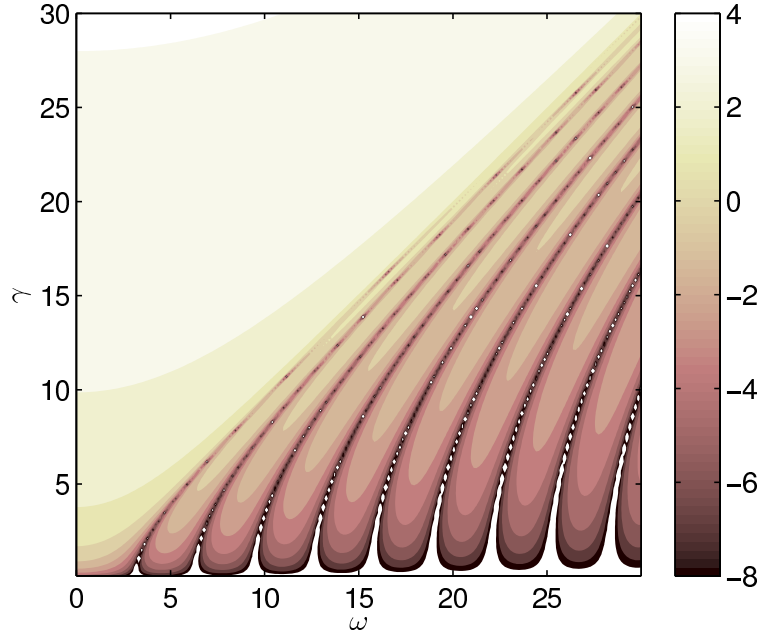


FIG. 1. Contour plot of the logarithm of the Lyapunov exponent (20) for a renovating randomly-oriented linear flow with period  $\tau = 1$ , as a function of the strain rate  $\gamma$  and half-vorticity  $\omega$ .

After some manipulation, we obtain the simple form

$$\lambda = \frac{1}{\tau} \langle \log \|\ell'\| \rangle = \frac{1}{2\tau} \log \left( \frac{\gamma^2 \cosh^2(\tau \sqrt{\gamma^2 - \omega^2}) - \omega^2}{\gamma^2 - \omega^2} \right), \quad \gamma > \omega, \quad (20)$$

for the (positive) Lyapunov exponent  $\lambda$ . This is clearly positive for  $\gamma^2 > \omega^2$ . The expression is also valid for the ‘vortical’ case  $\omega^2 > \gamma^2$ , but then it is preferable to write

$$\lambda = \frac{1}{2\tau} \log \left( \frac{\omega^2 - \gamma^2 \cos^2(\tau \sqrt{\omega^2 - \gamma^2})}{\omega^2 - \gamma^2} \right), \quad \gamma < \omega. \quad (21)$$

There are three limiting cases of interest:

(i) For  $\omega = 0$ , we get the pure-strain limit

$$\lambda = \frac{1}{\tau} \log \cosh(\tau\gamma), \quad \omega = 0. \quad (22)$$

Since  $\cosh|x| < e^{|x|}$  for  $x \neq 0$ , we have  $\lambda < |\gamma|$  for  $\tau\gamma \neq 0$ . The reorientation of the axes of stretching due to renovation thus always decreases the stretching that would occur due to constant strain, because it takes some time for our line segment to align itself with the new axes. When  $\tau|\gamma| \gg 1$ , we recover  $\lambda = |\gamma|$ , that is, the Lyapunov exponent is equal to the rate-of-strain, since for a long period the segment has plenty of time to re-orient and stretch fully at each period.

(ii) For  $\gamma = 0$ , we get the pure-rotation limit

$$\lambda = 0, \quad \gamma = 0, \quad (23)$$

so at least some strain is required to have a nonzero Lyapunov exponent.

(iii) Finally, for  $\gamma \rightarrow \omega$  we have  $A^2 = (\gamma^2 - \omega^2)I = 0$ , and we get the shear-flow limit:

$$\lambda = \frac{1}{2\tau} \log(1 + \tau^2 \omega^2), \quad \gamma = \omega. \quad (24)$$

Note that even though a simple shear flow does not have a positive exponent (its eigenvalues are zero), a renovating shear flow does: it behaves like a hyperbolic system. This highlights the crucial role of re-orientation as a mechanism in chaotic dynamics.

The magnitude of  $\lambda$  as a function of  $\gamma$  and  $\omega$  is plotted in Fig. 1: Notice the periodic windows where the exponent is zero for  $\omega > \gamma$ . These occur whenever  $\cos^2(\tau\sqrt{\omega^2 - \gamma^2}) = 1$  in (21), or  $\tau\sqrt{\omega^2 - \gamma^2} = m\pi$ ,  $m \in \mathbb{Z}$ . This corresponds to  $\zeta = i\pi m$  in (10), and leads to  $\exp(\tau RAR^T) = (-I)^m$ , with obviously no stretching.



## Lecture 12: Generalized Lyapunov exponents

Recall from last time, for a linear renewing flow:

$$2\langle \log \|\ell'\| \rangle = \log a + \frac{1}{\pi} \int_0^\pi \log(1 + \alpha \cos \psi) d\psi. \quad (1)$$

with

$$a = \cosh^2 \zeta + (\Gamma^2 + \Omega^2) \sinh^2 \zeta, \quad \alpha^2 = 1 - (\Gamma^2 \cosh 2\zeta - \Omega^2)^{-2}, \quad 0 \leq \alpha < 1. \quad (2)$$

More generally, consider the growth rate of  $\|\ell\|^q$ , which is obtained by computing

$$\langle \|\ell'\|^q \rangle = \frac{1}{\pi} \int_0^\pi (a + b \sin 2\theta + c \cos 2\theta)^{q/2} d\theta. \quad (3)$$

Using the same method as before,

$$\langle \|\ell'\|^q \rangle = a^{q/2} \frac{1}{\pi} \int_0^\pi (1 + \alpha \cos \psi)^{q/2} d\psi. \quad (4)$$

The integral can be evaluated in terms of a hypergeometric function,

$$\langle \|\ell'\|^q \rangle = a^{q/2} (1 - \alpha)^{q/2} {}_2F_1 \left( \frac{1}{2}, -\frac{q}{2}; 1; -\frac{2\alpha}{1-\alpha} \right). \quad (5)$$

We then define *generalized Lyapunov exponents* as

$$\ell(q) := \frac{1}{\tau} \log \langle \|\ell'\|^q \rangle = \frac{1}{\tau} \log \left( \left( \frac{1-\alpha}{1+\alpha} \right)^{q/4} {}_2F_1 \left( \frac{1}{2}, -\frac{q}{2}; 1; -\frac{2\alpha}{1-\alpha} \right) \right). \quad (6)$$

This is plotted in Fig. 1: observe that  $\ell(0) = \ell(-2) = 0$ , the curve has a minimum at  $q = -1$ , and it is symmetric about that value. These features all follow from the incompressibility of the flow, as we'll explain below.

For large  $q > 0$ , we can use the saddle point method to carry out the integral (4):

$$\begin{aligned} \langle \|\ell'\|^q \rangle &= a^{q/2} \frac{1}{\pi} \int_0^\pi \exp\left(\frac{1}{2}q \log(1 + \alpha \cos \psi)\right) d\psi \\ &\simeq a^{q/2} \frac{1}{\pi} \int_0^\infty \exp\left(\frac{1}{2}q \log\left(1 + \alpha - \frac{1}{2}\alpha\psi^2\right)\right) d\psi \\ &= a^{q/2} (1 + \alpha)^{q/2} \frac{1}{\pi} \int_0^\infty \exp\left(-\frac{1}{4}q \frac{\alpha}{1+\alpha} \psi^2\right) d\psi \\ &= (a(1 + \alpha))^{q/2} \sqrt{\frac{1 + \alpha}{\pi \alpha q}}, \end{aligned}$$

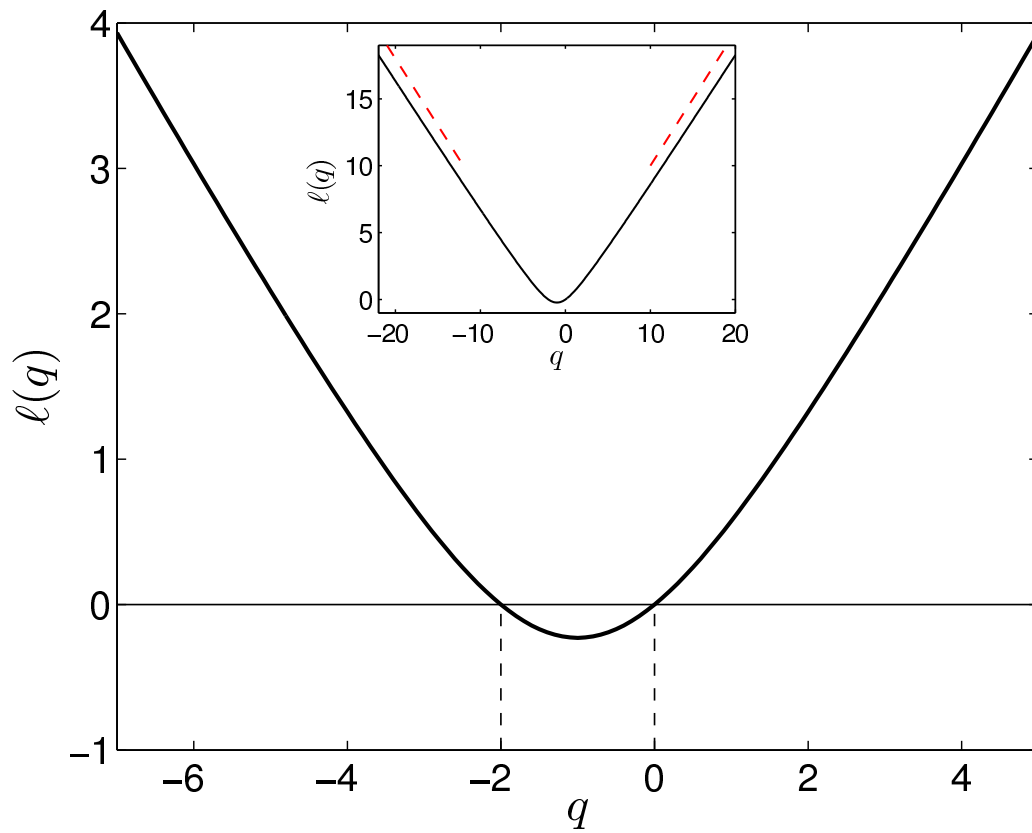


Figure 1: Generalized Lyapunov exponents  $\ell(q)$  from (6) for  $\gamma = 1$ ,  $\omega = 0$ ,  $\tau = 1$ . The inset shows the large- $|q|$  asymptotes  $\frac{1}{4\tau}|q|\log((1+\alpha)/(1-\alpha))$ .

so that

$$\ell(q) = \frac{1}{2\tau} q \log(a(1+\alpha)) - \frac{1}{2\tau} \log q + \frac{1}{2\tau} \log\left(\frac{1+\alpha}{\pi\alpha}\right) + O(q^{-1}), \quad q \gg 1. \quad (7)$$

We can do the same for  $q < 0$ ,  $|q| \gg 1$ ; the saddle point is then at the minimum  $\psi = \pi$ , and we find the leading order form  $\ell(q) \sim \frac{1}{4\tau} |q| \log((1+\alpha)/(1-\alpha))$ , which is shown in an inset to Fig. 1.

The ‘true’ Lyapunov exponent is  $\lambda = \ell'(0)$ :

$$\ell'(0) = \frac{1}{\tau} \left. \frac{\langle \|\ell'\|^q \log \|\ell'\| \rangle}{\langle \|\ell'\|^q \rangle} \right|_{q=0} = \frac{1}{\tau} \langle \log \|\ell'\| \rangle. \quad (8)$$

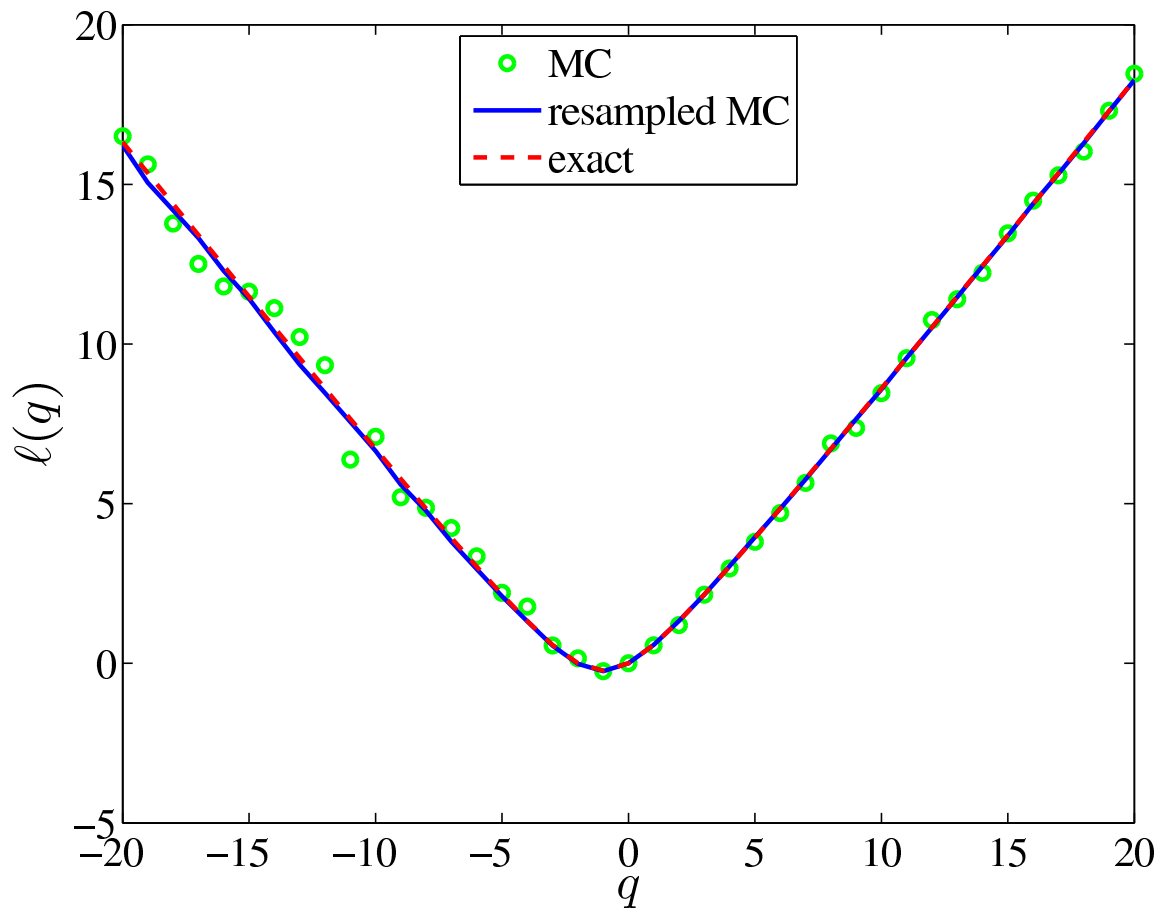


Figure 2: A comparison of resampled Monte–Carlo (J. VANNESTE, *Estimating generalized Lyapunov exponents for products of random matrices*, Phys. Rev. E, 81 (2010), p. 036701) and direct Monte–Carlo for the linear renewing flow, with  $\tau = 1$ ,  $\gamma = 1$ ,  $\omega = 0$ ,  $K = 100$ , and  $N = 100$ . The resampled MC is far better than plain MC, especially for negative  $q$ . See Fig. 3 for the Matlab code used to generate this figure.

```

% Resampled Monte-Carlo to compute the generalized Lyapunov exponents l(q)
% See J. Vanneste, Phys. Rev. E 81, 036701 (2010).
K = 100; N = 100; gamma = 1; omega = 0; q = -20:20;
zeta = sqrt(gamma^2-omega^2);
Gamma = gamma/zeta; Omega = omega/zeta;
cz = cosh(zeta); sz = sinh(zeta);
rng('default');
ev = zeros(K,2); ev(:,1) = 1; % initial vectors [1 0]
evh = zeros(size(ev)); normevh = zeros(1,K); beta = zeros(1,N);
ell = []; ell0 = []; ellth = []; A = zeros(K,2,2);
for iq = 1:length(q)
    for n = 1:N
        % random matrix
        th2 = 2*pi*rand(1,K); ct = cos(th2); st = sin(th2);
        A(:,1,1) = cz + Gamma*ct*sz; A(:,1,2) = (Gamma*st - Omega)*sz;
        A(:,2,1) = (Gamma*st + Omega)*sz; A(:,2,2) = cz - Gamma*ct*sz;
        % multiply matrix ev by matrix A (vectorized over realisations K)
        evh(:,1) = A(:,1,1).*ev(:,1) + A(:,1,2).*ev(:,2);
        evh(:,2) = A(:,2,1).*ev(:,1) + A(:,2,2).*ev(:,2);
        % make unit vector evh
        normevh = sqrt(evh(:,1).^2 + evh(:,2).^2);
        evh(:,1) = evh(:,1)./normevh; evh(:,2) = evh(:,2)./normevh;
        % save q^th power of the norm
        alpha = normevh.^q(iq);
        % resampling
        gamma = cumsum(alpha); beta(n) = gamma(K);
        eps = beta(n)*rand(1,K);
        for k = 1:K
            ii = find(gamma-eps(k) >= 0);
            ev(k,:) = evh(ii(1),:);
        end
    end
    % l(q) without resampling
    ell0 = [ell0 log(mean(alpha))];
    % l(q) with resampling
    ell = [ell mean(log(beta))-log(K)];
    % the analytic expression for l(q)
    aa = sqrt(1 - (Gamma^2*cosh(2*zeta) - Omega^2)^-2);
    ellth = [ellth log(((1-aa)/(1+aa))^(q(iq)/4)*...
        hypergeom([.5 -q(iq)/2],1,-2*aa/(1-aa)))]];
end

plot(q,ell0,'go','LineWidth',2), hold on
plot(q,ell,'-','LineWidth',2)
plot(q,ellth,'r--','LineWidth',2)
legend('MC','resampled MC','exact','Location','North')
xlabel('$q$', 'Interpreter','LaTeX','FontSize',22)
ylabel('$\ell(q)$', 'Interpreter','LaTeX','FontSize',22)
set(gca,'FontSize',18,'FontName','Times')
hold off
print -dpdf resampled_mc.pdf

```

Figure 3: The Matlab code resampled\_mc.m.

(see Vornost, 2010)

Relationship with Cramér function:

$$x_n = A_n x_{n-1}, \quad A_n \in \mathbb{R}^{d \times d} \text{ i.i.d.}$$

$$\|x_0\| = 1, \quad n = 1, \dots, N$$

$$l(q) = \lim_{N \rightarrow \infty} \frac{1}{N} \log \langle \|x_N\|^q \rangle$$

$$= \lim_{N \rightarrow \infty} \frac{1}{N} \log \langle \|A_1 \dots A_N\|^q \rangle$$

$$l(0) = 0.$$

Now consider  $h_N = \frac{1}{N} \log \|A_1 \dots A_N\|$   
(finite-time Lyapunov exponents)

$$\mathcal{P}_N(h) \asymp e^{-Ng(h)}$$

↑ asymptotic equivalence of logs as  $N \rightarrow \infty$

$$g(h) = \text{Cramér function}$$

$$\bar{h} = \lim_{N \rightarrow \infty} H_N = \text{Lyapunov exponent}$$

$g(h)$  is convex with a minimum at  $h = \bar{h}$ .

Now consider:

$$\langle \|x_N\|^q \rangle \simeq \int e^{Nqh} e^{-Ng(h)} dh \simeq e^{Nl(q)}$$

For large  $N$ , can use Laplace's method:

$$e^{N(qh - g(h))} \sim e^{N \sup_h (qh - g(h))}$$

$$l(q) = \sup_h (qh - g(h))$$

So the generalized Lyapunov exponents are the Legendre transform of the Cramer function.

Now consider a function:  $f: \mathbb{R}^d \rightarrow \mathbb{R}$ . Let

$$u_n(x) = \langle f(A_n \dots A_1 x) \rangle$$

$$\text{Then: } u_{n+1}(x) = \langle f(A_{n+1} A_n \dots A_1 x) \rangle$$

$$= \langle f(A_n \dots A_1 A x) \rangle$$

$$= \langle u_n(Ax) \rangle, \quad u_0(x) = f(x)$$

With  $f(x) = \|x\|^q$ ,  $u_n(x_0) = \langle \|x_n\|^q \rangle$ ,

$$u_n(x) = \lambda^n \|x\|^q \nu(\hat{e}), \quad \hat{e} = x / \|x\|$$

Insert:  $\langle u_n(Ax) \rangle = \lambda^n \langle \|Ax\|^q \nu(\underbrace{Ax / \|Ax\|}_{A\hat{e} / \|A\hat{e}\|}) \rangle$

$$= u_{n+1}(x)$$

To get eigenfunction,  
require:

$$= \lambda^{n+1} \|x\|^q \nu(\hat{e})$$

$$\frac{\langle \|Ax\|^q \nu(A\hat{e} / \|A\hat{e}\|) \rangle}{\|x\|^q} = \lambda \nu(\hat{e})$$

$$\langle \|A\hat{e}\|^q \rangle$$

So let:

eigenvalue problem on  $S^{d-1}$

$$(\mathcal{L}_q \nu)(\hat{e}) = \langle \|A\hat{e}\|^q \nu(A\hat{e} / \|A\hat{e}\|) \rangle$$

We then have the eigenvalue problem

$$\boxed{(\mathcal{L}_q \nu)(\hat{e}) = \lambda \nu(\hat{e})} \quad \hat{e} \in S^{d-1}$$



Since  $u_n(x_0) = \langle \|x_n\|^q \rangle$ ,

$$l(q) = \log \lambda_1$$

Largest eigenvalue of  $\mathcal{L}_q$ .

Now let:

$$(\tilde{\mathcal{L}}_q \nu)(\hat{e}) = \langle \|B\hat{e}\|^q \nu(B\hat{e}/\|B\hat{e}\|) \rangle$$

$$\text{with } B = A^{-1}/|\det A|^{1/q}.$$

Let's show that  $\tilde{\mathcal{L}}$  is related to the adjoint of  $\mathcal{L}$  in  $L^2(S^{d-1})$ .

$$\int_{S^{d-1}} \nu(\hat{e}) \mathcal{L} \nu(\hat{e}) d\hat{e} = \left\langle \int_{S^{d-1}} \|A\hat{e}\|^q \nu(\hat{e}) \nu(A\hat{e}/\|A\hat{e}\|) d\hat{e} \right\rangle$$

$$\text{let } \hat{e}' = \frac{A\hat{e}}{\|A\hat{e}\|}, \quad \frac{\partial \hat{e}}{\partial \hat{e}'} = \|A\hat{e}\| A^{-1}$$

$$\det \frac{\partial \hat{e}}{\partial \hat{e}'} = \frac{\|A\hat{e}\|^d}{\det A}$$

$$\begin{aligned} \text{Hence, } d\hat{e} &= \left| \det \frac{\partial \hat{e}}{\partial \hat{e}'} \right| d\hat{e}' \\ &= \frac{\|A\hat{e}\|^d}{|\det A|} d\hat{e}' = \frac{\|A^{-1}\hat{e}'\|^{-d}}{|\det A|} d\hat{e}' \end{aligned}$$

In the last step: used  $\|A\hat{e}\| = \|A^{-1}\hat{e}'\|^{-1}$ , since

$$\|A^{-1}\hat{e}'\|^{-1} = \left\| A^{-1} \frac{(A\hat{e})}{\|A\hat{e}\|} \right\|^{-1} = \|A\hat{e}\|$$

$$\begin{aligned} &\int_{S^{d-1}} w(\hat{e}) \mathcal{L} v(\hat{e}) d\hat{e} \\ &= \left\langle \int_{S^{d-1}} \|A^{-1}\hat{e}'\|^{-g} w(A^{-1}\hat{e}' / \|A^{-1}\hat{e}'\|) v(\hat{e}') \right. \\ &\quad \left. \times \frac{\|A^{-1}\hat{e}'\|^{-d}}{|\det A|} d\hat{e}' \right\rangle \end{aligned}$$

$$= \left\langle \int_{S^{d-1}} \|A^{-1}\hat{e}'\|^{-g-d} w(A^{-1}\hat{e}' / \|A^{-1}\hat{e}'\|) v(\hat{e}') \frac{d\hat{e}'}{|\det A|} \right\rangle$$

Hence,  $\mathcal{L}_g^{\dagger} = \tilde{\mathcal{L}}_{-g-d}$

Hence,  $\mathcal{L}_g$  and  $\tilde{\mathcal{L}}_{-g-d}$  have the same spectrum.

Conclude:

$$\mathcal{l}(g) = \mathcal{l}^{-}(-g-d)$$

↑  
time-reversed system,  $A \rightarrow A^{-1}$   
( $\det A = 1$ )

It follows immediately that  $\mathcal{l}(-d) = \mathcal{l}^{-}(0) = 0$ .

Vanneste also shows that if the matrices are symplectic then  $\mathcal{l}(g) = \mathcal{l}^{-}(-g-d)$

Symplectic matrices satisfy

$$A^T J A = J, \quad J = \begin{pmatrix} 0 & \mathbb{I} \\ -\mathbb{I} & 0 \end{pmatrix} \quad (d \text{ even})$$

$$J^{-1} = J^T = -J$$

↙  $\frac{d}{2} \times \frac{d}{2}$  identity matrix

Note then  $A^{-1} = J^T A^T J = -J A^T J$ ,

since  $(-J A^T J) A = -J (A^T J A) = -J^2 = I$ .

Hence,  $A \overset{\uparrow}{A^{-T}} = -J A J$ , so  $A J = J A^{-T}$ .

$$A^{-T} = (A^{-1})^T$$

Let  $(J\nu)(\hat{e}) = \nu(J\hat{e})$ .

$$\begin{aligned} (J \mathcal{L}_q \nu)(\hat{e}) &= \left\langle \|A J \hat{e}\|^q \nu\left(\frac{A J \hat{e}}{\|A J \hat{e}\|}\right) \right\rangle \\ &= \left\langle \|J A^{-T} \hat{e}\|^q \nu\left(\frac{J A^{-T} \hat{e}}{\|J A^{-T} \hat{e}\|}\right) \right\rangle \\ &= \left\langle \|A^{-T} \hat{e}\|^q \nu\left(\frac{J A^{-T} \hat{e}}{\|A^{-T} \hat{e}\|}\right) \right\rangle \\ &= \left\langle \|A^{-T} \hat{e}\|^q (J \nu)\left(\frac{A^{-T} \hat{e}}{\|A^{-T} \hat{e}\|}\right) \right\rangle \\ &= (\mathcal{L}_q^{-T} J \nu)(\hat{e}) \end{aligned}$$

where  $Z_q^{-T}$  is defined as for  $Z_q$ , but  
with  $A \rightarrow A^{-T}$ .

$$\text{So: } J Z_q = Z_q^{-T} J$$

which implies that  $Z_q$  and  $Z_q^{-T}$  have the  
same spectrum!

$$\text{Hence, } l(q) = l^{-T}(q)$$

But  $l^{-T}(q) = l^{-1}(q)$ , since the transpose  
don't matter in  $\|A_N^{-1} \dots A_1^{-1}\|$  vs  $\|A_N^{-T} \dots A_1^{-T}\|$ .

Conclude:  $l(q) = l(-q-d)$  for symplectic matrices.

Let's finish by showing that

$$\dot{x} = u, \quad \nabla \cdot u = 0 \tag{21}$$

leads to  $\dot{M} = GM$  with  $M = \frac{\delta x}{\delta X}$  symplectic.

$$G = (\nabla u)^T.$$

$$\nabla \cdot u = 0 \Rightarrow u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}$$

$$u = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \partial_x \psi \\ \partial_y \psi \end{pmatrix} = J \cdot \nabla \psi$$

$$\begin{aligned} \nabla u &= J : \nabla \nabla \psi = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \psi_{xx} & \psi_{xy} \\ \psi_{xy} & \psi_{yy} \end{pmatrix} \\ &= \begin{pmatrix} \psi_{xy} & \psi_{yy} \\ -\psi_{xx} & -\psi_{xy} \end{pmatrix} \end{aligned}$$

$$G = (\nabla u)^T = -\nabla \nabla \psi : J$$

$$\Rightarrow \dot{M} = GM = -\nabla \nabla \psi : J$$

Now we want to check:  $M^T J M = J$

Satisfied at  $t=0$  ( $M = I$ )

$$\begin{aligned} \frac{d}{dt} (M^T J M) &= \dot{M}^T J M + M^T J \dot{M} \\ &= M^T G^T J M + M^T J G M \\ &= M^T (G^T J + J G) M \\ &= M^T (J G - (J G)^T) M \end{aligned}$$

Hence we need  $JG$  symmetric for  $M^T J M = \text{const.}$ ,

$$JG = -J \nabla \nabla \psi J \quad \text{obviously symmetric.}$$

Conclude:  $M$  is symplectic.

Note that this works for  $d \geq 2$  ( $d$  even)  
for

$$u = J \cdot \nabla \psi$$

This defines Hamiltonian systems.

## Lecture 13: Rate of decay and local stretching

Recall from many lectures ago that for a 2D, extensional, incompressible flow, the concentration of a passive scalar:

$$\theta(x, y, t) \sim e^{-\lambda t} \quad \lambda = \text{rate of strain}$$

with a Gaussian cross-section.

Now imagine the blob is being subjected to a random renewing flow.

Assume that  $\tau$  (the correlation time) is large enough that our Gaussian blob aligns rapidly:

$$\lambda \tau \gg 1$$

Then at each application the intensity of  $\theta$  decays by a factor  $e^{-\lambda \tau}$

Consider  $\langle |\theta|^p \rangle$ , where the expected value is over the random matrices in our flow.



Then:

$$\langle |\theta|^p \rangle \sim \int e^{-p h \tau N} e^{-N g(h)} dh$$

↑  
time =  $N\tau$ 
↑  
Cramér function

$$\sim \int e^{-N(p h + g(h))} dh$$

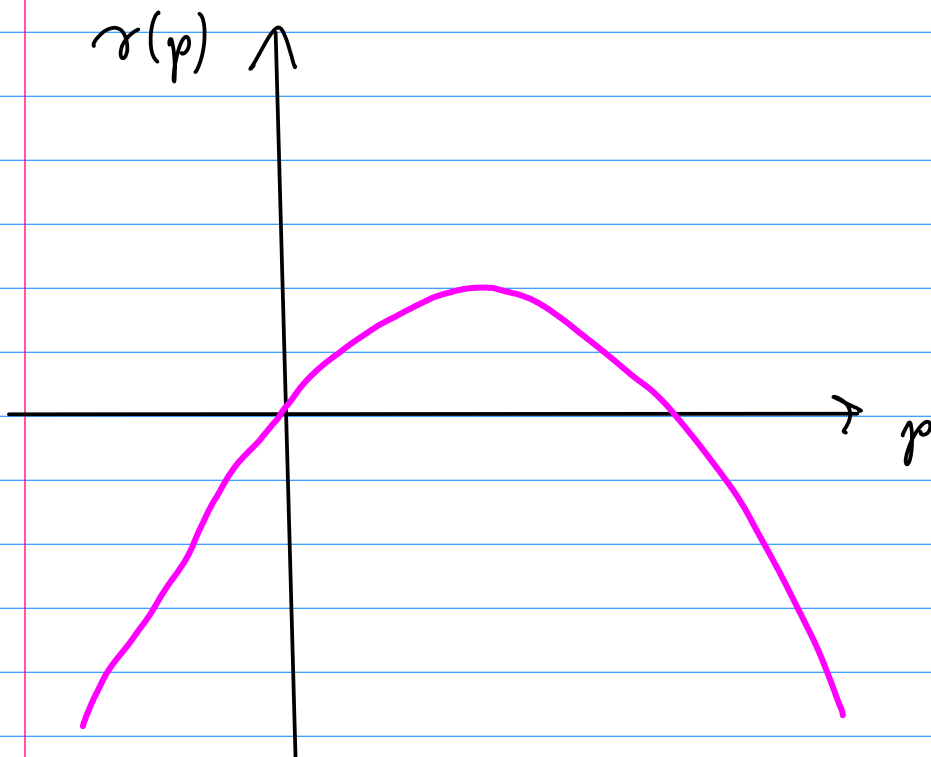
For large  $N$ , the rate of decay is thus

$$\gamma(p) = \frac{1}{\tau} \inf_h (p h + g(h))$$

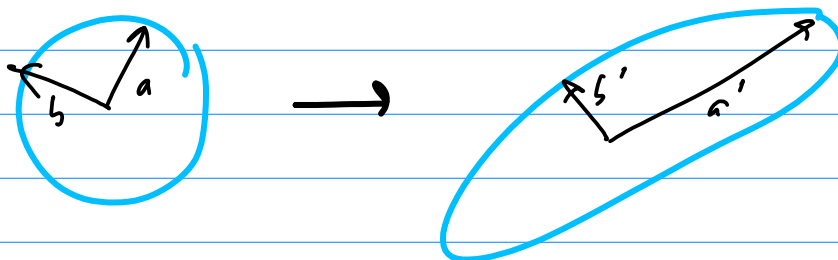
$$\langle |\theta|^p \rangle \sim e^{-\gamma(p)t}, \quad t \rightarrow \infty$$

Thus,  $\gamma(p) = -l(-p)$

↑ generalized Lyapunov exponent



There is one crucial difference: we cannot consider negative  $h$ . Why?  $h$  here is the stretching of the longest axis of the ellipsoid:



This is a nondecreasing quantity, unlike the growth of a line segment  $l$ .

So consider:  $r(p) = \inf_{h>0} (ph + g(h))$

Convex

saddle point

$$p + g'(h_*(p)) = 0$$

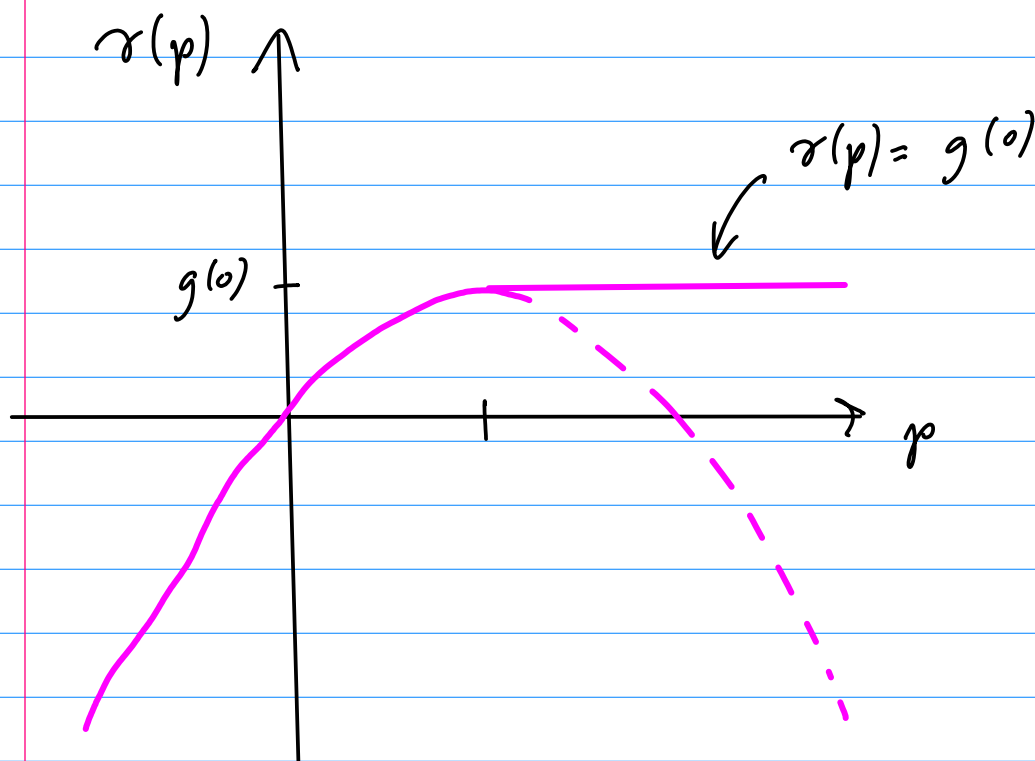
$$r(p) = ph_*(p) + g(h_*(p))$$

$$\begin{aligned} r'(p) &= h_*(p) + g'(h_*(p)) h'_*(p) \\ &\quad + p h'_*(p) \\ &= h'_*(p) \end{aligned}$$

So  $r'(p) = 0$  when  $h'_*(p) = 0 \Rightarrow h=0$ .

So the point where the saddle point coincides with the extremum of  $l(p)$ !

We must amend our figure for  $r(p)$  to reflect this:

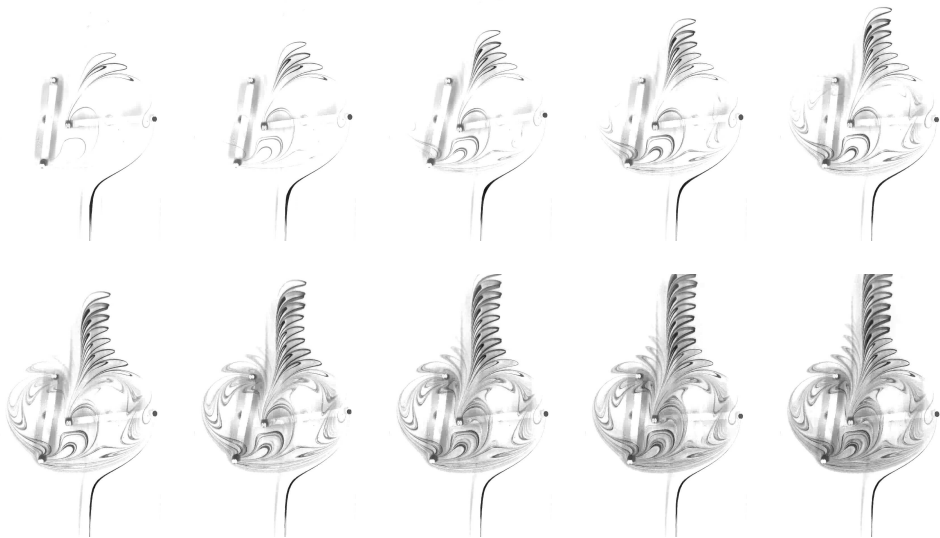


$$\text{Thus, } r(p) = \begin{cases} \inf_h (ph + g(h)) & , h_*(p) > 0 \\ g(0) & , h_*(p) < 0 \end{cases}$$

The decay of  $\langle |0|^p \rangle$  for large  $p$  is thus completely dominated by realizations with zero stretching.

→ becomes independent of  $p$ .

# Lecture 14: Strange eigenmodes and intermittency



play movie



- Gouillart, E., Dauchot, O., Thiffeault, J.-L., & Roux, S. (2009). *Phys. Fluids*, **21** (2), 022603.
- Haynes, P. H. & Vanneste, J. (2005). *Phys. Fluids*, **17**, 097103.
- Pierrehumbert, R. T. (1994). *Chaos Solitons Fractals*, **4** (6), 1091–1110.
- Rothstein, D., Henry, E., & Gollub, J. P. (1999). *Nature*, **401** (6755), 770–772.
- Vanneste, J. (2006). *Phys. Fluids*, **18**, 087108.

## Lecture 14: Strange eigenmodes & intermittency

Following Vanneste (2006), consider

$$\underline{u} = \begin{pmatrix} 0 \\ V(x, t) \end{pmatrix}$$

and concentration  $C(x, y, t)$ .

$$\partial_t C + V C_y = \kappa (C_{xx} + C_{yy})$$

Assume  $2\pi$ -periodic in  $x$  and  $y$ . Let

$$C(x, y, t) = \operatorname{Re} \left[ e^{i l y - n l^2 t} \hat{C}(x, t) \right]$$

Then

$$\hat{C}_t + i l V(x, t) \hat{C} = \kappa \hat{C}_{xx}$$

The velocity, being a function of  $x$  only, does not couple  $y$  modes

$$\hat{C}_t = f(\hat{C}, t)$$

Think of this as an  $\infty$ -dimensional dyn. sys

For long times, Oseledec says

$$\hat{C}(x, t) \sim D(x, t) e^{-\lambda t}, \quad t \rightarrow \infty$$

$\uparrow$   
 $\lim_{t \rightarrow \infty} \frac{\log D}{t} \rightarrow 0$  subexponential

Where  $\lambda = -\lim_{t \rightarrow \infty} \frac{1}{t} \log \frac{\|\hat{C}(t)\|}{\|\hat{C}(0)\|}$   $\leftarrow L_2$

$-\lambda$  is the largest Lyapunov exponent for the  $\hat{C}$  eq'n.

Finite time:  $\lambda_t = -\frac{1}{t} \log \frac{\|\hat{C}(t)\|}{\|\hat{C}(0)\|}$

Moment decay rates:

$$r_p = -\lim_{t \rightarrow \infty} \frac{1}{t} \log \frac{\langle \|\hat{C}\|^p \rangle(t)}{\langle \|\hat{C}\|^p \rangle(0)}$$

Recall also:  $r_p = \inf_{\lambda_t} [p \lambda_t + g(\lambda_t)]$

$\uparrow$   
 ensemble  
 average



$$\lambda_t \rightarrow \lambda \quad \text{as } t \rightarrow \infty$$

but  $r_p \neq p\lambda$  in general.

This is called temporal intermittency

Let's consider two models that show intermittency.

type 1:  $V(x, t) = f(t) \sin x$

$$\text{with } f(t) = \frac{\alpha}{l} \xi_n, \quad n \leq t < n+1 \\ n = 0, 1, 2, \dots$$

$\xi_n$  are i.i.d. Gaussian (mean 0, unit var.)

This is a renewing flow.

With  $\kappa = 0$ , the equation

$$\hat{C}_t = -il V(x, t) \hat{C}_x$$

can be solved from  $t = nt$  to  $(n+1)t$  as

$$\hat{C}(x, n+1) = e^{-i l \underbrace{V(x, n)}_{\substack{\downarrow \\ \text{constant over interval}}} } \hat{C}(x, n)$$

$n \leq t < n+1$

Let  $\hat{C}_n = \hat{C}(x, n)$ .

The solution with diffusion is tougher. It is more convenient to use pulsed diffusion:

$$\hat{C}_{n+1}(x) = e^{\kappa \frac{\partial^2}{\partial x^2}} \left[ e^{-i l V(x, n)} \hat{C}(x, n) \right]$$

This is easy to integrate numerically: see

ren flow ('type 1')

[matlab]

for an example

Note that the realizations plotted all decay at roughly the same rate, but there are fluctuations.

The moments decay exponentially, but the rate  $\rightarrow 0$  as  $\kappa \rightarrow 0$ .

Animation:

ren flow ('anim 1')

The animation clearly shows that  $\hat{C}$  converges to a kind of "eigenmode", though the phase fluctuates.

These are often called generalized eigenmodes or eigenmodes in the sense of Oseledec.

Look for intermittency:

renflow\_gamma

Very weak intermittency (deviation from linear)

The peaks in  $\hat{C}$  get narrower as  $n \rightarrow 0$ .

Vanneste does a boundary-layer analysis to find

$$\lambda = \tau_1 \sim 0.460 (n\alpha)^{2/3}$$

$$\tau_2 \sim 0.881 (n\alpha)^{2/3}$$

Let's look a bit at what Vanneste did.

The decay is slow compared to the period.

Suggests modeling as stochastic equation with white noise:

Stratonovich ( $W_t =$  Wiener process)

$$\hat{C}_t + i\alpha \sin x \hat{C}_0 \dot{W}_t = \kappa \hat{C}_{xx}$$

Rescale:  $x = \frac{\pi}{2} + \kappa^{1/6} \alpha^{-1/3} X$ ,  $t = (\kappa \alpha)^{-2/3} T$

$$\hat{C}(x, t) = e^{-i\alpha W_t} \tilde{C}(X, T)$$

$$\hat{C}_t = e^{-i\alpha W_t} \left( -i\alpha \tilde{C}_0 \dot{W}_t + \tilde{C}_T (\kappa \alpha)^{2/3} \right)$$

$$\hat{C}_{xx} = e^{-i\alpha W_t} \tilde{C}_{XX} (\kappa^{-1/6} \alpha^{1/3})^2$$

$$\begin{aligned} -i\alpha \dot{W}_t \tilde{C} + \tilde{C}_T (\kappa \alpha)^{2/3} + i\alpha \sin\left(\frac{\pi}{2} + \kappa^{1/6} \alpha^{-1/3} X\right) \tilde{C}_0 \dot{W}_t \\ = (\kappa^{2/3} \alpha^{2/3}) \tilde{C}_{XX} \end{aligned}$$

Use  $\sin\left(\frac{\pi}{2} + \varepsilon\right) = 1 - \frac{\varepsilon^2}{2} + o(\varepsilon^4)$ :

$$\begin{aligned} \cancel{i\alpha \tilde{C}_0 \dot{W}_t} + (\kappa \alpha)^{2/3} \tilde{C}_T + \\ i\alpha \left(1 - \kappa^{1/3} \alpha^{-2/3} X^2\right) \tilde{C}_0 \dot{W}_t = (\kappa \alpha)^{2/3} \tilde{C}_{XX} \end{aligned}$$

$$\tilde{C}_T - \frac{i}{2} (k\alpha)^{-1/3} X^2 \tilde{C} \circ \dot{W}_t = \tilde{C}_{XX}$$

But note that  $t = (k\alpha)^{-2/3} T$ ,

$$\sqrt{t} = (k\alpha)^{-1/3} \sqrt{T}$$

$W_t$  satisfies a "Brownian scaling":

$$W_t = \frac{1}{\sqrt{c}} W_{ct}$$

So if  $c = (k\alpha)^{-2/3}$ ,  $W_t = (k\alpha)^{1/3} W_T$ ,  
and hence

$$\tilde{C}_T - \frac{1}{2} i X^2 \tilde{C} \circ \tilde{W}_T = \tilde{C}_{XX}.$$

This shows that the width of the boundary layer  
scales as  $k^{1/6} \alpha^{2/3}$ .

It also gives us the time scale  $(k\alpha)^{-2/3}$ .

$$\tilde{C}(X, T) = e^{- (a(T)X^2 + b(T))}$$

$$\tilde{C}_X = -2Xa(T)\tilde{C}$$

$$\tilde{C}_{XX} = (-2a(T) + 4X^2a^2(T))\tilde{C}$$

$$\tilde{C}_T = \tilde{C}(-\dot{a}X^2 - \dot{b})$$

$$\begin{aligned} -\dot{a}X^2 - \dot{b} - \frac{i}{2}X^2\dot{W}_T \\ = -2a + 4a^2X^2 \end{aligned}$$

$$\text{Hence: } \left. \begin{aligned} \dot{a} &= -4a^2 - \frac{i}{2}\dot{W}_T \\ \dot{b} &= 2a \end{aligned} \right\} \text{coupled SDEs}$$

Now consider model type 2:

$$V(x, t) = \frac{\alpha}{l} \sin(x + \phi(t))$$

$$\phi(t) = \phi_n \in [0, 2\pi), \quad n \leq t < n+1$$

ren flow ('type 2')

ren flow ('anim 2')

This exhibits a lot more intermittency (see movies, figures at the end).

$$\sin(x + \phi(t)) = \sin x \cos \phi + \cos x \sin \phi$$

Note that  $E(\cos \phi) = E(\sin \phi) = 0$

$$E(\sin^2 \phi) = E(\cos^2 \phi) = \frac{1}{2}, \quad E(\sin \phi \cos \phi) = 0$$

So approximate  $\sin(x + \phi) \simeq \frac{\dot{W}_t^1}{\sqrt{2}} \sin x + \frac{\dot{W}_t^2}{\sqrt{2}} \cos x$

$\dot{W}_t^1, \dot{W}_t^2$  independent Wiener processes.

Hence

$$\hat{C}_t + \frac{i\alpha}{\sqrt{2}} \hat{C}_0 (\dot{W}_t^1 \sin x + \dot{W}_t^2 \cos x) = \kappa \hat{C}_{xx}$$

$$\text{let } \hat{C}(x,t) = \rho(x,t) e^{i\theta(x,t)} \quad \rho \in \mathbb{R}$$

$$\hat{C}_t = (\rho_t + i\theta\rho) \hat{C}$$

$$\hat{C}_x = (\rho_x + i\rho\theta_x) \hat{C}$$

$$\hat{C}_{xx} = (\rho_{xx} + 2i\rho_x\theta_x + i\rho\theta_{xx} - \rho\theta_x^2) \hat{C}$$

$$\Rightarrow \rho_t = \kappa\rho_{xx} - \kappa\rho\theta_x^2$$

$$\theta_t = -\frac{\alpha}{\sqrt{2}} (\dot{W}_t^1 \sin x + \dot{W}_t^2 \cos x) + \kappa \left( \theta_{xx} + \frac{2\rho_x\theta_x}{\rho} \right)$$

For short time, diffusion can be neglected,

The phase is then given by



$$\theta(x, t) = -\frac{\alpha}{\sqrt{2}} \left( W_t^1 \sin x + W_t^2 \cos x \right)$$

( $\theta(x, 0) = 0$ )

Hence,  $E \theta^2 \sim t$  The phase "diffuses"

Note that  $\rho_t = \underbrace{\kappa \rho_{xx}}_{\substack{\text{neglect} \\ \text{for now}}} - \kappa \rho \theta_x^2$  ( $\rho = \text{constant}$  for  $x=0$ )

$-\kappa \int_0^t \theta_x^2 dt$

$$\rho_t \approx -\kappa \rho \theta_x^2 \Rightarrow \rho(x, t) \approx \rho_0(x) e^{-\kappa \int_0^t \theta_x^2 dt}$$

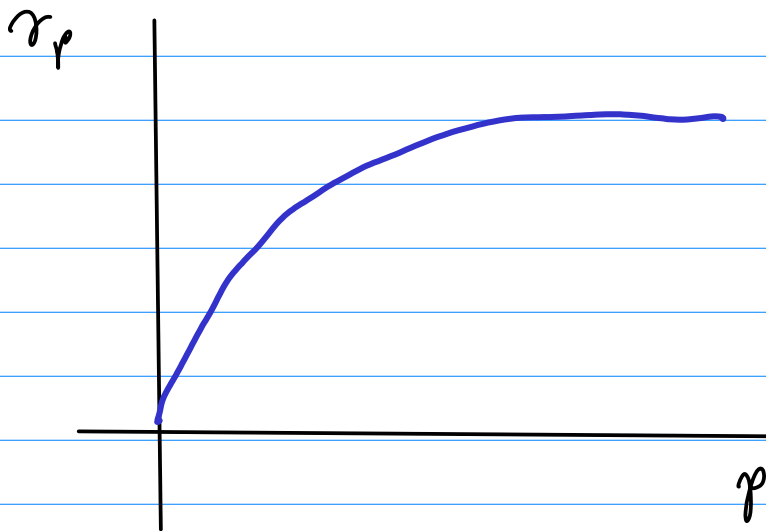
This describes the early stages of the evolution.

$$\rho \sim e^{-\kappa t^2} \quad \leftarrow \text{not an eigenmode}$$

Vanneste shows in Appendix B that

$$E \rho^p(x, t) = \frac{\rho_0^p(x)}{\cosh^{1/2} [\alpha (np)^{1/2} t]}$$

$$\sim \rho_0^p(x) e^{-\alpha (np)^{1/2} t/2}, \quad t \gg 1$$



Note that this agrees reasonably well with the numerical sim, even though we used an "early time" approx.

This is because  $r_p$  is dominated by the rare realizations with low stretching, which are in the "early time" regime.

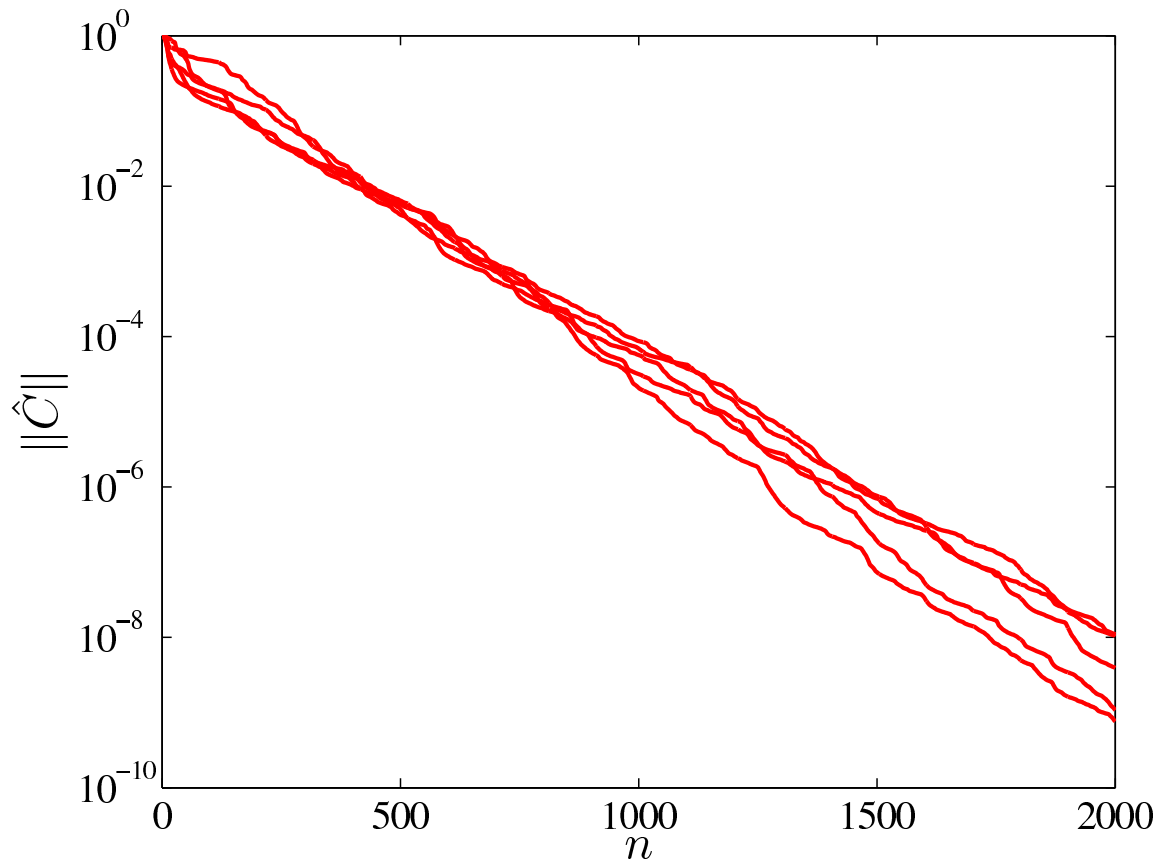


Figure 1: Decay of the concentration  $L_2$  norm for the Type 1 flow (random amplitude) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108, for five realizations.

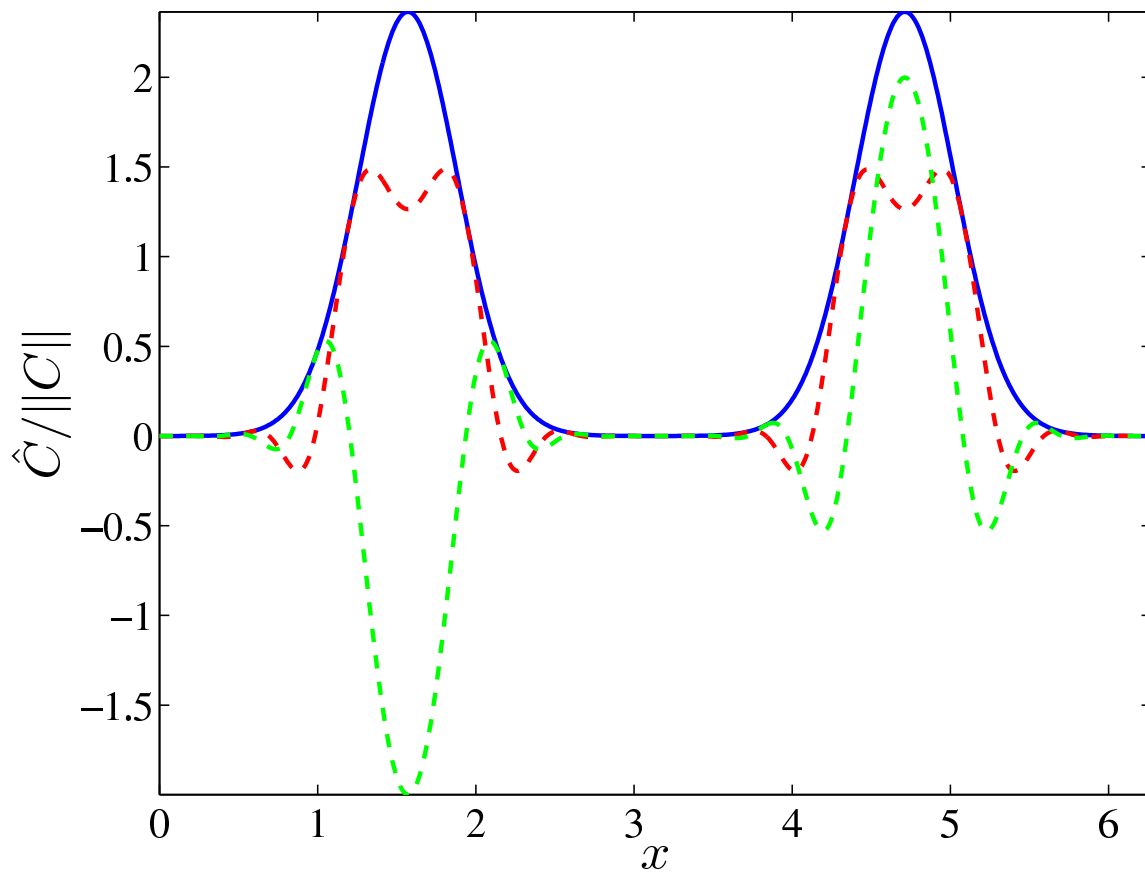


Figure 2: Generalized eigenmode of the Type 1 flow (random amplitude) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108.

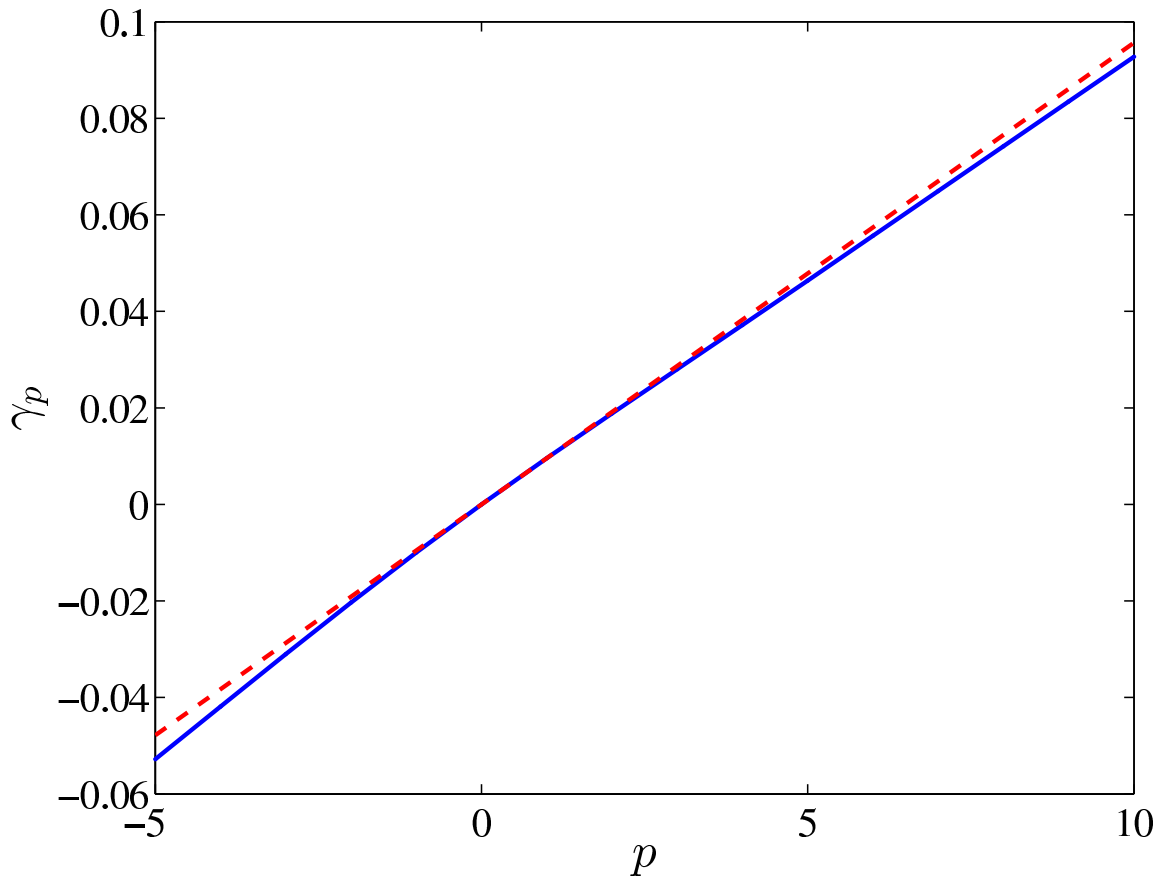


Figure 3: Weak intermittency of the Type 1 flow (random amplitude) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108. The dashed line is the linear scaling.

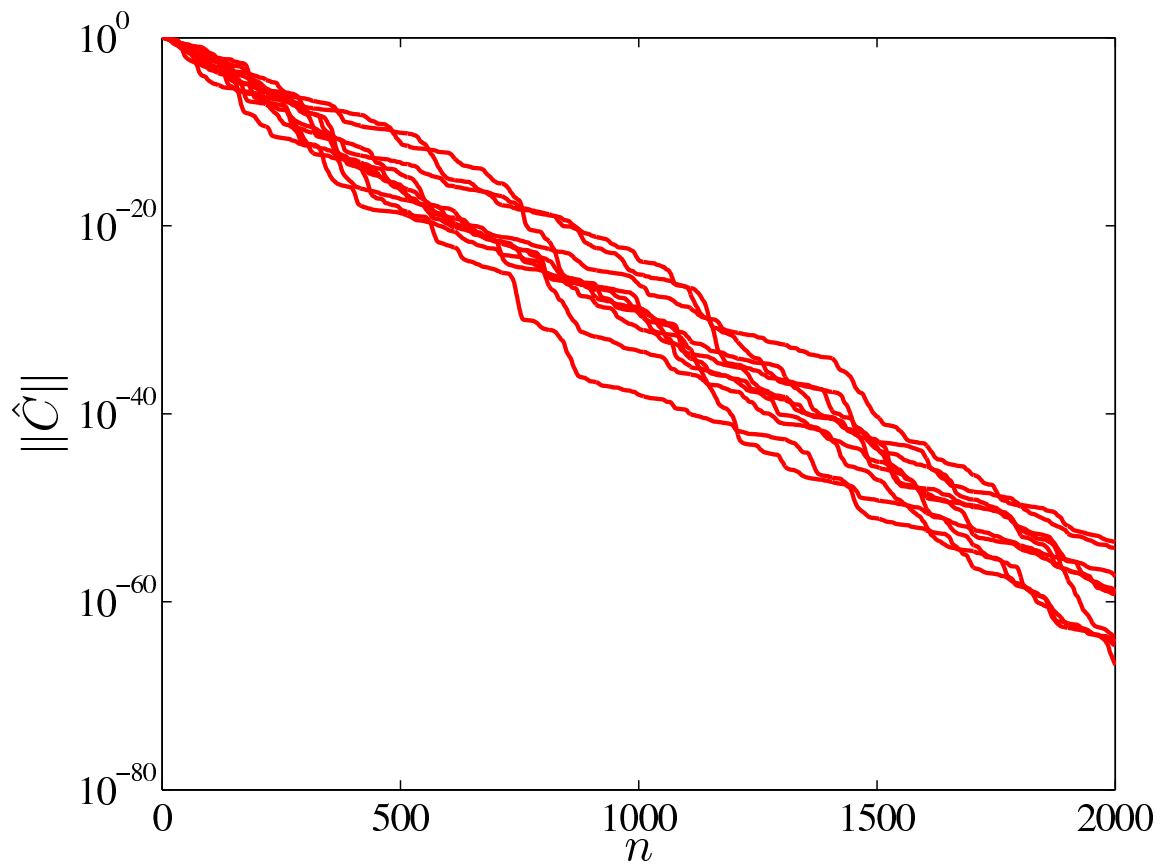


Figure 4: Decay of the concentration  $L_2$  norm for the Type 2 flow (random phase) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108, for five realizations.

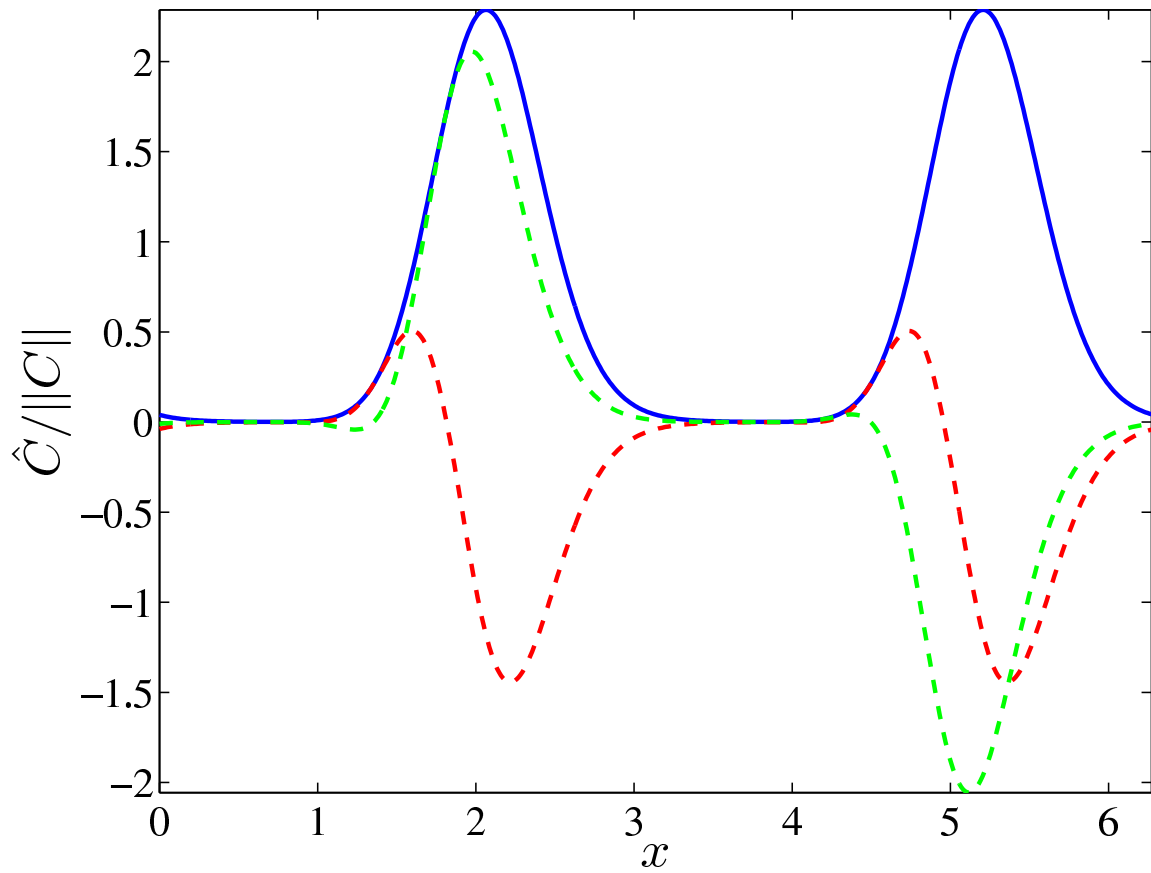


Figure 5: Generalized eigenmode of the Type 2 flow (random phase) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108.

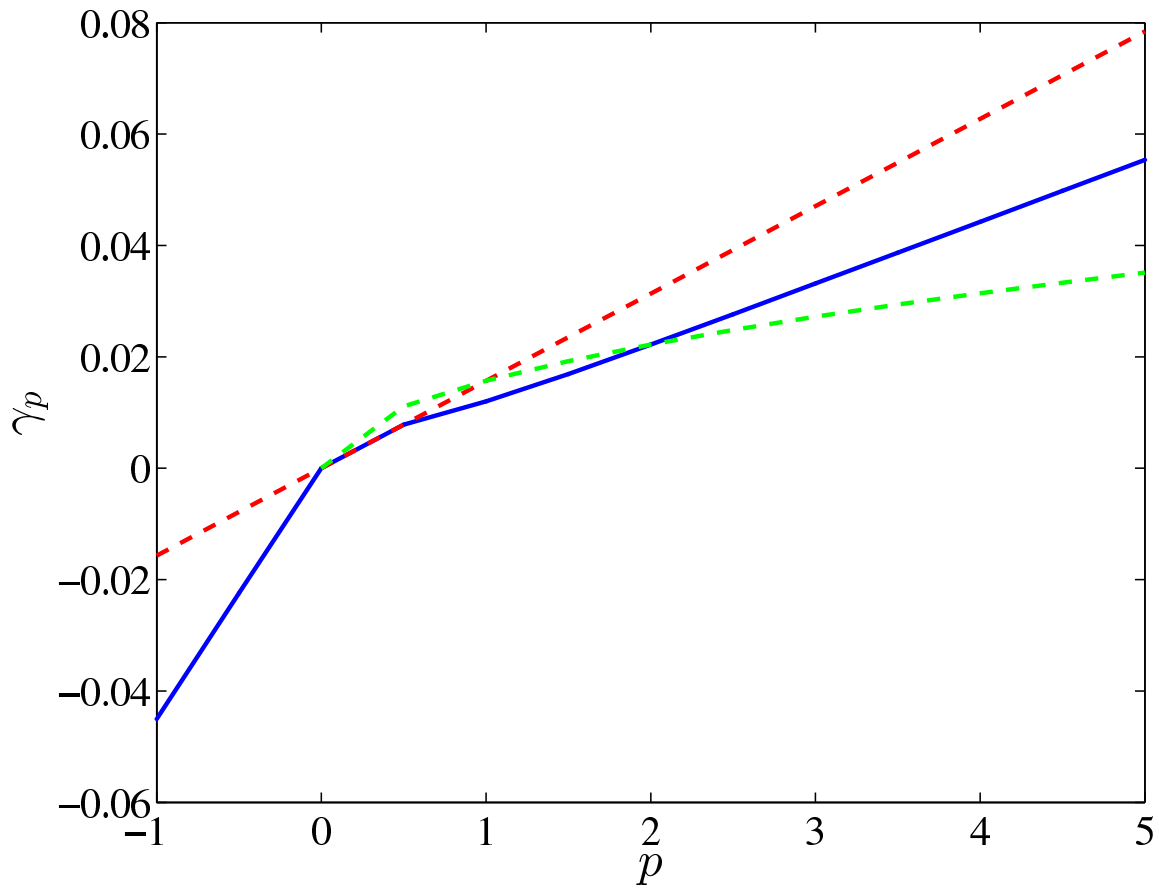


Figure 6: Weak intermittency of the Type 2 flow (random phase) in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108. The dashed line is the linear scaling.



```

function Cnorm = renflow

type = 1; % the two flows in Vanneste: type 1 or 2
Nstep = 2000; Nreal = 100; N = 512;
kappa = 1e-3; alpha = pi; p = -1:2;

kmin = floor(-(N-1)/2); kmax = floor((N-1)/2); k = [0:kmax kmin:-1];
x = linspace(0,2*pi,N+1); x = x(1:end-1);

switch type
case 1
    advec = @(C) exp(-i*alpha*randn(Nreal,1)*sin(x));
case 2
    advec = @(C) exp(-i*alpha*sin(tensorsum(x,2*pi*rand(Nreal,1)))));
end

Cnorm = zeros(Nreal,Nstep+1,length(p)); C = ones(Nreal,N);
Cnorm(:,1,:) = Cnorms(C,p);
diff = diag(sparse(exp(-kappa*k.^2)));
for n = 1:Nstep
    C = advec().*C; % advection step
    Ck = fft(C,[],2); % Fourier transform
    Ck = Ck*diff; % diffusion step
    C = ifft(Ck,[],2); % inverse Fourier transform
    Cnorm(:,n+1,:) = Cnorms(C,p);
end
Cnorm = squeeze(mean(Cnorm)); % average over realizations

%=====
function Cp = Cnorms(C,p)

Cp = zeros(size(C,1),length(p));
for ip = 1:length(p)
    Cp(:,ip) = sqrt(sum(C.*conj(C),2)/size(C,2)).^p(ip);
end

```

Figure 7: A simplified version of the Matlab code `renflow.m`, which implements the evolution of a passive scalar stirred by the two model flows in J. VANNESTE, *Intermittency of passive-scalar decay: Strange eigenmodes in random shear flows*, Phys. Fluids, 18 (2006), p. 087108.

## Lecture 15: Homogenization Theory

### I. MULTISCALE EXPANSION AND HOMOGENIZATION

We start with the advection–diffusion equation,

$$\partial_t \varphi(t, \mathbf{r}) + \mathbf{u}(\mathbf{r}) \cdot \nabla_{\mathbf{r}} \varphi(t, \mathbf{r}) = D \Delta_{\mathbf{r}} \varphi(t, \mathbf{r}). \quad (\text{I.1})$$

Assume typical lengthscale of  $\mathbf{u}$  is  $\ell$ , and that the initial condition varies on a scale  $L$  that is large with respect to  $\ell$ . Define  $\varepsilon = \ell/L \ll 1$ . We write  $\varphi(0, \mathbf{r}) = \varphi_0(\varepsilon \mathbf{r})$ .

Now introduce the large scale and slow time,

$$\mathbf{R} = \varepsilon \mathbf{r}, \quad T = \varepsilon^2 t, \quad (\text{I.2})$$

and assume that the concentration depends on these scales,

$$\varphi(t, \mathbf{r}) = \varphi^\varepsilon(T, \mathbf{r}, \mathbf{R}). \quad (\text{I.3})$$

Using  $\partial_t \rightarrow \varepsilon^2 \partial_T$ ,  $\nabla_{\mathbf{r}} \rightarrow \nabla_{\mathbf{r}} + \varepsilon \nabla_{\mathbf{R}}$ , Eq. (I.1) becomes

$$\mathcal{L} \varphi^\varepsilon + \varepsilon^2 \partial_T \varphi^\varepsilon + \varepsilon \mathbf{u}(\mathbf{r}) \cdot \nabla_{\mathbf{R}} \varphi^\varepsilon = 2\varepsilon D \nabla_{\mathbf{r}} \cdot \nabla_{\mathbf{R}} \varphi^\varepsilon + \varepsilon^2 D \Delta_{\mathbf{R}} \varphi^\varepsilon \quad (\text{I.4})$$

where the velocity field is assumed to only depend on the short lengthscale  $\mathbf{r}$ , and we have defined the linear operator

$$\mathcal{L} := -D \Delta_{\mathbf{r}} + \mathbf{u} \cdot \nabla_{\mathbf{r}}. \quad (\text{I.5})$$

We expand the concentration in a power series in  $\varepsilon$ ,

$$\varphi^\varepsilon(T, \mathbf{r}, \mathbf{R}) = \varphi^{(0)}(T, \mathbf{r}, \mathbf{R}) + \varepsilon \varphi^{(1)}(T, \mathbf{r}, \mathbf{R}) + \dots \quad (\text{I.6})$$

and at order  $\varepsilon^0$  obtain from Eq. (I.4),

$$\mathcal{L} \varphi^{(0)} = 0. \quad (\text{I.7})$$

The solution to (I.7) is  $\varphi^{(0)}(T, \mathbf{r}, \mathbf{R}) = \Phi(T, \mathbf{R})$ .

At order  $\varepsilon^1$ , Eq. (I.4) with the expansion (I.6) gives

$$\mathcal{L} \varphi^{(1)} + \mathbf{u} \cdot \nabla_{\mathbf{R}} \Phi = 0. \quad (\text{I.8})$$

We introduce the cell-average of a function  $f$ ,

$$\langle f \rangle := \frac{1}{V} \int_{\Omega} f \, d^3 r, \quad V := \int_{\Omega} d^3 r, \quad (\text{I.9})$$

and cell-average Eq. (I.8), using  $\langle \mathcal{L} \varphi^{(1)} \rangle = 0$ , to obtain

$$\langle \mathbf{u} \rangle \cdot \nabla_{\mathbf{R}} \Phi = 0 \quad (\text{I.10})$$

which is satisfied for  $\langle \mathbf{u} \rangle = 0$ .

From Eqs. (I.8) and (I.10) we must solve

$$\mathcal{L}\varphi^{(1)} + \mathbf{u} \cdot \nabla_{\mathbf{R}}\Phi = 0. \quad (\text{I.11})$$

The solution to this is  $\varphi^{(1)} = \boldsymbol{\chi}(\mathbf{r}) \cdot \nabla_{\mathbf{R}}\Phi$ , where

$$\mathcal{L}\boldsymbol{\chi} + \mathbf{u} = 0, \quad (\text{I.12})$$

the so-called *cell problem*. Note that we must have  $\langle \mathcal{L}\boldsymbol{\chi} \rangle = 0$  for the cell problem to have a solution, and that  $\boldsymbol{\chi}$  is not unique since we can add a constant to it. Without loss of generality, choose  $\langle \boldsymbol{\chi} \rangle = 0$ .

Assuming the cell problem (I.12) has been solved, we can proceed to order  $\varepsilon^2$  in Eq. (I.4),

$$\mathcal{L}\varphi^{(2)} + \partial_T\Phi + \mathbf{u} \cdot \nabla_{\mathbf{R}}\varphi^{(1)} = 2D\nabla_{\mathbf{r}} \cdot \nabla_{\mathbf{R}}\varphi^{(1)} + D\Delta_{\mathbf{R}}\Phi. \quad (\text{I.13})$$

Cell-averaging (I.13) and using  $\langle \mathcal{L}\varphi^{(2)} \rangle = 0$ , we find

$$\partial_T\Phi + \nabla_{\mathbf{R}} \cdot (\langle \mathbf{u}\boldsymbol{\chi} \rangle \cdot \nabla_{\mathbf{R}}\Phi) = 2D\nabla_{\mathbf{R}} \cdot (\langle \nabla_{\mathbf{r}}\boldsymbol{\chi} \rangle \cdot \nabla_{\mathbf{R}}\Phi) + D\Delta_{\mathbf{R}}\Phi. \quad (\text{I.14})$$

The average  $\langle \nabla_{\mathbf{r}}\boldsymbol{\chi} \rangle$  vanishes, and we thus finally obtain the homogenized diffusion equation

$$\partial_T\Phi = \nabla_{\mathbf{R}} \cdot (\mathbb{D}_{\text{eff}} \cdot \nabla_{\mathbf{R}}\Phi) \quad (\text{I.15})$$

where the effective diffusivity tensor is

$$\mathbb{D}_{\text{eff}} := D\mathbb{I} - \langle \mathbf{u}\boldsymbol{\chi} \rangle. \quad (\text{I.16})$$

## II. AN EXAMPLE

Consider the streamfunction for the *cellular flow*

$$\psi(x, y) = \sqrt{2}(U\ell/2\pi) \sin(2\pi x/\ell) \sin(2\pi y/\ell), \quad (\text{II.1})$$

with velocity

$$\begin{aligned} u(x, y) &= \partial_y\psi = \sqrt{2}U \sin(2\pi x/\ell) \cos(2\pi y/\ell), \\ v(x, y) &= -\partial_x\psi = -\sqrt{2}U \cos(2\pi x/\ell) \sin(2\pi y/\ell). \end{aligned} \quad (\text{II.2})$$

To compute the effective diffusivity, we need to solve the cell problem (I.12). Consider the ratio

$$\frac{|\mathbf{u} \cdot \nabla \boldsymbol{\chi}|}{|D\Delta \boldsymbol{\chi}|} \sim \frac{U\ell}{D} =: \text{Pe}, \quad (\text{II.3})$$

where Pe is the *Péclet number*. If the Péclet number is small, we can neglect the advection term in the cell problem, and get the simplified equation  $D\Delta \boldsymbol{\chi} = \mathbf{u}$ , or

$$D\Delta \chi_x = \sqrt{2}U \sin(2\pi x/\ell) \cos(2\pi y/\ell), \quad D\Delta \chi_y = -\sqrt{2}U \cos(2\pi x/\ell) \sin(2\pi y/\ell), \quad (\text{II.4})$$

with solution

$$\boldsymbol{\chi} = -\frac{\ell^2}{9\pi^2 D} \mathbf{u}. \quad (\text{II.5})$$

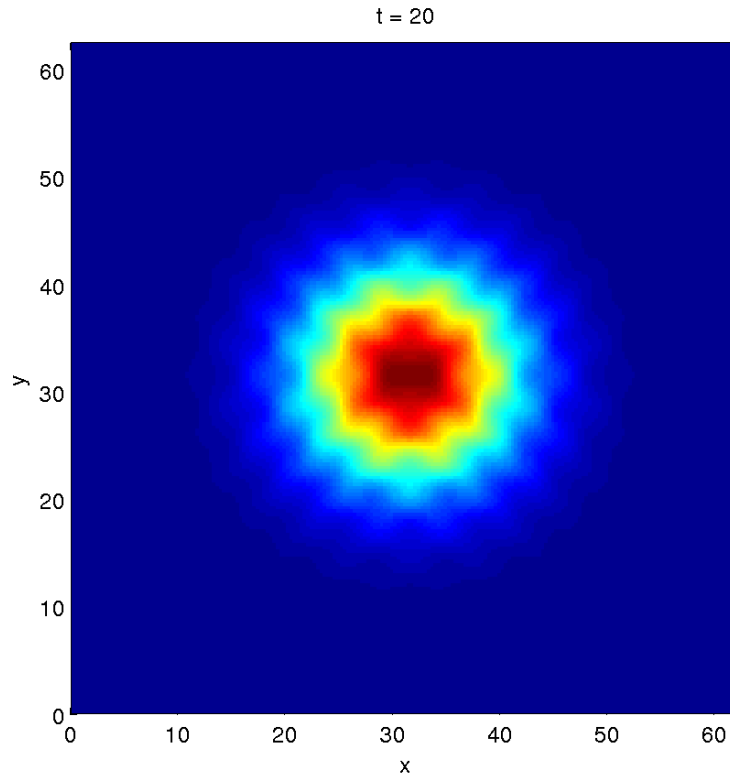


FIG. 1. Concentration field at  $t = 20$  for  $U = 1$ ,  $\ell = 2\pi$ ,  $D = 1$ .

We can then easily compute the effective diffusivity tensor by using  $\langle \mathbf{u}\mathbf{u} \rangle = \frac{1}{2}U^2\mathbb{I}$  in (I.16):

$$\mathbb{D}_{\text{eff}} := D \left( 1 + \frac{1}{16\pi^2} \text{Pe}^2 \right) \mathbb{I}. \quad (\text{II.6})$$

Figure 1 shows the concentration field for a numerical simulation at small Pe. In Figure 2 we compare the evolution of the variance to that implied by (II.6). Note that there is a short transient, since the initial condition has a small scale and so must spread out before scale separation is achieved.

- 
1. A. Crisanti, M. Falcioni, G. Paladin, and A. Vulpiani, ‘Anisotropic diffusion in fluids with steady periodic velocity fields,’ *J. Phys. A* **23**, 3307–3315 (1990).
  2. A. Famjiang and G. Papanicolaou, ‘Convection enhanced diffusion for periodic flows,’ *SIAM J. Appl. Math.* **54** (2), 333–408 (1994).
  3. E. Knobloch and W. J. Merryfield, ‘Enhancement of diffusive transport in oscillatory flows,’ *Astrophys. J.* **401**, 196–205 (1992).
  4. A. J. Majda and P. R. Kramer, ‘Simplified models for turbulent diffusion: Theory, numerical modelling and physical phenomena,’ *Physics Reports* **314** (4-5), 237–574 (1999).
  5. P. McCarty and W. Horsthemke, ‘Effective diffusion for steady two-dimensional flow,’ *Phys. Rev. A* **37** (6), 2112–2117 (1988).

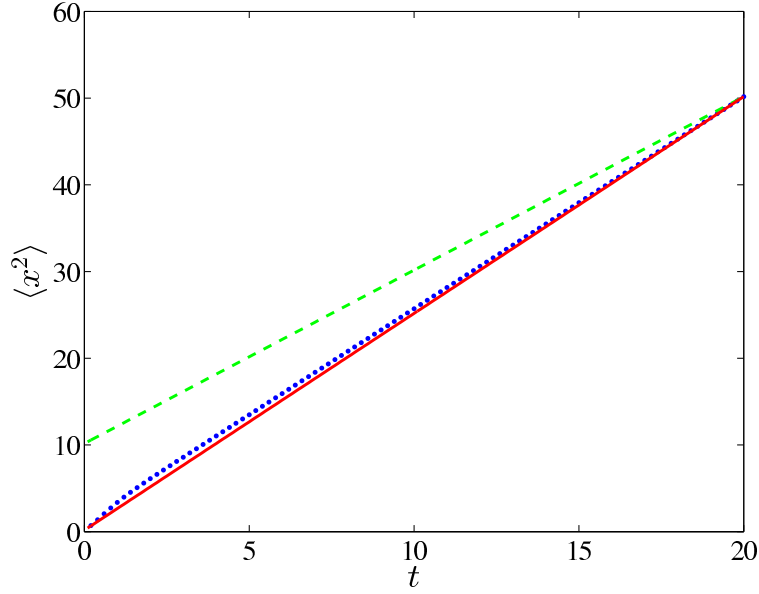


FIG. 2. Evolution of variance for  $U = 1$ ,  $\ell = 2\pi$ ,  $D = 1$ . The dots are numerical simulations, the green dashed line is  $2Dt$ , and the red line is  $2\mathbb{D}_{\text{eff}}t$ , where  $\mathbb{D}_{\text{eff}}$  is defined in (II.6).

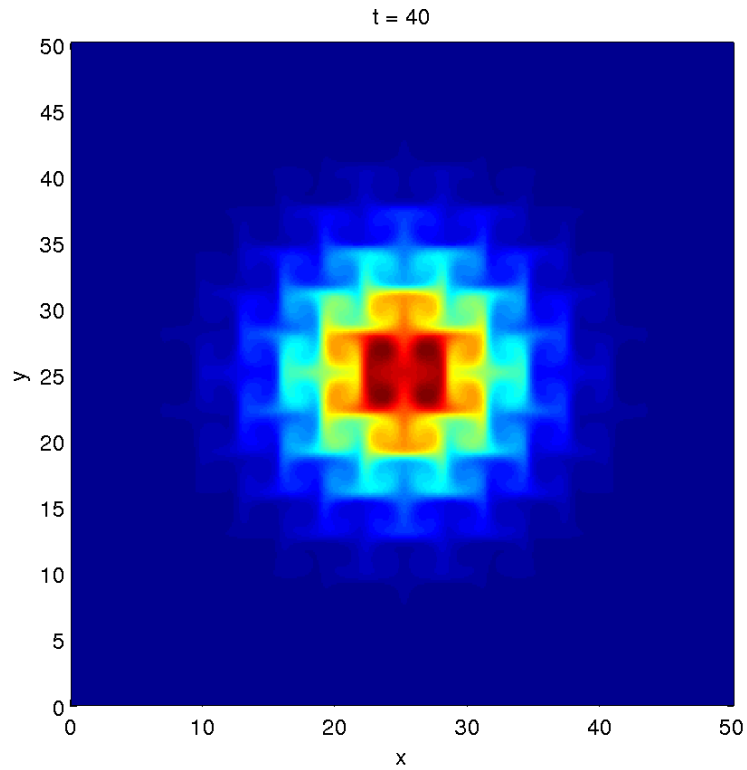


FIG. 3. Concentration field at  $t = 40$  for  $U = 1$ ,  $\ell = 2\pi$ ,  $D = 0.1$ .

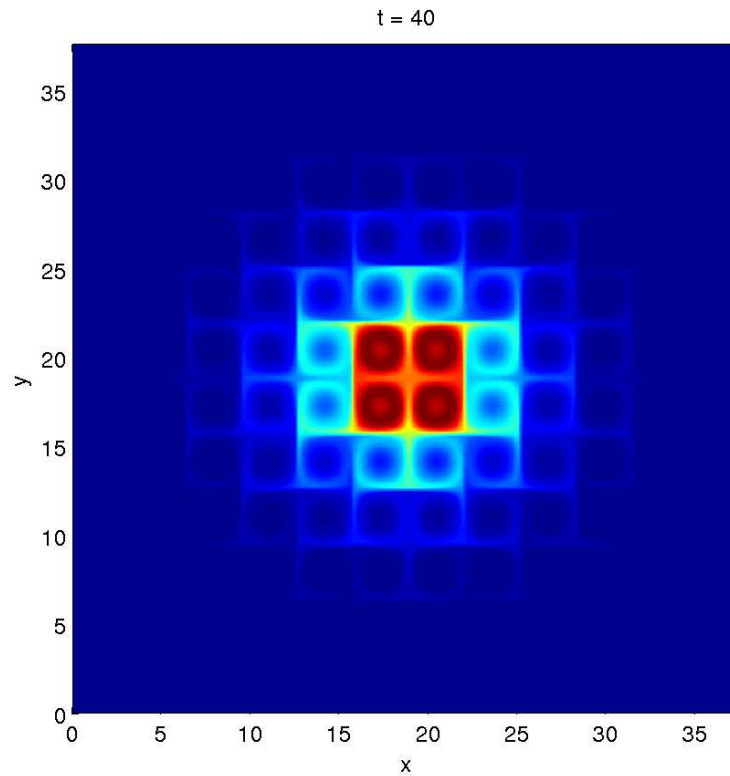


FIG. 4. Concentration field at  $t = 40$  for  $U = 1$ ,  $\ell = 2\pi$ ,  $D = 0.01$ .

6. M. N. Rosenbluth, H. L. Berk, I. Doxas, and W. Horton, 'Effective diffusion in laminar convective flows,' *Phys. Fluids* **30** (9), 2636–2647 (1987).
7. F. Sagues and W. Horsthemke, 'Diffusive transport in spatially periodic hydrodynamic flows,' *Phys. Rev. A* **34** (5), 4136–4143 (1986).
8. B. J. Shraiman, 'Diffusive transport in a Rayleigh–Bénard convection cell,' *Phys. Rev. A* **36** (1), 261–267 (1987).

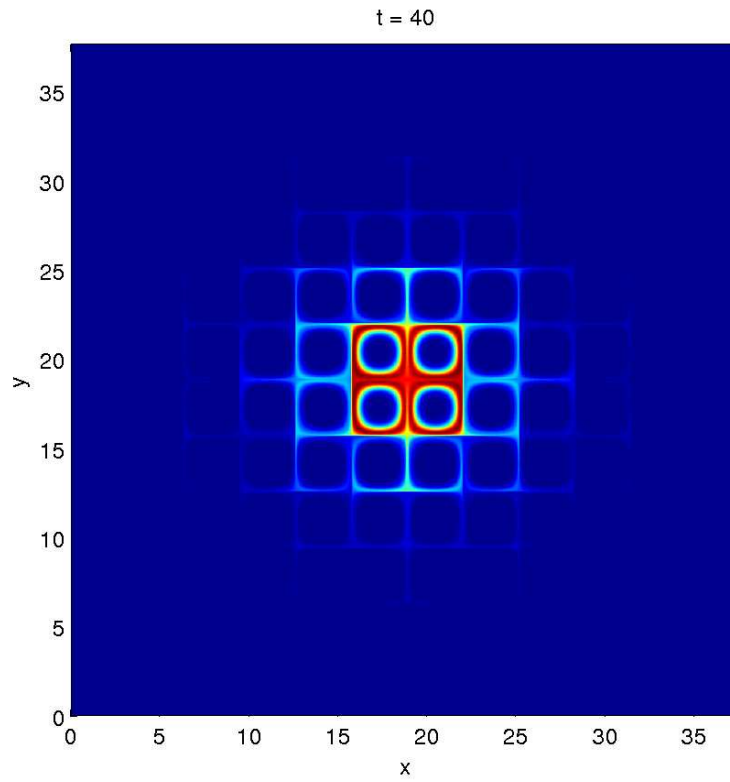


FIG. 5. Concentration field at  $t = 40$  for  $U = 1$ ,  $\ell = 2\pi$ ,  $D = 0.001$ .

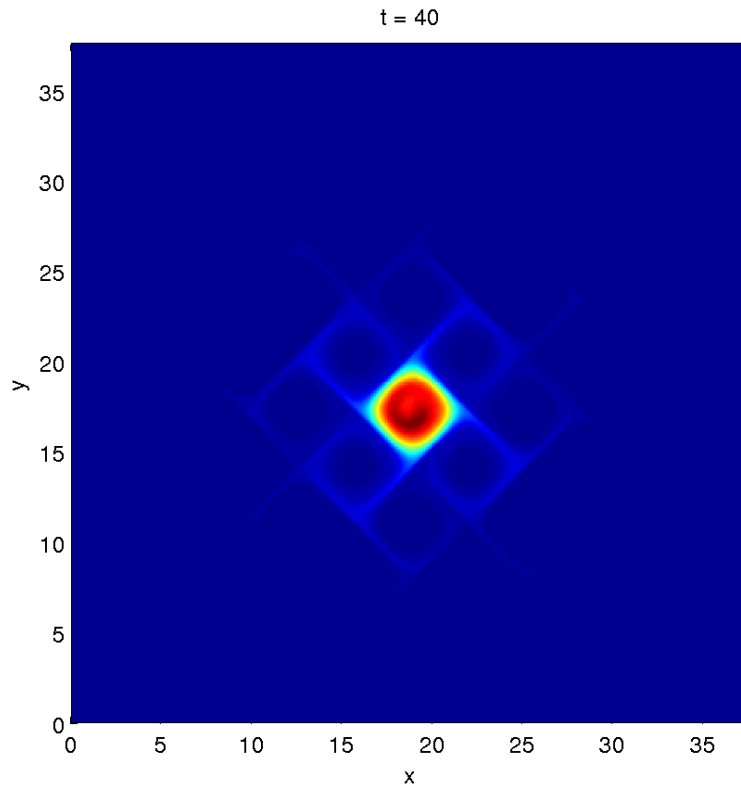


FIG. 6. Concentration field at  $t = 40$  for the flow  $\psi(x, y) = B \sin y + A \cos x$  with  $D = 0.01$ , and  $B = -A = 1$ . This flow has closed streamlines (see Crisanti *et al.*<sup>1</sup>).



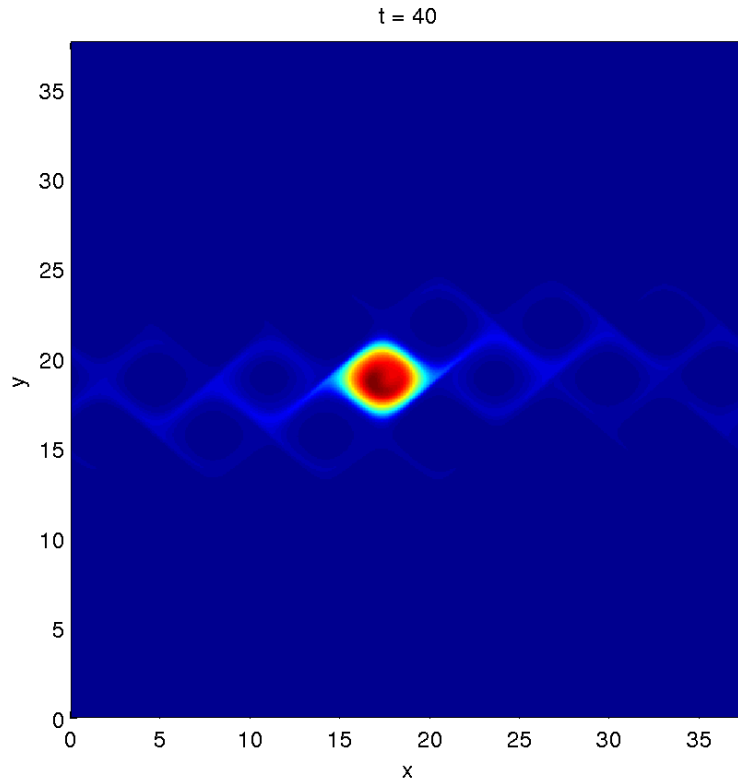


FIG. 7. Concentration field at  $t = 40$  for the flow  $\psi(x, y) = B \sin y + A \cos x$  with  $D = 0.01$ , and  $B = 1$ ,  $A = -1.3$ . This flow has open streamlines (see Crisanti *et al.*<sup>1</sup>).

## Lecture 16: Biomixing, part 1: hitting distribution

We use a simple model described by Thiffeault & Childress (2010) and refined by Lin *et al.* (2011), which is convenient for visualization and for taking limits. We assume there are  $N$  swimmers in a volume  $V$ , so the number density of swimmers is  $n = N/V$ . Initially, each swimmer travels at a speed  $U$  in a uniform random direction. They keep moving along a straight path for a time  $\tau$ , so that each traces out a segment of length  $\lambda = U\tau$ . After this a new direction is chosen randomly and uniformly, and the process repeats — each swimmer again moves along a straight path of length  $\lambda$ . Though far from realistic, this model captures many essential features of the system, as found by Thiffeault & Childress (2010); Lin *et al.* (2011) and as we'll explore further in this paper. We will discuss later how this model could be refined.

We wish to follow the displacement of an arbitrary ‘target fluid particle.’ The swimmers are all simultaneously affecting this fluid particle, but in practice only the closest swimmers significantly displace it. It is thus convenient to introduce an imaginary ‘interaction sphere’ of radius  $R$  centered on the target fluid particle, and count the number  $M_t$  of ‘interactions,’ that is the number of times a swimmer enters this sphere. (Our treatment applies to two-dimensional systems simply by changing ‘sphere’ to ‘disk’ and ‘volume’ to ‘area.’) Figure 1 illustrates the situation.

Each time a swimmer enters the interaction sphere, the target particle is displaced by some distance. We will address this in the next section and see how to sum the displacements due to many swimmers to obtain the distribution of the net displacement  $x$ . For now, let us find the distribution of  $M_t$ , the number of times a swimmer crosses the interaction sphere during a time  $t$ .

The probability that the swimmer starts inside a small volume  $dV$  is  $dV/V$ , where  $V$  is the total volume. The probability of a swimmer actually starting inside the interaction sphere is then  $V_{\text{sph}}(R)/V$ , where  $V_{\text{sph}}(R)$  is the volume of a sphere of radius  $R$ . (We assume the interaction sphere fits completely within the volume  $V$ .) We define the event

$$H_t = \text{a swimmer crosses the interaction sphere once during time } t < \tau (= \lambda/U), \quad (1)$$

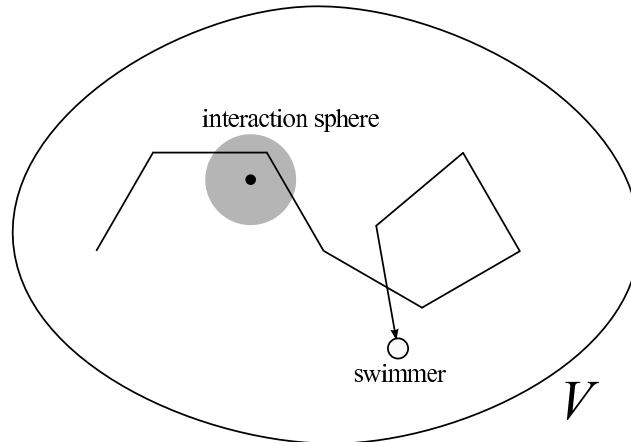


Figure 1: A swimmer moving inside a volume  $V$  along a series of straight paths, each of length  $\lambda$  and in a uniform random direction. The interaction sphere around the target particle (black dot) is shown in gray. Here the swimmer ‘interacts’ twice with the target particle, since two of its paths intersect the sphere.

that is, the center of the swimmer is inside the interaction sphere at some point while traveling on a straight path of length  $Ut < \lambda$ , where  $U$  is the uniform speed of a swimmer. To determine the probability of  $H_t$ , observe that because of the homogeneity and isotropy of the swimmers this probability is proportional to the volume swept out by the interaction sphere if it moves a distance  $Ut$ , with  $0 \leq t < \tau$ :

$$p_t := \mathbb{P}(H_t) = V_{\text{swept}}(R, \lambda)/V, \quad V_{\text{swept}}(R, \lambda) := V_{\text{cyl}}(R, \lambda) + V_{\text{sph}}(R), \quad (2)$$

where

$$V_{\text{cyl}}(R, \lambda) := \begin{cases} 2R\lambda, & (2\text{D}); \\ \pi R^2\lambda, & (3\text{D}); \end{cases} \quad V_{\text{sph}}(R) := \begin{cases} \pi R^2, & (2\text{D}); \\ \frac{4}{3}\pi R^3, & (3\text{D}); \end{cases} \quad (3)$$

are respectively the volume of the cylinder of radius  $R$  swept out in time  $t$  and the volume of the interaction sphere, which gives the probability that a swimmer starts inside the interaction sphere. This assumes that all points on the interaction sphere’s surface are at least a distance  $\lambda$  from the boundary of  $V$ .

For  $N$  swimmers, let  $M_t$  be the total number of interactions with the sphere during time  $t$ . In Appendix we use a generating function approach to find the probability distribution of  $M_t$ , and show that

$$\langle M_t \rangle = n \{ V_{\text{swept}}(R, \lambda) (t/\tau) + V_{\text{sph}}(R) \} \quad (4)$$

where  $n = N/V$  is the number density of swimmers. In this form we can take the limits  $N \rightarrow \infty$  and  $V \rightarrow \infty$  while keeping  $n$  constant, which doesn't change the expectation value.

Also from Appendix , the variance of  $M_t$  is

$$\text{Var } M_t = N \left( p_\tau(1 - p_\tau)(t/\tau) + \frac{1}{3}p_\tau^2 - \frac{1}{3}(V_{\text{sph}}(R)/V)(2p_\tau + 2(V_{\text{sph}}(R)/V) - 3) \right) \quad (5)$$

where  $V_{\text{sph}}(R)$  is the volume (or area) of the interaction sphere. Any term in (5) quadratic in  $V_{\text{sph}}(R)$  or  $p_\tau$  will vanish as  $V \rightarrow \infty$ , and we are left with

$$\text{Var } M_t \sim \langle M_t \rangle, \quad V \rightarrow \infty. \quad (6)$$

For large  $\langle M_t \rangle$  we thus expect that a typical value of  $M_t$  will be very close to the mean, since  $\langle M_t \rangle / \sqrt{\text{Var } M_t}$  is small. In that case, the central limit theorem applies ( $M_t$  is the sum of i.i.d. random variables) and we have the Gaussian approximation

$$\mathbb{P}\{M_t = m\} \simeq \frac{1}{\sqrt{2\pi \text{Var } M_t}} e^{-(m - \langle M_t \rangle)^2 / 2 \text{Var } M_t}, \quad \langle M_t \rangle \gg 1, \quad (7)$$

with  $\langle M_t \rangle$  defined in (4). The mean and variance equations (4) and (5) are exact as long as the interaction sphere is more than a path length  $\lambda$  away from the boundary of  $V$ ; equation (7) further requires  $\langle M_t \rangle \gg 1$ , which typically happens for long times. Figures 2(a)–2(b) show the convergence to a Gaussian distribution for numerical simulations of moving swimmers, in 2D and 3D.

## Appendix: Generating function approach for random phases

The generating function of a sequence  $\{a_n\}$  is defined as Feller (1968)

$$G(a_n; x) = \sum_{n=0}^{\infty} a_n x^n. \quad (8)$$

Now let  $a_n$  give the probability of having  $n$  events  $H_t$ . For a single swimmer moving for a time  $t < \tau$ , we can only have  $n = 0$  or  $1$  events, with probability  $a_0 = (1 - p_t)$  and  $a_1 = p_t$ ; hence,

$$G_t(x) = a_0 + a_1 x = (1 - p_t) + p_t x, \quad t < \tau. \quad (9)$$

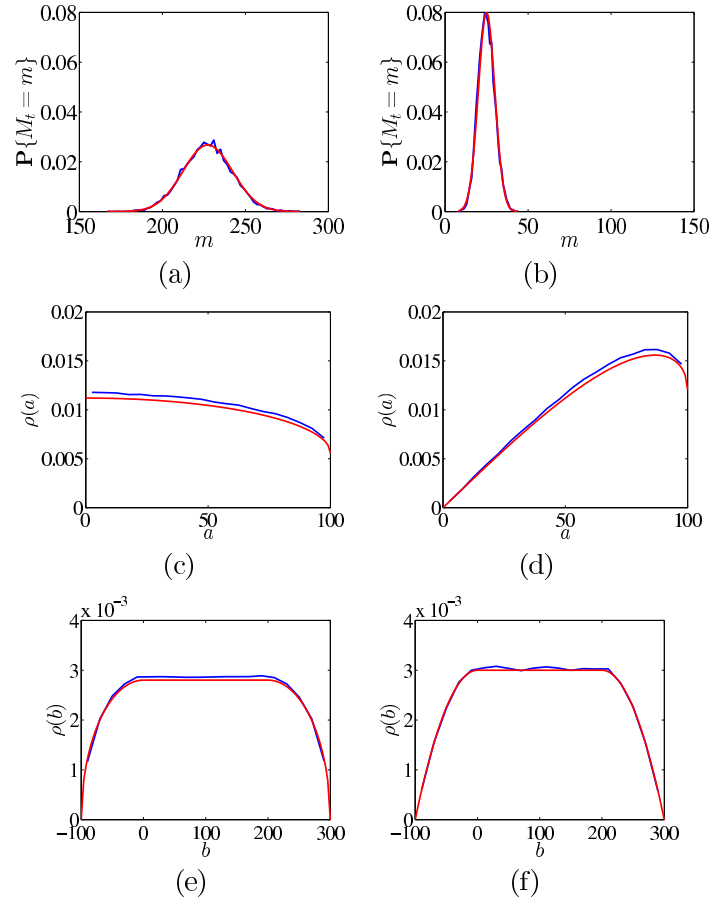


Figure 2: Probability distribution function for  $N = 1000$  swimmers to enter the interaction sphere  $M_t = m$  times, in (a) 2D and (b) 3D. The interaction sphere has radius  $R = 100$ , the path length  $\lambda = 200$ , the total volume is a sphere of radius  $L = 1000$ , and the number of steps is  $k = \lfloor Ut/\lambda \rfloor = 10$ . Shown in red is the Gaussian approximation (7). (c)–(d) Marginal probability densities  $\rho(a)$  in 2D and 3D, respectively. (e)–(f) Marginal probability densities  $\rho(b)$  in 2D and 3D, respectively.

The expected number of events is  $\langle M_t \rangle = G'_t(1) = p_t$ . If the swimmer moves for a time  $t = k\tau$ ,  $k \in \mathbb{Z}^+$ , the total number of events is the sum of the events at each interval  $\tau$ . The resulting generating function is then  $G_\tau^k(x)$ , assuming that the swimmer starts on its first path at  $t = 0$ . More generally, if the swimmer has already started on a path before  $t = 0$ , then

$$G_t(x) = G_{\tau_0}(x) G_\tau^{k_{t,\tau_0}}(x) G_{\tau_1}(x) \quad (10)$$

where  $\tau_0 + \tau k_{t,\tau_0} + \tau_1 = t$ ,  $k_{t,\tau_0} = \lfloor (t - \tau_0)/\tau \rfloor$ , and  $0 \leq \tau_i < \tau$ . The two  $\tau_i$  pieces account for the partial paths traversed at the beginning and at the end of the motion. We take  $\tau_0 \in [0, \tau)$  to be a uniformly-distributed random variable;  $\tau_1$  then follows from  $\tau_1 = t - \tau_0 - \tau k_{t,\tau_0}$ .

Now write  $p_t = \alpha t + \beta$ , where the constants  $\alpha$  and  $\beta$  come from (2). The expected number of events  $H_t$  is

$$\begin{aligned} \langle M_t \rangle &= \langle p_{\tau_0} + k_{t,\tau_0} p_\tau + p_{\tau_1} \rangle = \langle \alpha(\tau_0 + \tau k_{t,\tau_0} + \tau_1) + (k_{t,\tau_0} + 2)\beta \rangle \\ &= \alpha t + \beta(2 + \langle k_{t,\tau_0} \rangle). \end{aligned}$$

To compute  $\langle k_{t,\tau_0} \rangle$ , let  $t/\tau = \ell + \delta$ ,  $\ell = \lfloor t/\tau \rfloor$ ,  $\delta \in [0, 1)$ . Then  $\langle k_{t,\tau_0} \rangle = \langle \lfloor (t - \tau_0)/\tau \rfloor \rangle = \ell + \langle \lfloor \delta - \tau_0/\tau \rfloor \rangle$ , with  $|\delta - \tau_0/\tau| < 1$ , and

$$\langle \lfloor (\delta - \tau_0)/\tau \rfloor \rangle = \frac{1}{\tau} \int_0^\tau \lfloor \delta - \tau_0/\tau \rfloor d\tau_0 = \int_0^1 \lfloor \delta - \xi \rfloor d\xi = \int_\delta^1 (-1) d\xi = \delta - 1.$$

Thus,

$$\langle k_{t,\tau_0} \rangle = \ell + \delta - 1 = t/\tau - 1, \quad (11)$$

and we finally conclude

$$\langle M_t \rangle = (\alpha\tau + \beta)t/\tau + \beta = p_\tau (t/\tau) + \beta. \quad (12)$$

The extra  $\beta$  at the end arises from the possibility of swimmers starting inside the interaction sphere at  $t = 0$ .

We can also compute the variance exactly. For a single swimmer,

$$\text{Var } M_t = G''_t(1) + G'_t(1) - [G'_t(1)]^2 = p_t - (p_t)^2 = p_t(1 - p_t), \quad t < \tau, \quad (13)$$

and for longer time

$$\begin{aligned} \text{Var } M_t &= \langle p_{\tau_0}(1 - p_{\tau_0}) + k_{t,\tau_0} p_\tau(1 - p_\tau) + p_{\tau_1}(1 - p_{\tau_1}) \rangle \\ &= \langle M_t \rangle - \langle p_{\tau_0}^2 + k_{t,\tau_0} p_\tau^2 + p_{\tau_1}^2 \rangle \leq \langle M_t \rangle. \end{aligned}$$

Now we need to compute the expectation value of this over  $\tau_0$ . This is a slightly tedious calculation which we do not present; the final result is

$$\text{Var } M_t = p_\tau(1 - p_\tau) (t/\tau) + \frac{1}{3}\alpha^2\tau^2 + \beta(1 - \beta). \quad (14)$$

For  $N$  swimmers, because we are still summing the number of displacements the generating function will be the product of several copies of (10):

$$G_t^N(x) = \prod_{j=1}^N G_{\tau_{0,j}}(x) G_\tau^{k_{t,\tau_{0,j}}}(x) G_{t-\tau_{0,j}-\tau k_{t,\tau_{0,j}}}(x) \quad (15)$$

where each swimmer has its own random initial partial path  $\tau_{0,j}$ . The probability distribution will thus be a convolution of all these generating functions, and the expected value and variance will add up. The net result is to multiply the expected number of events (12) and its variance (14) by  $N$ . After substituting the value of  $\alpha$  and  $\beta$  from  $p_t = \alpha t + \beta$  and (2) and using  $n = N/V$ , we obtain equations (4) and (5).

## References

- Feller, W. (1968). *An Introduction to Probability Theory and Its Applications*, volume 1. New York: John Wiley & Sons, third edition.
- Lin, Z., Thiffeault, J.-L., & Childress, S. (2011). *J. Fluid Mech.* **669**, 167–177. <http://arxiv.org/abs/1007.1740>.
- Thiffeault, J.-L. & Childress, S. (2010). *Phys. Lett. A*, **374**, 3487–3490.

## Lecture 17: Biomixing, part 2: Effective diffusivity

In the previous lecture we derived an expression for the distribution of number of interactions  $m$  with a sphere of radius  $R$ :

$$\mathbb{P}\{M_t = m\} \simeq \frac{1}{\sqrt{2\pi \text{Var } M_t}} e^{-(m - \langle M_t \rangle)^2 / 2 \text{Var } M_t}, \quad \langle M_t \rangle \gg 1, \quad (1)$$

where the expected number of interactions is

$$\langle M_t \rangle = n \{V_{\text{swept}}(R, \lambda) (t/\tau) + V_{\text{sph}}(R)\}, \quad (2)$$

with  $n$  the number density of swimmers,  $t$  the time elapsed,  $\tau$  the duration of a path,  $\lambda$  the length of a path, and  $V_{\text{swept}}(R, \lambda)$  and  $V_{\text{sph}}(R)$  the volume of a cylinder and sphere.

Now that we've examined how often swimmers interact with a sphere of radius  $R$  centered around a target particle, we will look at how the particle gets displaced. Figure 1 shows the setup of an interaction. Since the system is homogeneous and isotropic, only two 'impact parameters'  $a$  and  $b$  are needed to describe an interaction. These are depicted in the figure: here  $C$  is the point along the line of motion that is closest to the initial position of the particle, and  $a \in [0, R]$  is this closest distance. The parameter  $b \in [-R, \lambda + R]$  is the distance from  $C$  to the initial position of the swimmer. A negative value of  $b$  means the swimmer started its path beyond the point  $C$ .

Following Lin *et al.* (2011), we start from a distribution of displacements  $\Delta_\lambda(a, b)$  induced by a single swimmer. Here the impact parameters  $a$  and  $b$  describe the encounter between the swimmer and a target particle, and  $\lambda$  is the path length of swimming (Fig. 1). Each time a swimmer enters the interaction sphere we have an 'encounter,' which causes a displacement of the target particle; thus, after  $m$  encounters, the  $x$  displacement is

$$X_m = \sum_{k=1}^m \Delta_\lambda(a_k, b_k) \cos \psi_k \quad (3)$$



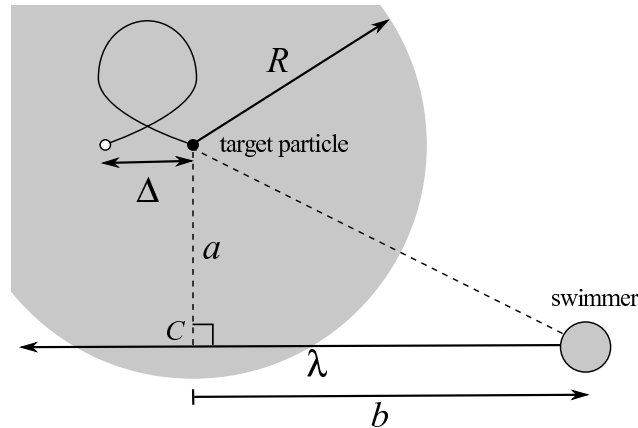


Figure 1: Definition of impact parameters  $a$  and  $b$ , displacement  $\Delta = \Delta_\lambda(a, b)$ , and swimming path length  $\lambda$ . In this picture the parameter  $b$  is positive; negative  $b$  corresponds to the swimmer starting its trajectory past the point  $C$  of smallest initial perpendicular distance to the line of motion. The filled dot is the initial position of the target particle and the hollow dot is its final position after the swimmer has moved by a distance  $\lambda$ . The ‘interaction sphere’ of radius  $R$  is also shown. (After Lin *et al.* (2011).)

where each encounter has random i.i.d. values of the impact parameters  $a_k$  and  $b_k$  and angle  $\psi_k$ . We select the  $X$  displacement here, but by isotropy the statistics in any direction are the same.

The probability density of  $X_m$  can be related to that of  $X_t$ , the  $x$  displacement after a time  $t$ , by first observing that  $\mathbb{P}\{X_m \in [x, x + dx]\} = \mathbb{P}\{X_t \in [x, x + dx] \mid M_t = m\}$ , and

$$\begin{aligned} \mathbb{P}\{X_t \in [x, x + dx]\} &= \sum_{m=0}^{\infty} \mathbb{P}\{X_t \in [x, x + dx], M_t = m\} \\ &= \sum_{m=0}^{\infty} \mathbb{P}\{X_t \in [x, x + dx] \mid M_t = m\} \mathbb{P}\{M_t = m\}, \end{aligned} \quad (4)$$

where  $\mathbb{P}\{M_t = m\}$  is the probability of getting  $m$  encounters in time  $t$ . If the latter is sharply peaked, such as in the Gaussian limit (1), then we can just use  $m \simeq \langle M_t \rangle$ . But for now let us focus on  $\mathbb{P}\{X_m \in [x, x + dx]\}$ .

We wish to derive the PDF of the total  $x$  displacement  $X_m$ , assuming that the random variables  $a_k$ ,  $b_k$ ,  $\psi_k$  are independent for different  $k$  and identically distributed, with probability densities  $\rho_{ab}(a_k, b_k)$  and  $\rho_\psi(\psi_k)$ . Because of isotropy, the angular

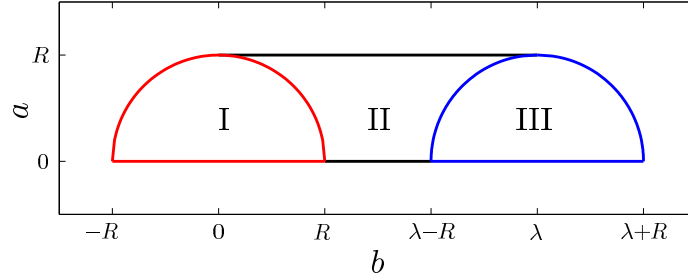


Figure 2: The domain  $\Omega_{ab} = \text{I} \cup \text{II} \cup \text{III}$  of the impact parameters  $a$  and  $b$  for fixed path length  $\lambda$  (see Fig. 1). Region I corresponds to swimmers that start their path inside the interaction sphere; swimmers in Region II cross the sphere completely; swimmers in Region III finish their path inside the sphere. Note that the figure depicts  $\lambda > 2R$ , but all the formulas hold for  $\lambda < 2R$  as well, when regions I and III overlap because some trajectories both start and finish inside the sphere.

variables have simple densities:

$$\rho_\psi(\psi) = 1/2\pi, \quad \Omega_\psi = [0, 2\pi] \quad (2\text{D}); \quad \rho_\psi(\psi) = \frac{1}{2} \sin \psi, \quad \Omega_\psi = [0, \pi] \quad (3\text{D}), \quad (5)$$

for  $\psi \in \Omega_\psi$ . In two dimensions, the joint density  $\rho_{ab}(a, b)$  is uniform over the domain  $\Omega_{ab} = \{0 \leq a \leq R, -\sqrt{R^2 - a^2} \leq b \leq \lambda + \sqrt{R^2 - a^2}\}$  depicted in Fig. 2. These are the values of  $a$  and  $b$  for which a swimmer's straight path intersects the interaction sphere. After normalizing, we find the density

$$\rho_{ab}(a, b) = 2/V_{\text{swept}}(R, \lambda) \quad (2\text{D}). \quad (6)$$

In three dimensions, the domain in Fig. 2 is interpreted as a surface of revolution about  $a = 0$ , leading to the density

$$\rho_{ab}(a, b) = 2\pi a/V_{\text{swept}}(R, \lambda) \quad (3\text{D}). \quad (7)$$

For both the 2D and 3D cases,  $\rho_{ab}(a, b)$  is then normalized such that

$$\int_{\Omega_{ab}} \rho_{ab}(a, b) da db = \int_0^R \int_{-\sqrt{R^2 - a^2}}^{\lambda + \sqrt{R^2 - a^2}} \rho_{ab}(a, b) db da = 1. \quad (8)$$

We have the convenient forms

$$\langle M_t \rangle \rho_{ab} \simeq 2nt/\tau \quad (2\text{D}); \quad \langle M_t \rangle \rho_{ab} \simeq 2\pi ant/\tau \quad (3\text{D}), \quad (9)$$

in terms of the expected values (2). These are valid for  $t \gg \tau$ , so we can neglect the extra added spherical volume in (2).

We can now compute the effective diffusivity. We have of course  $\langle X_M \rangle = 0$  because of isotropy. The variance is then

$$\langle X_m^2 \rangle = \sum_{k=1}^m \langle \Delta_\lambda^2(a_k, b_k) \cos^2 \psi_k \rangle = m \langle \Delta_\lambda^2(a, b) \rangle \langle \cos^2 \psi \rangle \quad (10)$$

since the variables are i.i.d. The angular average is

$$\langle \cos^2 \psi \rangle = \frac{1}{2\pi} \int_0^{2\pi} \cos^2 \psi \, d\psi = \frac{1}{2} \quad (2D); \quad (11)$$

$$\langle \cos^2 \psi \rangle = \frac{1}{2} \int_0^\pi \cos^2 \psi \sin \psi \, d\psi = \frac{1}{3} \quad (3D). \quad (12)$$

So now we define the *effective diffusivity*  $D$

$$\langle X_m^2 \rangle = \frac{m}{d} \langle \Delta_\lambda^2(a, b) \rangle = 2Dt \quad (13)$$

where  $d$  is the dimension of space. We have finally

$$D = \frac{m}{2dt} \langle \Delta_\lambda^2(a, b) \rangle, \quad (14)$$

where

$$\langle \Delta_\lambda^2(a, b) \rangle = \int_{\Omega_{ab}} \rho_{ab}(a, b) \Delta_\lambda^2(a, b) \, da \, db. \quad (15)$$

Assume now that  $m = \langle M_t \rangle$ , which will be satisfied if there are many encounters. Then using (9) we find

$$D = \frac{n}{2\tau} \int_{\Omega_{ab}} \Delta_\lambda^2(a, b) \, da \, db, \quad (2D); \quad (16)$$

and

$$D = \frac{\pi n}{3\tau} \int_{\Omega_{ab}} \Delta_\lambda^2(a, b) a \, da \, db, \quad (3D). \quad (17)$$

where recall that  $\tau = \lambda/U$  is the path length of swimming. Notice the extra  $a$  in the 3D integrand, due to the fact that there is a ‘ring’ of points a distance  $a$  from the target. This extra  $a$  will modify the dependence in 2D and 3D quite dramatically.

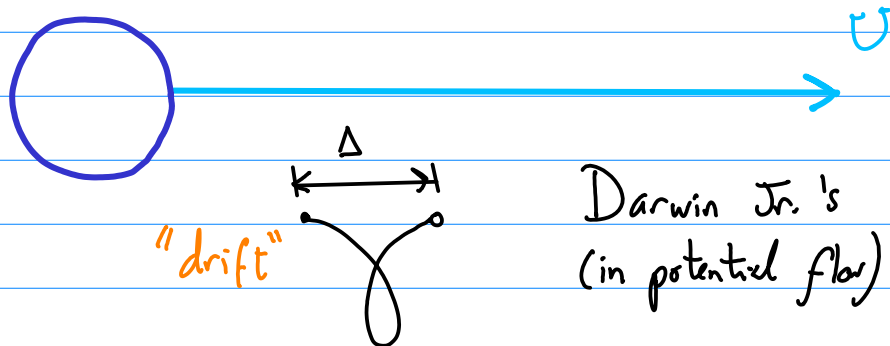
So far everything is quite general, as long as the density of swimmers is low enough. In the next lecture we will discuss the most crucial part: how to model  $\Delta_\lambda(a, b)$ . This depends heavily on the kind of swimmer and the type of fluid.

## References

- Lin, Z., Thiffeault, J.-L., & Childress, S. (2011). *J. Fluid Mech.* **669**, 167–177.  
<http://arxiv.org/abs/1007.1740>.
- Thiffeault, J.-L. & Childress, S. (2010). *Phys. Lett. A*, **374**, 3487–3490.

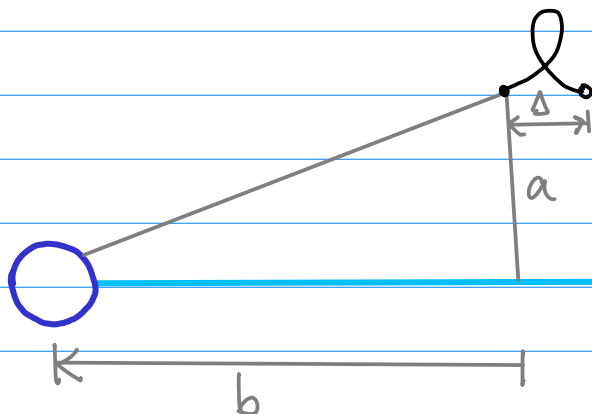
# Lecture 18: Displacements in inviscid flow

Single swimmer: take a cylinder



Darwin Jr.'s "elastica"  
(in potential flow)

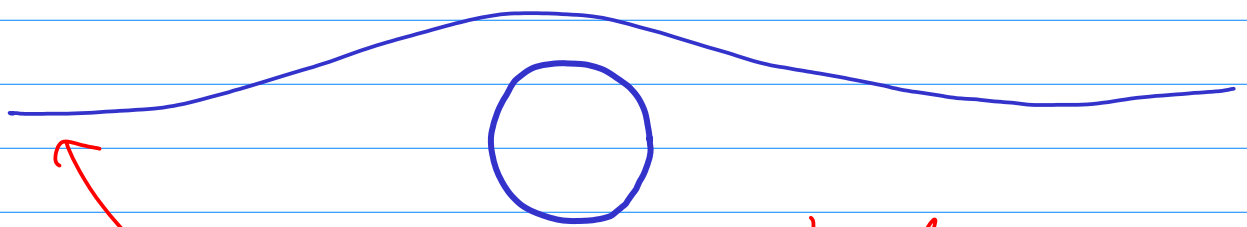
How do we compute  $\Delta$ ?



- Swimming velocity is  $U$  (const.)
- Straight line for distance  $\lambda$ .
- Axially-symmetric, steady swimmer
- $a, b$  are "impact parameters" ( $a > 0$ )

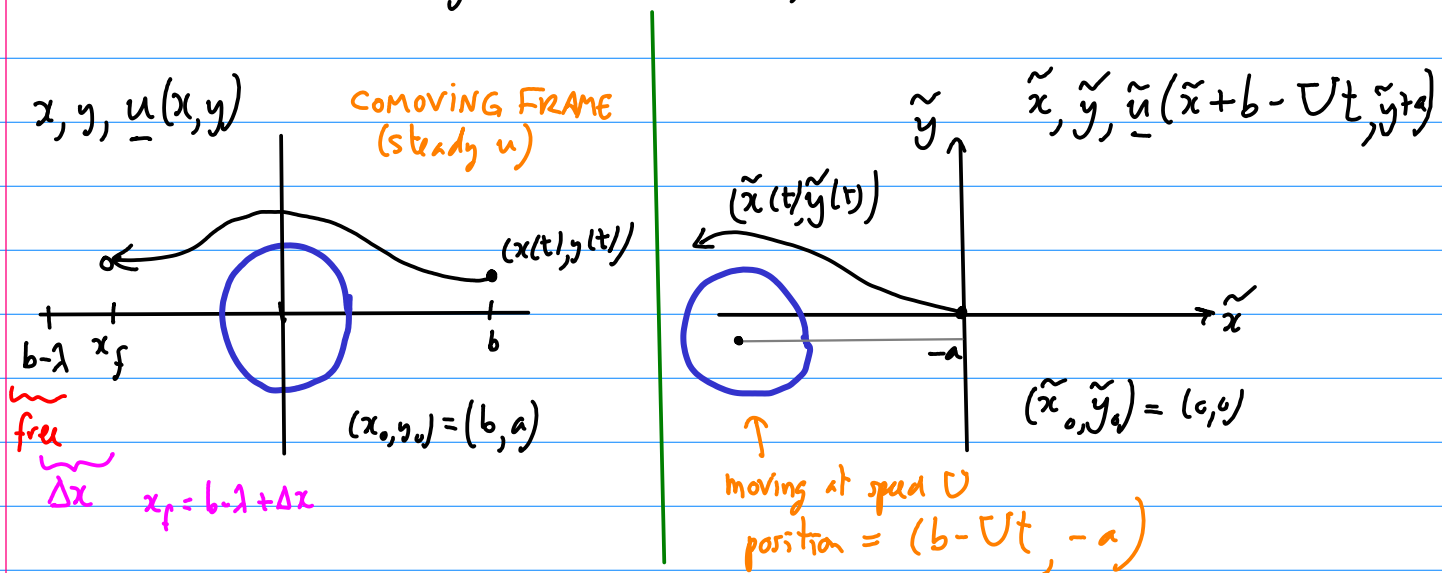
Compute  $\Delta_\lambda(a, b)$

Far field: displacements tiny. Orbits almost closed,



a particle stays on the same streamline in comoving frame  
 $\Delta y = 0$

Do 2D case (axisymmetric 3D similar):



$$\frac{d\tilde{x}}{dt} = \tilde{u}(\tilde{x}(t) + b - Ut, \tilde{y}(t) + a), \quad x = \tilde{x} + b - Ut$$

$$\frac{dx}{dt} + U = \tilde{u}(x, y) \iff \frac{dx}{dt} = -U + \tilde{u}(x, y) = u(x, y)$$

$$-\lambda + \Delta x = \int_0^{T=\lambda/U} u(x(t), y(t)) dt$$

need both

For y:  $\frac{dy}{dt} = \tilde{v}(x, y)$

Alternate form:  $T = \frac{\lambda}{U} = \int_b^{x_f} \frac{dx}{u(x, y)}$ ,  $x_f = b - \lambda + \Delta x$

$$\frac{\lambda}{U} = \int_b^{b-\lambda+\Delta x} \frac{dx}{u} = - \int_b^{b-\lambda+\Delta x} \frac{dx}{|u|} = \int_{b-\lambda+\Delta x}^b \frac{dx}{|u|}$$

$$= \int_{b-\lambda}^b \frac{dx}{|u|} + \int_{b-\lambda+\Delta x}^{b-\lambda} \frac{dx}{|u|}$$

↑  
how far particle moves  
w/o "free-streaming"

← If particle doesn't move  
much and  $|b-\lambda|$  "large",  
then  $|u| \approx U$

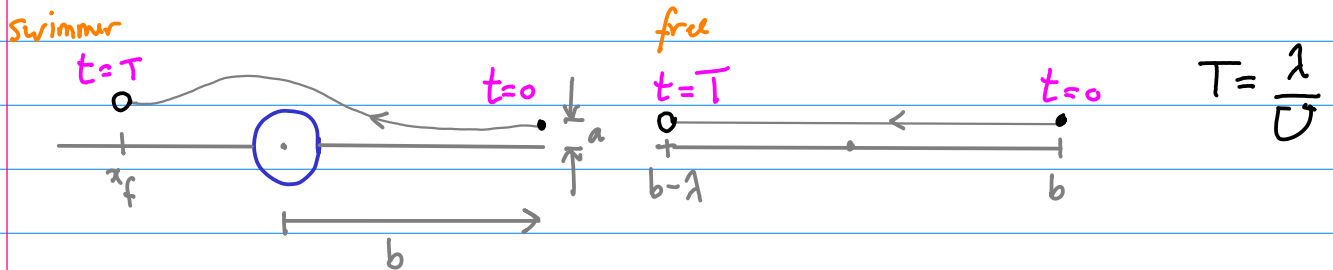
$$\frac{\lambda}{U} \approx \int_{b-\lambda}^b \frac{dx}{|u|} - \frac{\Delta x}{U} \Leftrightarrow \Delta x = \int_{b-\lambda}^b \frac{dx}{|u|} - \frac{\lambda}{U}$$

$$\Delta x \approx \int_{b-\lambda}^b \left( \frac{1}{|u|} - \frac{1}{U} \right) dx$$

Better form, since now can take  $b \rightarrow \infty$ ,  $b-\lambda \rightarrow -\infty$  if we want.

"Rayleigh form"

Intuitively, this formula measures the "lag" behind a free-streaming particle:



2D incompressible:  $u = \frac{\partial \psi}{\partial y}$ ,  $v = -\frac{\partial \psi}{\partial x}$

$$\psi(x_f, a + \Delta y) = \psi(b, a) \quad \text{Same streamline}$$

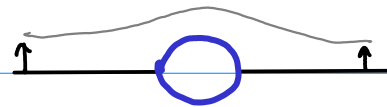
$$\psi(b - \lambda + \Delta x, a + \Delta y) = \psi(b, a) \quad \text{solve for } \Delta y, \text{ given } \Delta x$$

If  $|b - \lambda| \gg \Delta x$ ,  $\psi(b - \lambda, a + \Delta y) \approx \psi(b, a)$  solve for  $\Delta y$

If also  $\Delta y \ll a$ ,  $\psi(b - \lambda, a) + \Delta y \underbrace{\partial_y \psi(b - \lambda, a)}_{u(b - \lambda, a)} \approx \psi(b, a)$

$$\Delta y \approx \frac{\psi(b, a) - \psi(b - \lambda, a)}{u(b - \lambda, a)}$$

Now for infinite  $\lambda$ , we have:



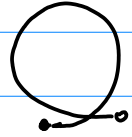
$$\Delta y = \frac{\psi(\infty, a) - \psi(-\infty, a)}{U} = 0!$$

$\Delta y = 0$  for  $\lambda \rightarrow \infty, b-1 \rightarrow -\infty$

Cylinder in potential flow:

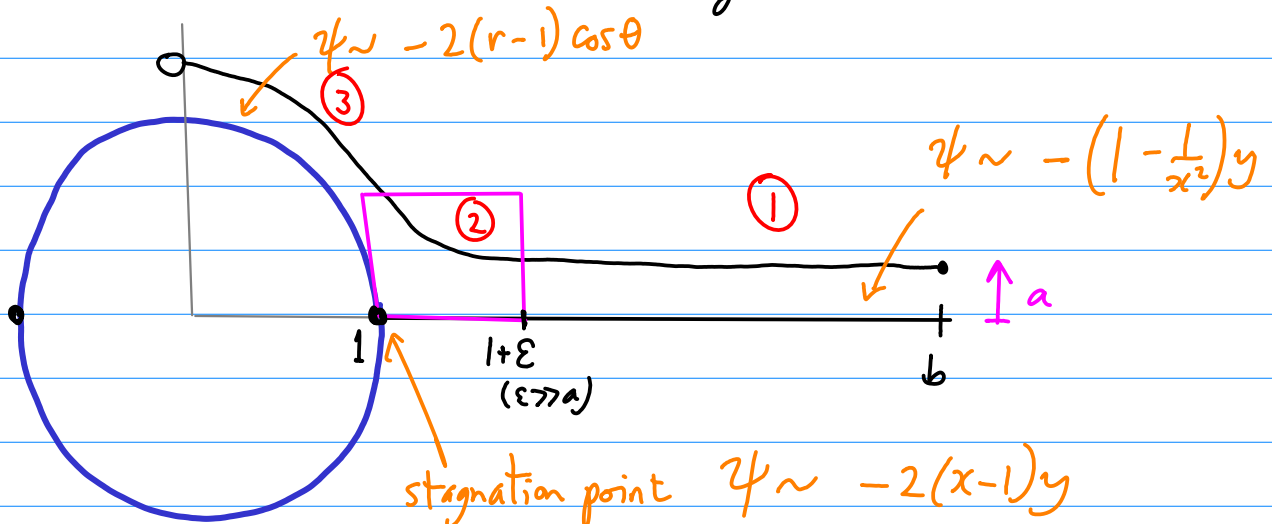
$$\psi(x, y) = -Uy \left( 1 - \frac{l^2}{x^2 + y^2} \right) \quad \text{Set } U = l = 1$$

Far away,  $\tilde{\psi} \sim \frac{y}{r^2}$ , so  $\tilde{u} \sim \frac{1}{r^2}$  in fixed frame

However, trajectories are almost closed,   $\frac{1}{a}$

Net result is  $\Delta(a) \sim \frac{1}{a^3}$  Much smaller  $\frac{1}{a^3}$  than overall excursion!

The limit  $a \ll 1$  is more interesting:





Need to calculate  $\int \left(\frac{1}{u} + 1\right) dx$  over each region ①, ②, ③.

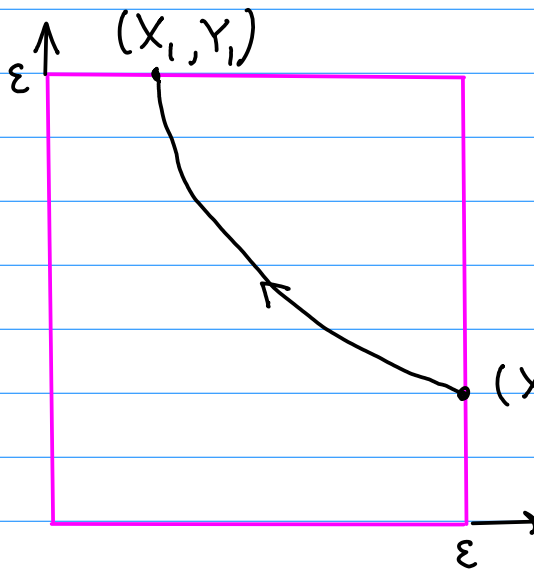
Region 1:  $\psi_0 = \psi(b, a) = -(1 - b^{-2})a$

$$u = -(1 - x^{-2}), \quad T_1 = \int_b^{1+\varepsilon} \left(\frac{1}{u} + 1\right) dx = \int_b^{1+\varepsilon} \frac{dx}{1 - x^2}$$

transit time

After using  $\varepsilon \ll 1, b \gg 1$ :  $T_1 \approx \frac{1}{2} \log(2/\varepsilon) + \varepsilon/4 - b^{-1} + O(\varepsilon^2, b^{-2})$

Region 2:



$$X = x - 1, \quad Y = y$$

$$\psi = -2XY$$

At  $X_0, Y_0$ ,  $\psi = -2X_0Y_0 = -(1 - b^{-2})a$  *neglect*

$$\Rightarrow Y_0 = \frac{a}{2\varepsilon}$$

$(X_0, Y_0) = (\varepsilon, a/2\varepsilon)$

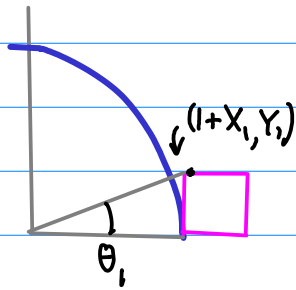
But also  $Y_1 = \varepsilon$ , so  $X_1 = a/2\varepsilon$ .

$$T_2 = \int_{X_0}^{X_1} \left(\frac{1}{u} + 1\right) dx = \int_{\varepsilon}^{a/2\varepsilon} \left(\frac{1}{(-2X)} + 1\right) dx = -\frac{1}{2} \log\left(\frac{a/2\varepsilon}{\varepsilon}\right) + \frac{a}{2\varepsilon} - \varepsilon$$

*small*

$$u = -1 + \frac{\cos 2\theta}{\text{Mixing } \frac{1}{2}}$$

Region 3:



$$T_3 = \int_{\theta_1}^{\pi/2} \left( \frac{1}{u} + 1 \right) \frac{dx}{d\theta} d\theta$$

$$= \frac{1}{2} \int_{\theta_1}^{\pi/2} \frac{\cos 2\theta}{\sin \theta} d\theta, \quad \theta_1 \text{ small.}$$

*u = -r \sin \theta*

$$T_3 \approx -1 + \frac{1}{2} \log 2 - \frac{1}{2} \log \theta_1 + O(\theta_1^2)$$

$$\tan \theta_1 = \frac{y_1}{1+x_1} = \frac{\epsilon}{1+a/\epsilon} \approx \epsilon (1 - a/\epsilon) = \epsilon - a$$

$$\therefore T_3 \approx -1 + \frac{1}{2} \log 2 - \frac{1}{2} \log \epsilon + \frac{1}{2} \frac{a}{\epsilon} + O((a/\epsilon)^2)$$

Add everything together:

$$T = T_1 + T_2 + T_3 = \left( \frac{1}{2} \log(2/\epsilon) - b^{-1} \right) + \left( -\frac{1}{2} \log(a/2\epsilon) \right)$$

*divergent log ε terms cancel*

$$T = -\frac{1}{2} \log a - 1 + \frac{3}{2} \log 2 - b^{-1} + \left( -1 + \frac{1}{2} \log 2 - \frac{1}{2} \log \epsilon \right)$$

*to leading order.*

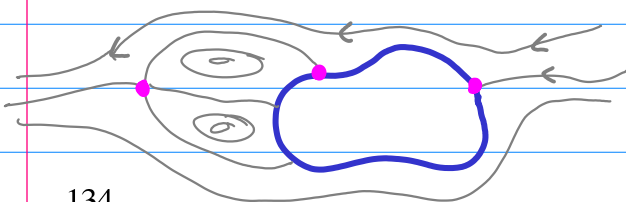
*Dominant term for small a.*

*Comes only from region 2, near stagnation point.*

$$T \rightarrow \infty \text{ for } a \rightarrow 0.$$

*particle gets stuck!*

The total drift is given by  $2T$ , since the body is fore-aft symmetric.



In general, the coefficient of  $\log a$  is given by summing over the linearization coeffs for each (hyperbolic) stagnation pt. encountered. *(not true for no-slip!)*

Note that to pick up the  $-\log a$  contribution, the target particle must come in the vicinity of the stagnation points

$$\Delta_\lambda(a, b) = \begin{cases} -\log a & , 0 \leq b \leq 1 \\ \text{(neglect)} & , \text{otherwise} \end{cases}$$

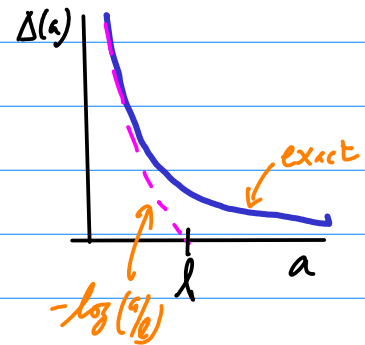
Cylinder.  $\kappa \simeq \frac{2 U n \lambda}{\lambda} \int_0^l \log^2\left(\frac{a}{x}\right) da$

$$\int \log^2 x dx = x \log^3 x - 2x \log x + 2x$$

$$\int_0^1 \log^2 x dx = 2 \quad (\text{numerical answer: } 2.37)$$

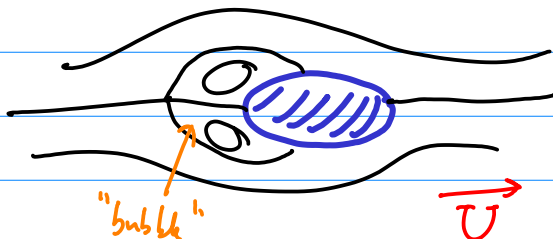
$$\kappa \simeq U n l^3$$

$$(\text{numerical: } \kappa = 1.19 U n l^3)$$



Note that this is completely independent of  $\lambda$ !

Another example: consider a swimmer with a bubble "wake":



If a particle is trapped in the bubble, moves by  $\lambda$ .

$$\Delta_\lambda(a, b) = \begin{cases} \lambda, & \text{particle inside bubble} \\ \text{neglect}, & \text{otherwise} \end{cases}$$

$$6\kappa = \frac{2U_n}{\lambda} \int \lambda^2 da db = U_n \lambda V_{\text{bubble}}$$

*inside bubble*

Total volume of bubble

The 2 goes away since  $2 da db$  is volume element

$$\kappa = \frac{1}{6} U_n \lambda V_{\text{bubble}}$$

$$V_{\text{bubble}} = \begin{aligned} &= \text{area in 2D} \\ &= \text{volume in 3D} \end{aligned}$$

Now this depends on path length  $\lambda$ . This can be much larger than for untrapped fluid. Real swimmer probably in between

## Lecture 19: Biomixing, part 4: Viscous swimmer

### 1 The dumbbell swimmer

The Green's function for the Stokes equation

$$-\nabla p + \mu \nabla^2 \mathbf{u} = -\mathbf{f} \delta(\mathbf{r}), \quad \nabla \cdot \mathbf{u} = 0, \quad (1)$$

is  $\mathbf{f} \cdot \mathbb{G}(\mathbf{r})$ , where  $\mathbb{G}(\mathbf{r})$  is the Oseen tensor:

$$\mathbb{G}(\mathbf{r}) = \frac{1}{8\pi\mu\|\mathbf{r}\|} \left( \mathbb{I} + \frac{\mathbf{r}\mathbf{r}}{\|\mathbf{r}\|^2} \right). \quad (2)$$

We model the dumbbell swimmer as two Stokeslets along the  $z$  axis Hernandez-Ortiz *et al.* (2005):

$$\mathbf{u}(\mathbf{r}) = F \hat{\mathbf{z}} \cdot \mathbb{G}(\mathbf{r} - A \hat{\mathbf{z}}) + f \hat{\mathbf{z}} \cdot \mathbb{G}(\mathbf{r} - a \hat{\mathbf{z}}). \quad (3)$$

Force balance then dictates

$$F + f = 0, \quad (4)$$

so that

$$\mathbf{u}(\mathbf{r}) = F \hat{\mathbf{z}} \cdot (\mathbb{G}(\mathbf{r} - A \hat{\mathbf{z}}) - \mathbb{G}(\mathbf{r} - a \hat{\mathbf{z}})). \quad (5)$$

Setting  $A = 0$  momentarily, note that

$$\lim_{a \rightarrow 0} \frac{1}{a} (\mathbb{G}(\mathbf{r}) - \mathbb{G}(\mathbf{r} - a \hat{\mathbf{z}})) = \hat{\mathbf{z}} \cdot \nabla \mathbb{G}(\mathbf{r}). \quad (6)$$

Recall that  $\nabla \mathbf{r} = \mathbb{I}$ ,  $\nabla \|\mathbf{r}\| = \hat{\mathbf{r}}$ ; we have

$$\begin{aligned} 8\pi\mu \partial_i \mathbb{G}_{jk}(\mathbf{r}) &= \partial_i \left( \frac{1}{\|\mathbf{r}\|} \right) \left( \delta_{jk} + \frac{r_j r_k}{\|\mathbf{r}\|^2} \right) + \frac{1}{\|\mathbf{r}\|} \partial_i \left( \frac{r_j r_k}{\|\mathbf{r}\|^2} \right) \\ &= -\frac{r_i}{\|\mathbf{r}\|^3} \left( \delta_{jk} + \frac{r_j r_k}{\|\mathbf{r}\|^2} \right) + \frac{1}{\|\mathbf{r}\|^3} (\delta_{ij} r_k + r_j \delta_{ik}) - 2r_i \frac{r_j r_k}{\|\mathbf{r}\|^5} \\ &= \frac{1}{\|\mathbf{r}\|^3} \left( \delta_{ij} - 3 \frac{r_i r_j}{\|\mathbf{r}\|^2} \right) r_k + \frac{1}{\|\mathbf{r}\|^3} (\delta_{ik} r_j - \delta_{jk} r_i). \end{aligned}$$

The first term (symmetric in  $i$  and  $j$ ) is the *stresslet*:

$$\mathbb{S}_{ijk} := \frac{1}{8\pi\mu\|\mathbf{r}\|^3} \left( \delta_{ij} - 3\frac{r_i r_j}{\|\mathbf{r}\|^2} \right) r_k. \quad (7)$$

The second term (antisymmetric in  $i$  and  $j$ ) is the *rotlet*:

$$\mathbb{R}_{ijk} := \frac{1}{8\pi\mu\|\mathbf{r}\|^3} (\delta_{ik}r_j - \delta_{jk}r_i). \quad (8)$$

Hence,

$$\begin{aligned} \mathbf{u}(\mathbf{r}) &\sim aF \hat{\mathbf{z}}\hat{\mathbf{z}} : \nabla\mathbb{G}(\mathbf{r}) \\ &= \frac{aF}{8\pi\mu\|\mathbf{r}\|^2} \left\{ \left( 1 - 3\frac{zz}{\|\mathbf{r}\|^2} \right) \frac{\mathbf{r}}{\|\mathbf{r}\|} + \mathcal{O}(a^2/\|\mathbf{r}\|^2) \right\} \end{aligned} \quad (9)$$

is the far-field form of the dipole—a pure stresslet. We simply replace  $a$  by  $(a - A)$  to restore  $A \neq 0$ , since the corrections incurred are of higher order.

Now assume our dumbbell swimmer is in a frame moving at constant velocity  $U\hat{\mathbf{z}}$ , so there is an apparent flow  $-U\hat{\mathbf{z}}$ . We take the positions  $a(t)$  and  $A(t)$  to be time-periodic in the comoving frame. The forces exerted on the fluid are due to drag on a sphere of radius  $R$  at  $x = A(t)$  and a sphere of radius  $r(t)$  at  $x = a(t)$ :

$$F(t) = 6\pi\mu R(U + \dot{A}(t)), \quad f(t) = 6\pi\mu r(t)(U + \dot{a}(t)). \quad (10)$$

We take the frame to move at the mean swimming velocity  $U$ ; this is obtained from the constraint that the time-averaged velocities of the Stokeslets must vanish in the comoving frame:

$$\langle \dot{A} \rangle = \langle \dot{a} \rangle = 0. \quad (11)$$

From (10) and (4), we have

$$R(U + \dot{A}(t)) = -r(t)(U + \dot{a}(t)), \quad (12)$$

which upon time-averaging gives

$$RU = -U\langle r \rangle - \langle r\dot{a} \rangle, \quad (13)$$

and so the mean swimming velocity is

$$U = -\langle r\dot{a} \rangle / (R + \langle r \rangle). \quad (14)$$

The prescribed functions are  $\dot{a}(t)$  and  $r(t)$ ;  $U$  is then obtained from (14) and  $\dot{A}(t)$  from (12).

The simplest time-dependence we can put is

$$a(t) = a_0 + a_1 \cos(2\pi t/\tau), \quad r(t) = r_0 + r_1 \sin(2\pi t/\tau + \phi) \quad (15)$$

which gives

$$U = \frac{\pi a_1 r_1 \cos \phi}{\tau(r_0 + R)}. \quad (16)$$

The phase  $\phi = 0$  yields the fastest mean swimming velocity: it corresponds to the sphere expanding during the power stroke, and shrinking during the recovery stroke. We thus set  $\phi = 0$  for simplicity. We then have

$$\dot{A}(t) = -U - r(t)(U + \dot{a}(t))/R \quad (17)$$

$$= \frac{\pi a_1 r_1}{R\tau} \left\{ \left( \frac{2r_0}{r_1} - \frac{r_1}{r_0 + R} \right) \sin(2\pi t/\tau) - \cos(4\pi t/\tau) \right\} \quad (18)$$

which can be integrated to find  $A(t) = \int^t \dot{A} dt$ . We choose the integration constant to be zero, so that the swimmer's main body oscillates about the origin in the comoving frame.

The far-field stresslet coefficient from (9) with  $a \rightarrow (a - A)$  is

$$\frac{(a - A)F}{8\pi\mu} = -\frac{3}{4}(a - A)r(U + \dot{a}), \quad (19)$$

which is a complicated function involving many harmonics. The time-averaged coefficient of the stresslet has the simple form

$$\frac{1}{8\pi\mu} \langle (a - A)F \rangle = \frac{3\pi a_0 a_1 r_1}{4\tau(1 + r_0/R)} = \frac{3}{4} a_0 R U. \quad (20)$$

## 2 Particle displacement

We now address the question of particle displacements due to a moving stresslet, when the stresslet is aligned with the direction of motion. The other case (when the stresslet is perpendicular to the direction of motion) is more complicated, since it is no longer axially symmetric.

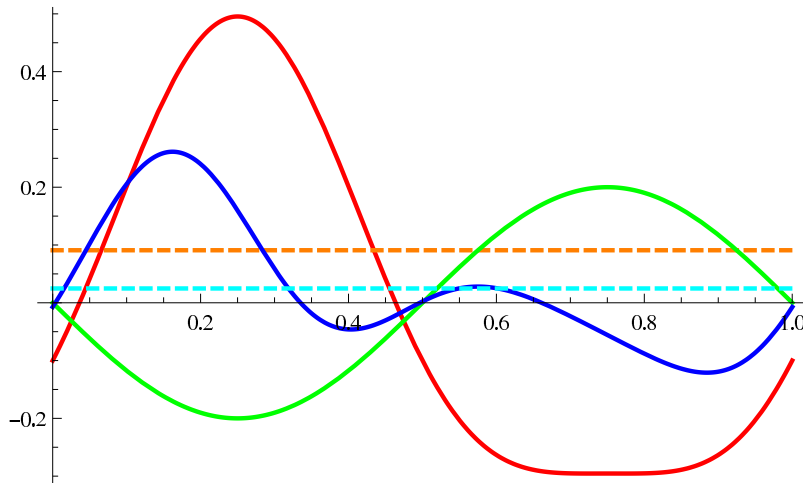


Figure 1: Body velocity  $\dot{A}$  (red), flagellum velocity  $\dot{a}$  (green), and stresslet coefficient (blue) as a function of time. The dashed lines are the time-averaged stresslet coefficient (cyan) and swimming velocity (orange).

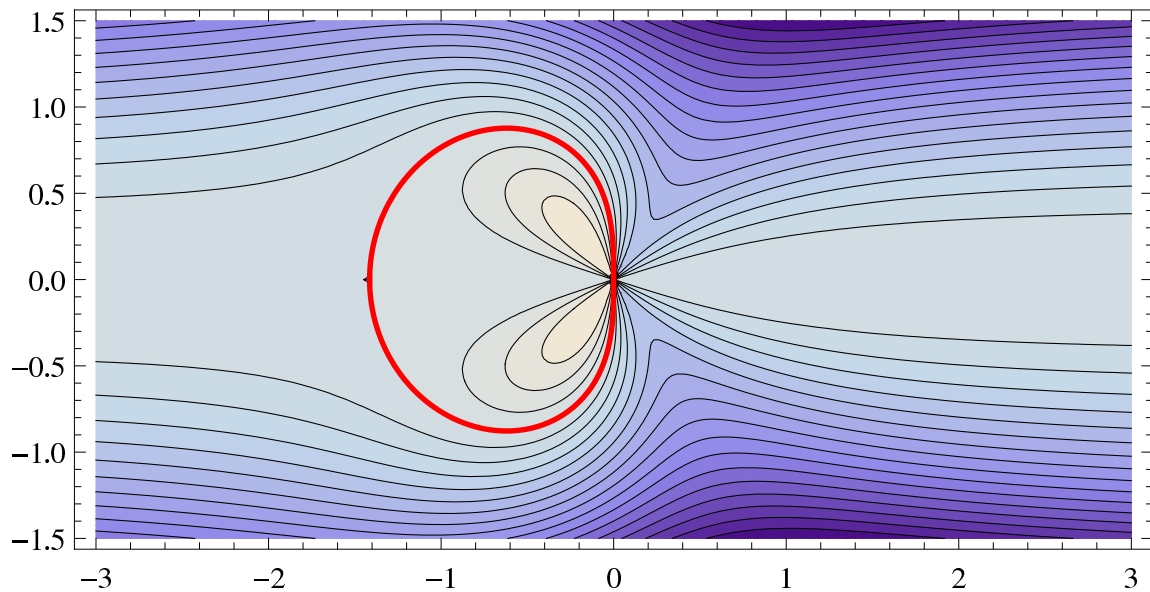


Figure 2: The streamlines in the comoving frame for the moving stresslet (Eq. (21)). The thick line shows the 'atmosphere' (closed streamline in the comoving frame).



## 2.1 Streamline pattern and atmosphere

We take the velocity field in a comoving frame to be

$$\mathbf{u}_{\text{comov}}(\mathbf{r}) = -\hat{\mathbf{z}} + \beta \left( 1 - 3 \frac{zz}{\|\mathbf{r}\|^2} \right) \frac{\mathbf{r}}{\|\mathbf{r}\|^3}, \quad (21)$$

so that the stresslet is moving at unit speed in the  $\hat{\mathbf{z}}$  direction, with a resulting apparent flow in the  $-\hat{\mathbf{z}}$  direction. The streamfunction in the comoving frame is

$$\psi_{\text{comov}}(\rho, z) = -\frac{1}{2}\rho^2 - \beta \frac{z\rho^2}{\|\mathbf{r}\|^3}, \quad (22)$$

with

$$u_\rho = -\rho^{-1}\partial_z\psi, \quad u_z = \rho^{-1}\partial_\rho\psi. \quad (23)$$

Figure 2 shows the streamline pattern in the comoving frame (for  $\beta = 1$ ), which suggests the presence of an atmosphere: a closed streamline in the comoving frame. We can find the equation for the atmosphere by solving  $\psi_{\text{comov}} = 0$ ,

$$-\frac{1}{2} - \beta \frac{z}{(\rho^2 + z^2)^{3/2}} = 0, \quad (24)$$

so that

$$\rho_{\text{atm}}^2(z) = -z \left( z + (2\beta)^{2/3}(-1/z)^{1/3} \right). \quad (25)$$

Note that  $\rho_{\text{atm}}(0) = \rho_{\text{atm}}(-\text{sign}(\beta)\sqrt{2|\beta|}) = 0$ , which means the atmosphere extends from  $z = 0$  to  $z = -\text{sign}(\beta)\sqrt{2|\beta|}$ . The atmosphere is plotted as a thick line in Fig. 2.

We also have an explicit expression for the volume of the atmosphere, for instance for  $\beta > 0$ :

$$V_{\text{atm}} = \int_{-\sqrt{2\beta}}^0 \pi \rho_{\text{atm}}^2(z) dz = \frac{8}{15} \sqrt{2} \pi |\beta|^{3/2}, \quad (26)$$

where the final expression is also valid for  $\beta < 0$ . The volume is useful for computing the transport due to particles trapped in the atmosphere.

## 2.2 Displacement for far field

Recall the definition of the two impact parameters,  $a > 0$  and  $b$  (see Lin *et al.* (2011)). We set  $U = \beta = 1$ , and define

$$\mathbf{u}(\mathbf{r}, t) = \left( 1 - 3 \frac{ZZ}{\|\mathbf{R}\|^2} \right) \frac{\mathbf{R}}{\|\mathbf{R}\|^3}, \quad (27)$$

where

$$\mathbf{R}(t) = (X, Y, Z(t)) = (x, y + a, z + b - t). \quad (28)$$

The stresslet starts at  $(0, -a, -b)$  at  $t = 0$  and proceeds to move in the positive  $z$  direction. The particle starts at  $\mathbf{r} = 0$  and its motion takes place in the  $y$ - $z$  plane. If the particle is far from the swimmer, then  $\mathbf{r}$  remains small throughout the trajectory, and we can expand to leading order in  $\|\mathbf{r}\|$ :

$$u_y = \frac{a(a^2 - 2(b-t)^2)}{H^5(a, b-t)} + O(\|\mathbf{r}\|), \quad u_z = \frac{(b-t)(a^2 - 2(b-t)^2)}{H^5(a, b-t)} + O(\|\mathbf{r}\|). \quad (29)$$

where the hypotenuse function is

$$H(a, b) := \sqrt{a^2 + b^2}. \quad (30)$$

At this order the particle feels a velocity field that is independent of its position. We can then solve for the particle motion by integrating  $\dot{y} = u_y$  and  $\dot{z} = u_z$ :

$$y(t) = \frac{ab}{H^3(a, b)} - \frac{a(b-t)}{H^3(a, b-t)}, \quad (31a)$$

$$z(t) = \frac{H^2(a, \sqrt{2}b)}{H^3(a, b)} - \frac{H^2(a, \sqrt{2}(b-t))}{H^3(a, b-t)} \quad (31b)$$

valid to leading order in  $\|\mathbf{r}\|$ . Both coordinates achieve extrema at  $t = b \pm \frac{1}{\sqrt{2}}a$ , and  $z(t)$  has an additional extremum at  $t = b$ . The fact that both coordinates achieve extrema at the same time is reflected by the two ‘cusps’ visible in Fig. 3. The coordinates of the two cusps are

$$y_{\text{cusp}} = \pm \frac{2}{3\sqrt{3}} a^{-1} + \frac{ab}{H^3(a, b)}, \quad z_{\text{cusp}} = -\frac{4}{3}\sqrt{\frac{2}{3}} a^{-1} + \frac{H^2(a, \sqrt{b})}{H^3(a, b)}. \quad (32)$$

After a time  $t = \lambda$  (recall that  $U = 1$ , so  $\lambda = U\tau = 1$ ), the net total displacement in each direction is  $y(\lambda)$  and  $z(\lambda)$ . Examining Fig. 3 and using the location of the cusps (32) we find that the maximum displacement in  $y$  is bounded:

$$|y(\lambda)| \leq \frac{2}{3\sqrt{3}} a^{-1} + \frac{a|b|}{H^3(a, b)} \leq \frac{4}{3\sqrt{3}} a^{-1}. \quad (33)$$

The maximum displacement is achieved for  $\lambda = \sqrt{2}a$ ,  $b = \pm a/\sqrt{2}$ . The displacement in  $z$  is also bounded:

$$|z(\lambda)| \leq -\frac{4}{3}\sqrt{\frac{2}{3}} a^{-1} + \frac{H^2(a, \sqrt{2}b)}{H^3(a, b)} \leq \left(\frac{4}{3}\sqrt{\frac{2}{3}} - 1\right) a^{-1}, \quad (34)$$

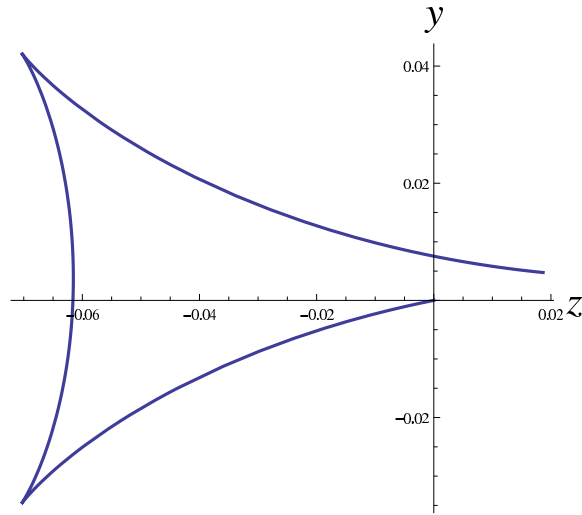


Figure 3: Particle trajectory for  $a = 10$ ,  $b = 50$ , with  $t$  running from 0 to 150.

where the maximum is achieved for  $\lambda = b = a/\sqrt{2}$ .

In the limit of infinite path length, we have

$$|y(\lambda)| \sim \frac{ab}{H^3(a, b)}, \quad |z(\lambda)| \sim \frac{H^2(a, \sqrt{2}b)}{H^3(a, b)}, \quad \lambda \rightarrow \infty. \quad (35)$$

If we then take  $b$  to  $\infty$  as well (the swimmer starts very far away), the displacement goes to zero. In this case, we have to expand the velocity field to next order to obtain the net displacement. It will not be necessary to do so here.

The total net displacement is

$$\Delta_\lambda(a, b) := \sqrt{y^2(\lambda) + z^2(\lambda)}. \quad (36)$$

To compute the effective diffusivity, we can evaluate the integral

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} a^2 \Delta_\lambda^2(a, b) db d(\log a) = 4\lambda \quad (37)$$

whose integrand is plotted in Fig. 4. The resemblance to the numerical solution in Fig. 4(b) of Lin *et al.* (2011) is striking. The contributions to the integral (37) are  $\frac{4}{3}\lambda$  from  $y(\lambda)$  and  $\frac{8}{3}\lambda$  from  $z(\lambda)$ . The displacement values for small  $a$  are not well predicted by this small displacement approach, but since the integral (4) downplays the importance of small  $a$  this will not lead to a large error.

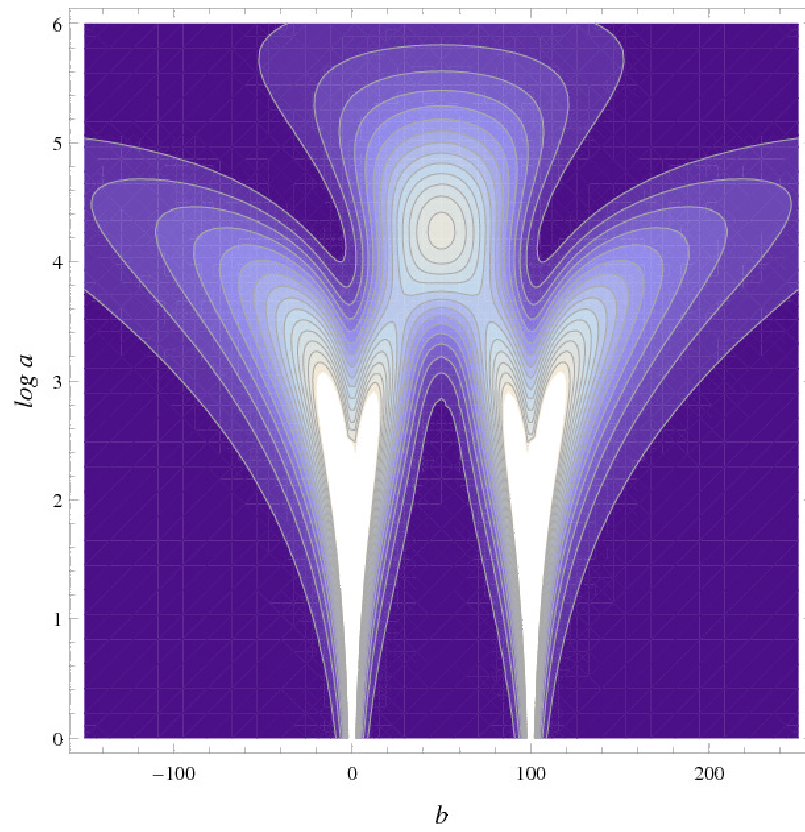


Figure 4: The integrand  $a^2 \Delta_\lambda^2(a, b)$  for  $\lambda = 100$ .

Finally, we can use the result (37) in Eq. (2.6) of Lin *et al.* (2011) to compute the effective diffusivity:

$$D_{\text{eff}} = \frac{4}{3}\pi\beta^2 nU \quad (38)$$

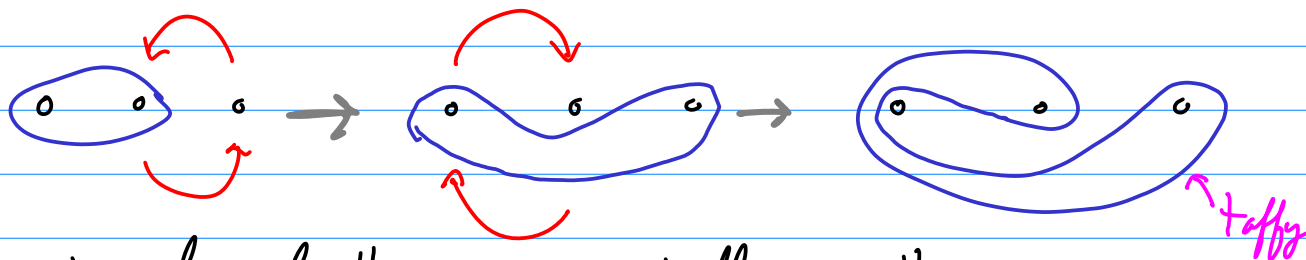
where we restored all units ( $\beta$  has units of squared length times velocity). The path length  $\lambda$  drops out. In Lin *et al.* (2011) the definition of  $\beta$  is slightly different (replace our  $\beta$  by  $\frac{3}{4}\beta\ell^2$  to recover their definition). Converting to their prefactor, we find a numerical coefficient of 2.356, whereas the numerical result in Lin *et al.* (2011) is 2.1 — a 10% difference, which is not bad for an analytic result! The difference is probably due to our large- $a$  overestimating the displacement for small  $a$ .

## References

- Hernandez-Ortiz, J. P., Dtolz, C. G., & Graham, M. D. (2005). *Phys. Rev. Lett.* **95**, 204501.
- Lin, Z., Thiffeault, J.-L., & Childress, S. (2011). *J. Fluid Mech.* **669**, 167–177. <http://arxiv.org/abs/1007.1740>.

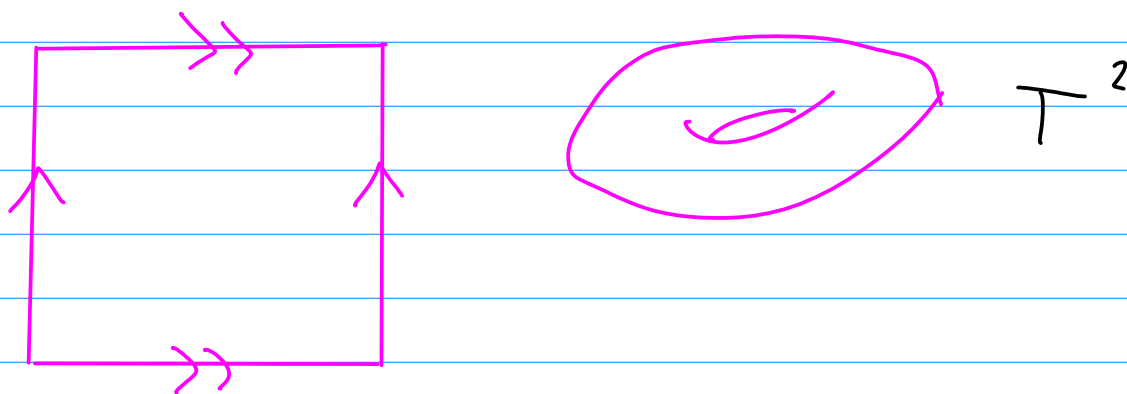
## Lecture 20: Topological mixing on the torus

Stirring by moving rods [movie]  $\left\{ \begin{array}{l} \text{fluids (viscous)} \\ \text{elastic bodies (bread, taffy)} \end{array} \right.$



Repeat: line length grows exponentially in this case.

How do we characterize this? A lot of insight obtained from first considering the torus.



Homeomorphisms  $T^2 \rightarrow T^2$

← orientation-preserving

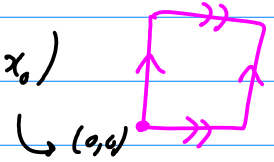
$\text{Homeo}^+(T^2)$ .

Invertible, continuous  
with continuous inverse.

$\text{Homeo}^+(T^2)$  is a group under composition of functions.

Define:

$$\text{MCG}(T^2) = \text{Homeo}^+(T^2) / \text{isotopy}$$

Mapping Class  
Group of  $T^2$ (inherits the group structure of  $\text{Homeo}^+(T^2)$ )assume they have  
a fixed pointWhat does  $\text{MCG}(T^2)$  look like? $\text{Homeo}^+(T^2, x_0)$ Consider an induced homomorphism on  $\pi_1(T^2, x_0)$ fundamental group of  $T^2$ ,  
with loops based at  $x_0$ 

$$f: T^2 \rightarrow T^2, \quad f_*: \pi_1(T^2) \rightarrow \pi_1(T^2)$$

 $\mathbb{Z}^2$ 

linear

$$f_*([f] + [g]) = f_*[f] + f_*[g]$$

Hence,  $f_*$  given by matrices

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad a, b, c, d \in \mathbb{Z}$$

 $\in \text{GL}(2, \mathbb{Z})$ But also  $f \circ f^{-1} = \text{id} \Rightarrow f_* \circ f_*^{-1} = I$  so  $f_*$  invertible. $ad - bc \neq 0$ Let  $m = ad - bc \neq 0$ . We have also  $f_*^{-1}: \mathbb{Z}^2 \rightarrow \mathbb{Z}^2$ , so

$$f_*^{-1}: \frac{1}{m} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}, \text{ so need all entries } \in \mathbb{Z}.$$

 $\hookrightarrow m$  divides every entrylet  $a = m\alpha$ ,  $b = m\beta$ ,  $c = m\gamma$ ,  $d = m\delta$ ,  $\alpha, \beta, \gamma, \delta$  integers.

$$\text{Then } m = ad - bc = m^2(\alpha\delta - \beta\gamma) \Rightarrow 1 = m(\alpha\delta - \beta\gamma).$$

Since all integers, need  $m = \pm 1$ .  $m = +1 \Rightarrow$  orientable.

Hence,  $\boxed{\text{MCG}(T^2) = \text{SL}(2, \mathbb{Z})}$  *Why is this?*

Now, how do we classify the elements of this group?

Look at eigenvalues.  $\det\left(\underbrace{\begin{pmatrix} a & b \\ c & d \end{pmatrix}}_M - xI\right) = x^2 - (a+d)x + \underbrace{ad-bc}_1$   
 $M (= f_x)$

Let  $\tau = a+d$  (trace)

note:  $p(M) = M^2 - \tau M + I = 0$   
 Cayley-Hamilton thm

Characteristic polynomial:  $p(x) = x^2 - \tau x + 1$

Eigenvalues:  $x = \frac{1}{2}(\tau \pm \sqrt{\tau^2 - 4})$  So  $|\tau| = 2$  important.

Let's examine different cases.

1)  $|\tau| < 2$ .  $\tau = -1, 0, 1$ .

If  $\tau = 0$ , then  $p(M) = M^2 + I = 0 \Rightarrow M^2 = -I \Rightarrow \boxed{M^4 = I}$



$$\text{If } \tau = \pm 1, \quad p(M) = M^2 \mp M + I \Rightarrow M^2 = \pm M - I$$

$$M^3 = M(\pm M - I)$$

Either way, we can write

$$M^{12} = I, \quad |\tau| < 2$$

$$= \pm M^2 - M$$

$$= \pm(\pm M - I) - M$$

$$= \mp I \quad \boxed{M^6 = I}$$

This is called finite-order. After applying  $f$  enough times, it is isotopic to the identity map.

2)  $|\tau| = 2$ : Then eigenvalues are both  $\pm 1$  ( $= \tau/2$ )

$$M^2 \mp 2M + I = (M \mp I)^2 = 0 \Rightarrow M = \pm I + N, \quad N^2 = 0$$

$\begin{pmatrix} a - \tau/2 \\ c \end{pmatrix}$  is the eigenvector:  $\leftarrow$  This is for  $c \neq 0$  otherwise take  $\begin{pmatrix} b \\ d - \tau/2 \end{pmatrix}$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} a - \tau/2 \\ c \end{pmatrix} = \begin{pmatrix} a(a - \frac{\tau}{2}) + bc \\ c(a - \frac{\tau}{2}) + cd \end{pmatrix} = \begin{pmatrix} a(a - \frac{\tau}{2}) + \overbrace{(bc - ad)}^{-1} + ad \\ c \underbrace{(a + d - \tau/2)}_{\tau} \end{pmatrix}$$

$$= \begin{pmatrix} a(\tau/2) - 1 \\ c\tau/2 \end{pmatrix} = \frac{\tau}{2} \begin{pmatrix} a - \tau/2 \\ c \end{pmatrix} \quad \begin{matrix} \text{since } \frac{2}{\tau} = \tau/2 \\ (\tau/2 = \pm 1) \end{matrix}$$

Hence, the homotopy classes given by  $\begin{pmatrix} a - \tau/2 \\ c \end{pmatrix}$  are invariant (or reverse direction) under  $M$ .

$\Rightarrow$  invariant curve (called reducible)

Let  $R = \begin{pmatrix} 1 & a^{-\tau/2} \\ 0 & c \end{pmatrix}$ . Then:

$$R^{-1}MR = \begin{pmatrix} \tau/2 & 0 \\ 1 & \tau/2 \end{pmatrix} \quad \text{Jordan form}$$

other  
logs not  
invariant

Simplest type:  $M = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$ , so  $M \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ ,  $M \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Next time: case 3)  $|\tau| > 2$  !

## Lecture 21: Anosov homeomorphisms

Recall  $f: T^2 \rightarrow T^2$  homeomorphism (orientation-preserving)

$$f_*: \pi_1(T^2) \rightarrow \pi_1(T^2) \quad f_* \in SL(2, \mathbb{Z})$$

$$f_* = M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad ad - bc = 1$$

$$p(x) = x^2 - \tau x + 1 \quad \text{characteristic polynomial with } \tau = a + d = \text{trace}$$

$$x = \frac{1}{2}(\tau \pm \sqrt{\tau^2 - 4})$$

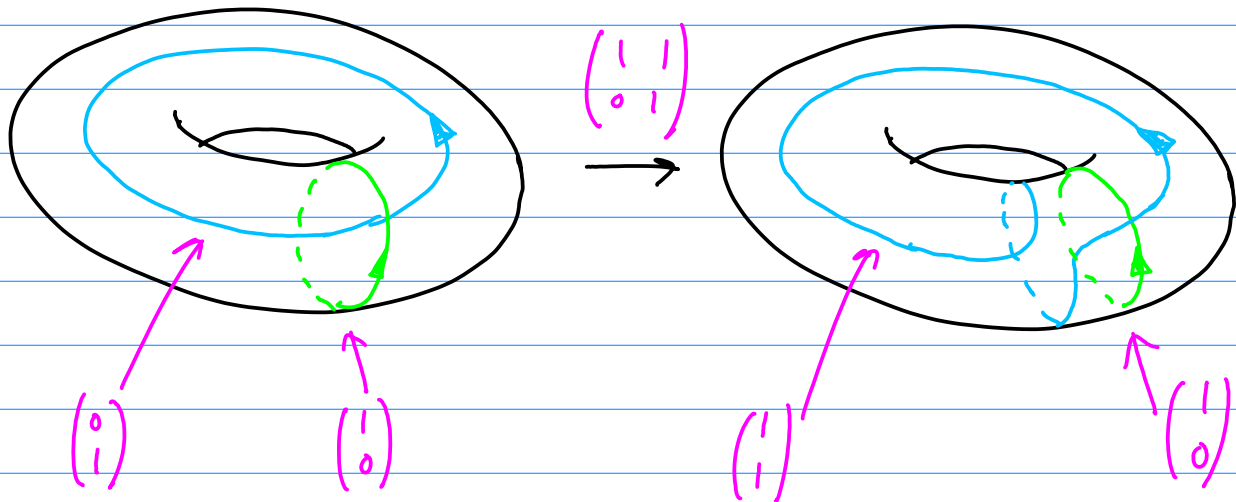
So far: 1)  $|\tau| < 2 \Rightarrow M^{12} = I$  finite-order

2)  $|\tau| = 2 \Rightarrow M = \pm I$  (also finite-order)

or  $MN = \pm N$  for some  $N \in \pi_1(T^2)$

example:

$\Rightarrow$  invariant loop (reducible)



That is:  $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$  leaves the loop  $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$  invariant.

3)  $|\tau| > 2$ : In that case we get two distinct real roots:

$$x_{\pm} = \frac{1}{2}(\tau \pm \sqrt{\tau^2 - 4}), \text{ with } x_+ x_- = 1$$

The roots are inverse of each other, and we define

$$\lambda = \max(|x_+|, |x_-|) > 1.$$

The eigenvectors of  $M$  are  $\pm = \text{sign}(\tau)$

$$u = \begin{pmatrix} \pm\lambda - d \\ c \end{pmatrix}, \quad s = \begin{pmatrix} \pm\lambda^{-1} - d \\ c \end{pmatrix}, \quad \lambda > 1$$

Check: (with  $\tau > 2$ )

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \lambda - d \\ c \end{pmatrix} = \begin{pmatrix} a\lambda - ad + bc \\ c\lambda - cd + cd \end{pmatrix} = \begin{pmatrix} (-\lambda^2 + (a+d)\lambda - 1) + \lambda^2 - \lambda d \\ c\lambda \end{pmatrix} = \lambda \begin{pmatrix} \lambda - d \\ c \end{pmatrix}.$$

Claim:  $\lambda$  is irrational  $\leftarrow$  There are easier ways, but this is nice...

Assume  $\tau > 2$  (positive):  $\leftarrow \tau < -2$  of course has nearly identical proof.

$$\lambda = \frac{1}{2}(\tau + \sqrt{\tau^2 - 4}) = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots}}}$$

$$\lambda^2 - \tau\lambda + 1 = 0 \iff \lambda - \tau + \lambda^{-1} = 0$$

$$\lambda = \underbrace{\tau - 1}_{\text{Positive integer}} + \underbrace{(1 - \lambda^{-1})}_{0 < 1 - \lambda^{-1} < 1} = a_0 + (1 - \lambda^{-1})$$

$\therefore a_0 = \tau - 1$

Fractional part

$$\lambda - a_0 = 1 - \lambda^{-1} = \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots}}}$$

$$\frac{1}{1 - \lambda^{-1}} = a_1 + \frac{1}{a_2 + \frac{1}{\dots}}$$

$\phi = \text{Golden ratio}$

$$1 - \lambda^{-1} = 1 - (\tau - \lambda) \quad \text{since } \lambda^2 - \tau\lambda + 1 = 0$$

$$= (1 - \tau) + \lambda$$

$\phi^2!$

Since  $\lambda = \frac{1}{2}(\tau + \sqrt{\tau^2 - 4})$  is monotonic,  $\min \lambda = \frac{1}{2}(3 + \sqrt{5})$   
for  $|\tau| = 3$ .

$$\text{Hence, } 1 < \frac{1}{1 - \lambda^{-1}} \leq \frac{1}{1 - \max \lambda^{-1}} = 2.61803\dots$$

$$= \frac{1}{\frac{1}{2}(\sqrt{5} - 1)}$$

$$= \frac{2(\sqrt{5} + 1)}{4} = \frac{1}{2}(1 + \sqrt{5}) < 2.$$

$$\max \lambda^{-1} = (\min \lambda)^{-1} = \frac{1}{2}(3 - \sqrt{5})$$

$$\therefore 1 < \frac{1}{1 - \lambda^{-1}} < 2. \quad \text{Conclude: } a_1 = 1.$$

$$\left(\frac{1}{1 - \lambda^{-1}} - 1\right)^{-1} = a_2 + \frac{1}{a_3 + \frac{1}{\dots}}$$

$$\left(\frac{1}{1 - \lambda^{-1}} - 1\right)^{-1} = \left(\frac{\lambda - (\lambda - \lambda^{-1})}{1 - \lambda^{-1}}\right)^{-1} = \left(\frac{1}{\lambda - 1}\right)^{-1} = \lambda - 1.$$

$$\lambda - 1 = a_2 + \frac{1}{a_3 + \frac{1}{\dots}} \iff \lambda = \underbrace{(a_2 + 1)}_{a_0 = \tau - 1} + \frac{1}{a_3 + \frac{1}{\dots}}$$

Same expression as when we started!

$$\therefore a_2 = \tau - 2.$$

One more time:

$$(\lambda - 1 - a_2)^{-1} = a_3 + \frac{1}{a_4 + \frac{1}{\dots}}$$

$$\lambda^2 - \tau\lambda + 1 = 0$$

$$\lambda - \tau + \lambda^{-1} = 0$$

$$\lambda - \tau + 2 = 2 - \lambda^{-1}$$

$$(\lambda - 1 - a_2)^{-1} = (\lambda - \tau + 1)^{-1} = (1 - \lambda^{-1})^{-1}$$

$$\frac{1}{1 - \lambda^{-1}} = a_3 + \frac{1}{a_4 + \frac{1}{\dots}} \quad \therefore \begin{aligned} a_3 &= a_1 \\ a_4 &= a_2 \\ a_5 &= a_1 \\ a_6 &= a_2 \dots \end{aligned}$$

→ We've seen this before!

... Periodic!

Continued fraction representation:

$$\lambda = [\tau - 1; \underbrace{1, \tau - 2, 1, \tau - 2, \dots}_{\text{period-2}}]$$

Of course, this shouldn't have surprised us: the irrational solutions of quadratic equations with integer coefficients are periodic. But this polynomial has a particularly simple form.

More importantly, this shows that  $\lambda$  is irrational for any  $|\tau| > 2$  (negative  $\tau$  proceeds identically)

$[f]$  is the isotopy class of an Anosov homeomorphism with dilatation  $\lambda > 1$ .

## Lecture 22: From the torus to the sphere

Recall:  $f: T^2 \rightarrow T^2$  homeomorphism.

Specific map in an isotopy class:  $f(x,y) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{1}$   
 (Note:  $f(0,0) = (0,0)$ )  
 $ad - bc = 1$

If  $|\text{trace}| > 2$ , we have an Anosov homeomorphism.  
 These are "complex", in the sense that their action on  $\pi_1(T^2)$  gives exponential growth in the number of twists.

Now we want to relate this to something more "physical": sphere (eventually disk)

Consider the map:  $\iota: T^2 \rightarrow T^2$   $\iota \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -x \\ -y \end{pmatrix} \pmod{1}$   
 ( $\iota^2 = \text{id}$  involution)

Fixed points:  $x = -x \pmod{1} \Rightarrow x = -x + n \Rightarrow 2x = n \Rightarrow x = 0, \frac{1}{2}$   
 $y = -y \pmod{1} \Rightarrow y = -y + m \Rightarrow 2y = m \Rightarrow y = 0, \frac{1}{2}$

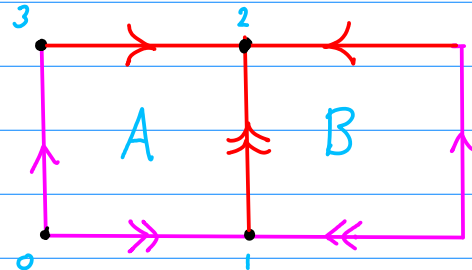
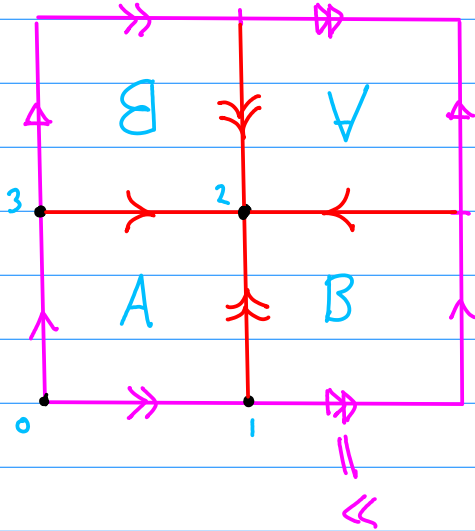
Hence,  $\iota$  has fixed points  $(0,0), (\frac{1}{2}, 0), (0, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2})$

Claim:  $T^2 / \iota \cong S^2$  (each with 4 points removed)

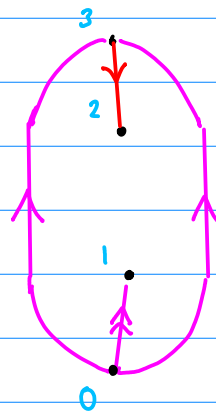
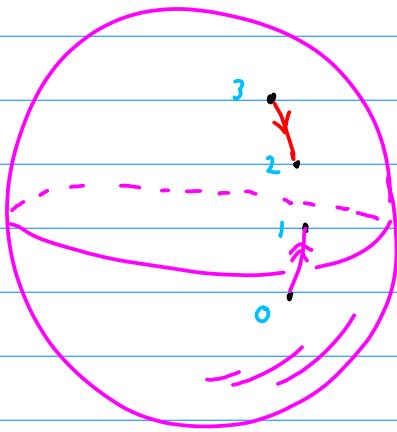
We show this by making the appropriate identifications.

The line  $(x, \frac{1}{2}) \rightarrow (-x, -\frac{1}{2}) = (1-x, \frac{1}{2})$

The line  $(\frac{1}{2}, y) \rightarrow (-\frac{1}{2}, -y) = (\frac{1}{2}, 1-y)$



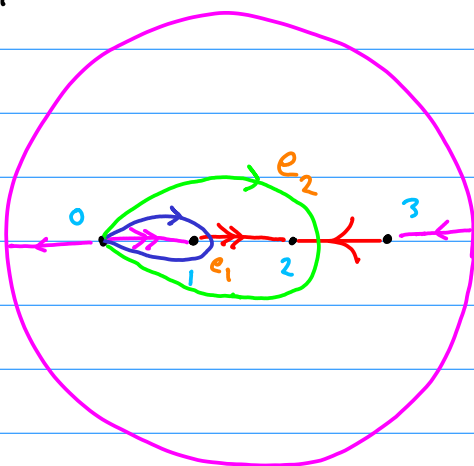
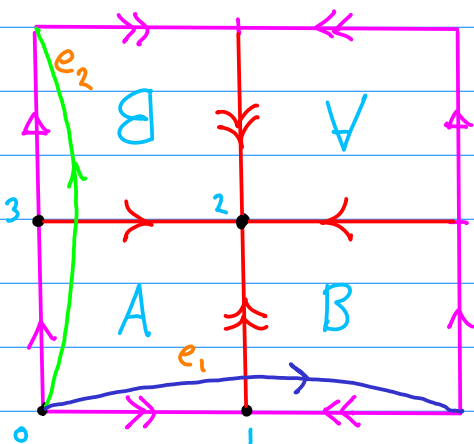
"pacmanize"



$L$  is called the hyperelliptic involution  
The 4 points are the Weierstrass points.

We say that the torus (with 4 punctures) is a branched double cover of the sphere (with 4 punctures).

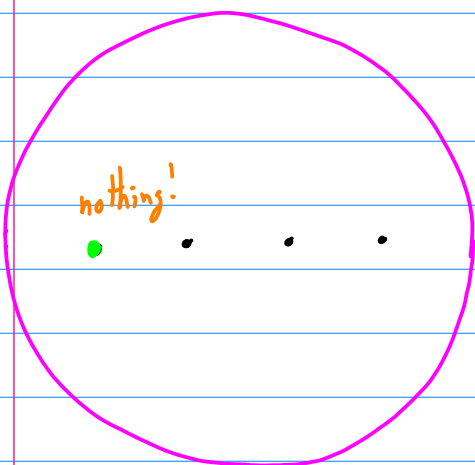
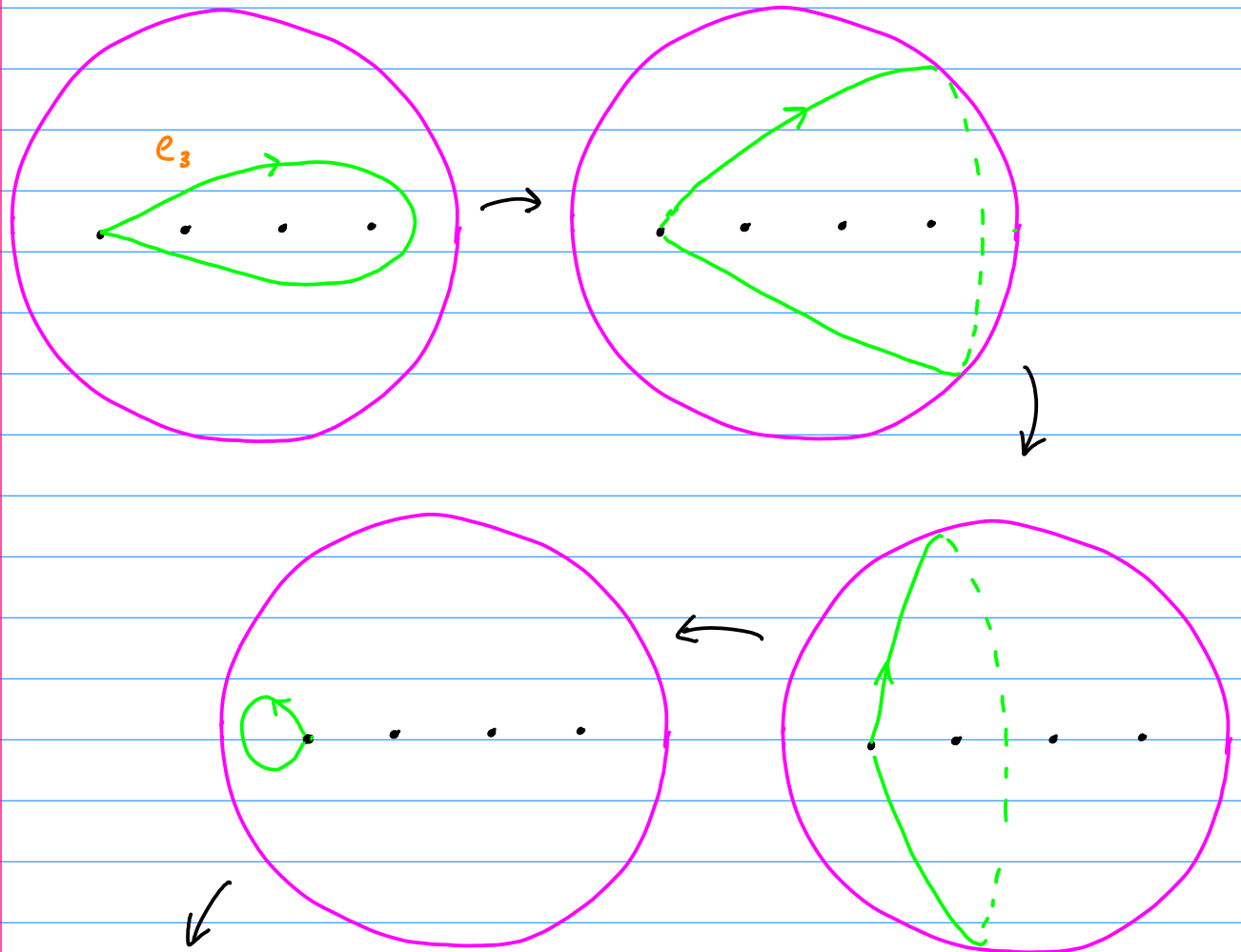
What does an element of  $\pi_1(T^2, (0,0))$  map to?



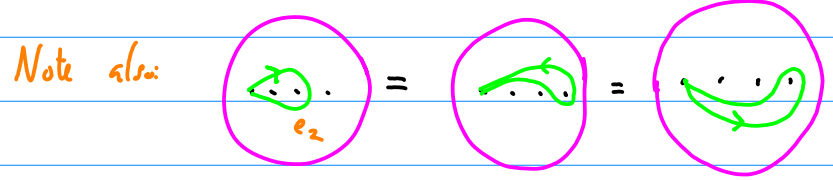
These are loops on the sphere!



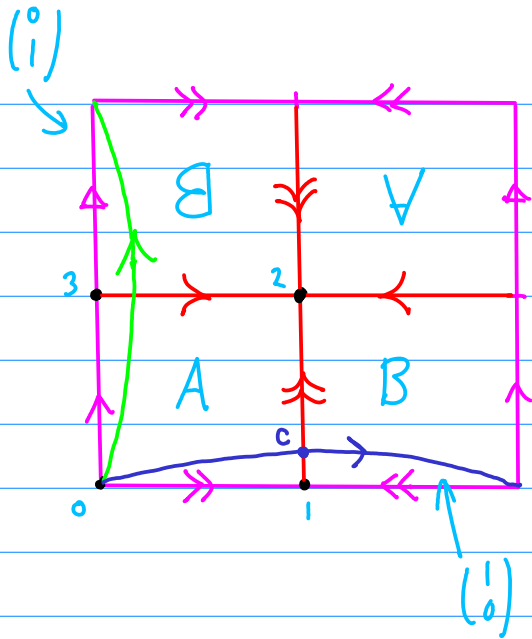
Are we missing a loop in  $S^2 - 4$  punct., around puncture 3?



Hence,  $e_3 = \text{trivial loop} = \text{id in } \pi_1.$



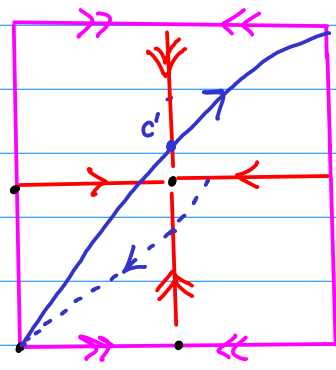
Now consider  $T_1(x,y) = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \text{ mod } 1.$



$$T_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad T_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$c' = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1/2 \\ 3 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 3 + 1/2 \end{pmatrix}$$

$\downarrow T_1$



$$T_1 \begin{pmatrix} 1/2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \quad T_1 \begin{pmatrix} 1/2 \\ 1 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}, \quad T_1 \begin{pmatrix} 0 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$$

$T_1$  fixes 0, 3, interchanges 1 and 2.

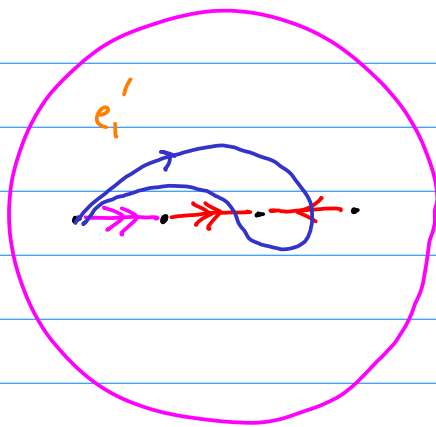
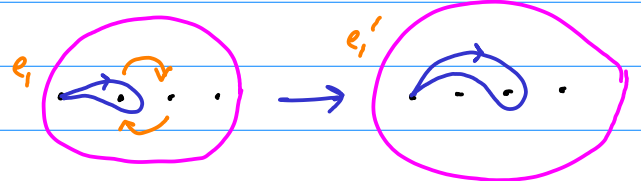


image of  $e_1$  ( $e_2$  unchanged: )



Exactly like swapping 1 & 2 clockwise!

Note:  
Cannot write as  $e_1, e_2$  by dragging, because of the punctures.

$T_1^{-1}$  swaps counterclockwise.

(Can make this clearer by considering homotopy:

$$T_1(x, y, t) = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \text{ mod } 1, \quad 0 \leq t \leq 1$$

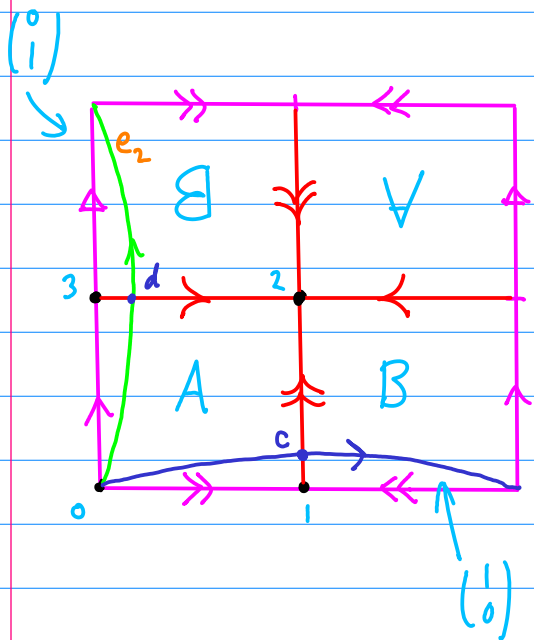
Why doesn't it lead to isotopy?  
→ not involutive on torus

Also define:  $T_2(x,y) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \text{ mod } 1.$

$$T_2 \begin{pmatrix} 1/2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}, \quad T_2 \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1/2 \end{pmatrix}, \quad T_2 \begin{pmatrix} 0 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}$$

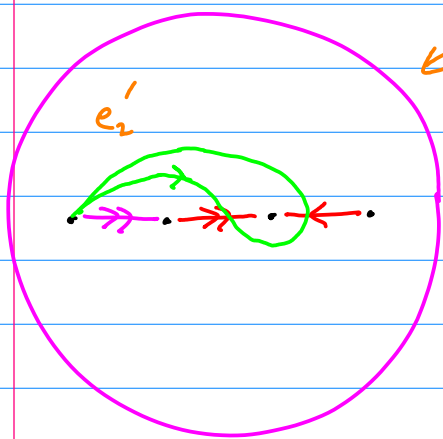
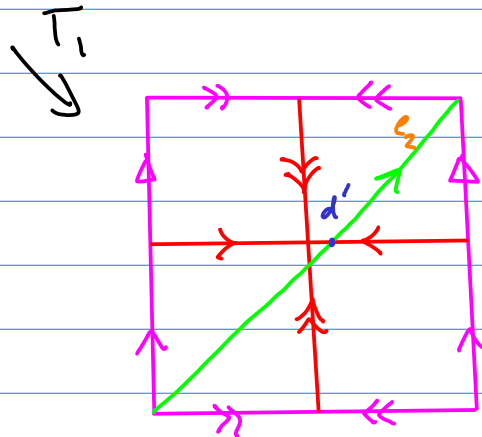
1
1
2
3
3
2

$T_2$  fixes 0, 1, interchanges 2 and 3.

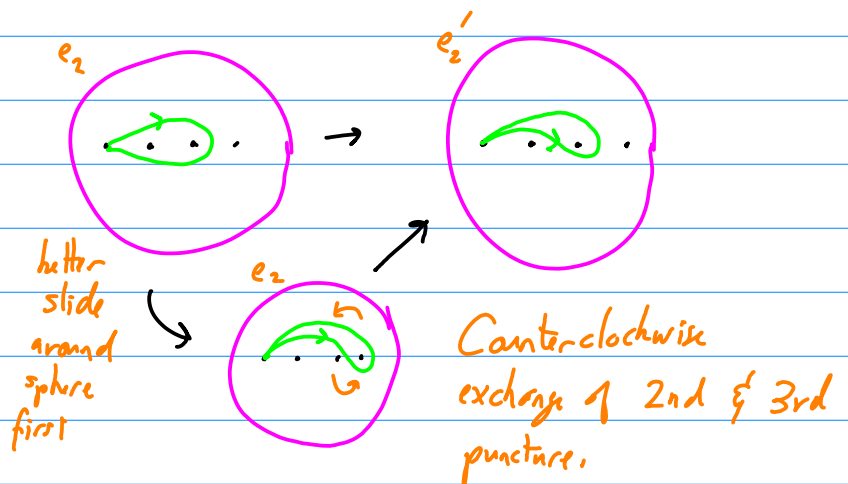


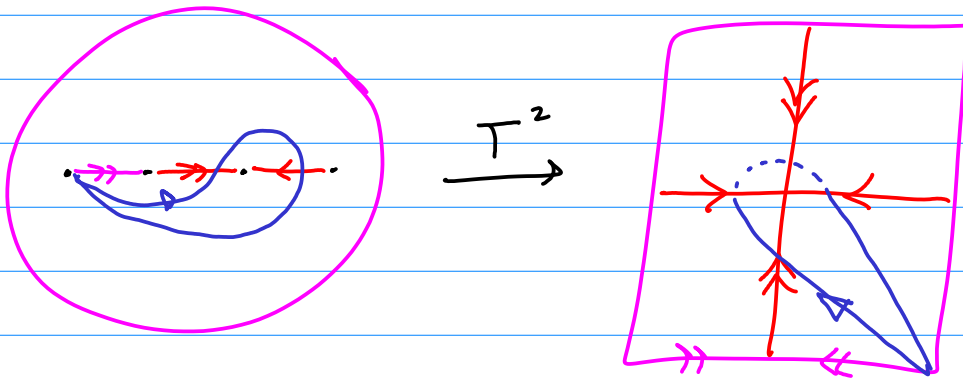
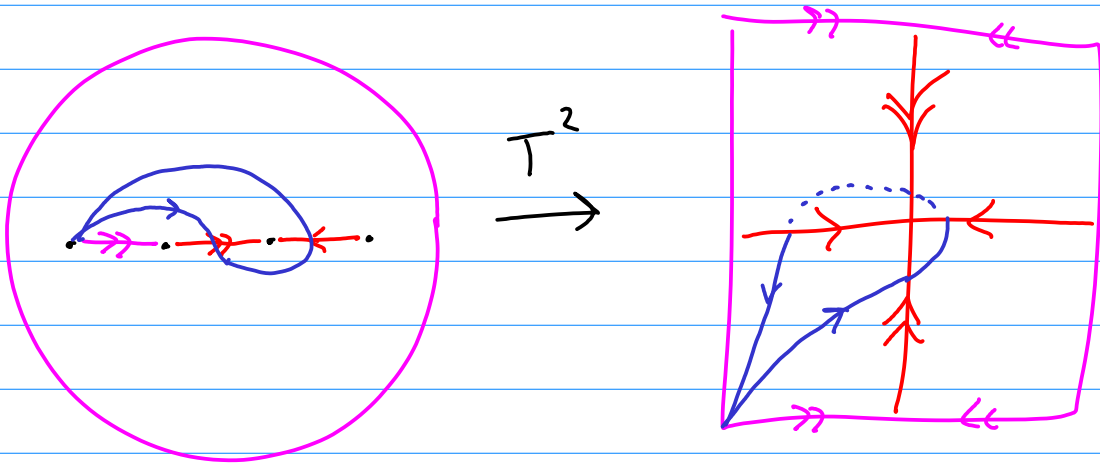
$$T_2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad T_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$d' = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \varepsilon \\ 1/2 \end{pmatrix} = \begin{pmatrix} \varepsilon \\ \varepsilon + 1/2 \end{pmatrix}$$



← image of  $e_2$  ( $e_1$  unchanged) under  $T_2$



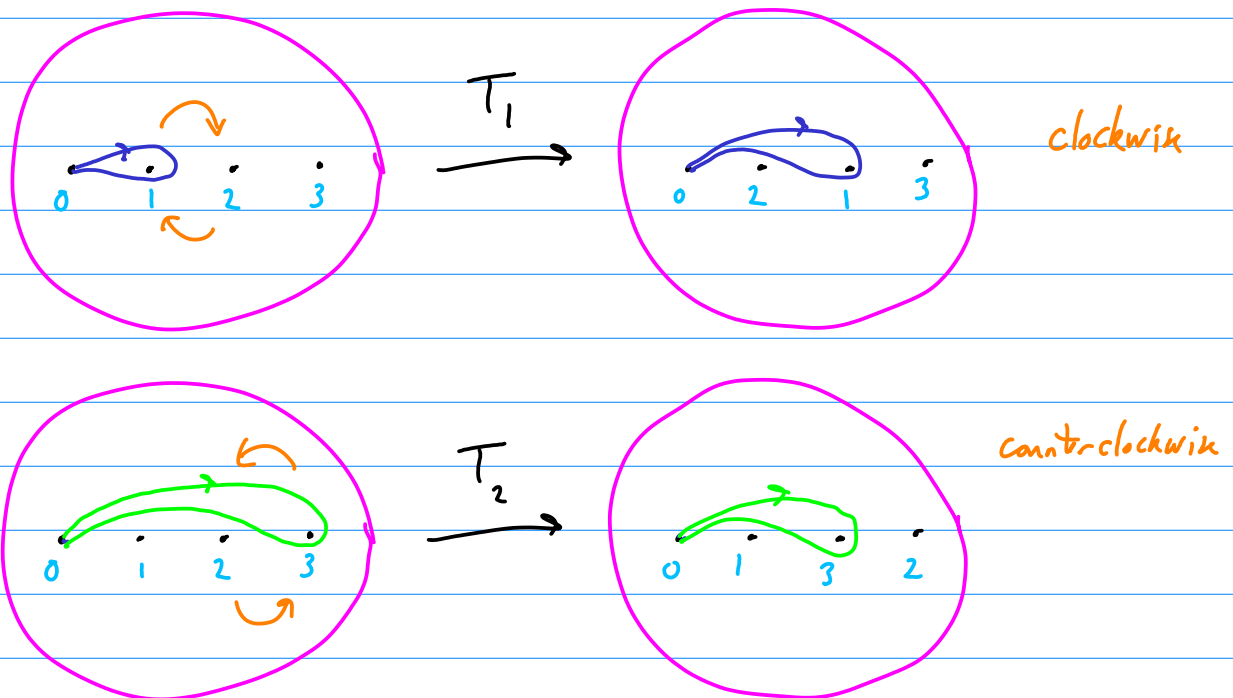


## Lecture 23: Topological stirring

Last time:  $T^2 / \mathcal{L} \cong S^2$      $\mathcal{L} =$  hyperelliptic involution  
(4 branch points)

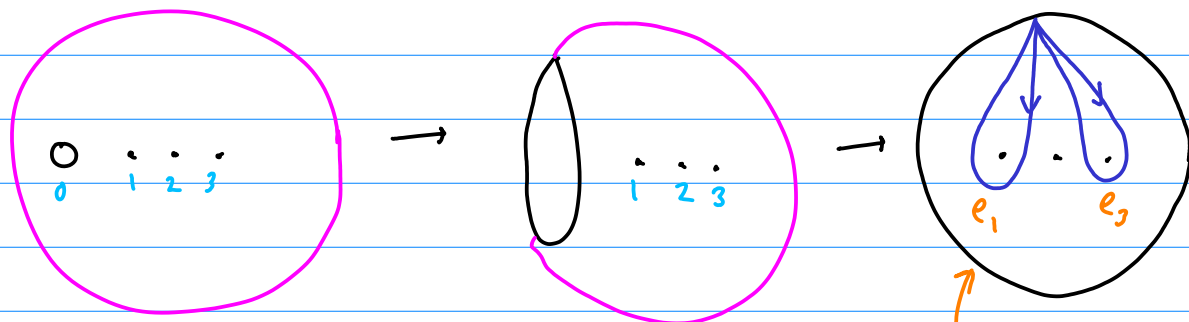
Two special mappings:  $T_1 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{1}$      $T^2 \rightarrow T^2$   
 $T_2 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{1}$

When projected down to the sphere, these can be interpreted as maps that "exchange" punctures  $1 \leftrightarrow 2$  or  $2 \leftrightarrow 3$ .



$T_1$  and  $T_2$  generate the mapping class group of the sphere with 4 punctures, with one puncture fixed.

Now take out a disk at puncture 0:



boundary of a disk  
with 3 punctures, not sphere.

Now we can connect homeos on  $T^2$   
to motion of points (or rods) in a two-dimensional domain.

$\Rightarrow$  stirring a fluid. [movie]

Movie 1: (boyland 1)

$$T_1 T_2^{-1} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}$$

Trace  $< 2$ , so expect some power = id. (recall classification)

$$\begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}^2 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad \begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}^4 = \text{id}$$

Movie 2: (boyland 2)

$$T_1^{-1} T_2^{-1} = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix}$$

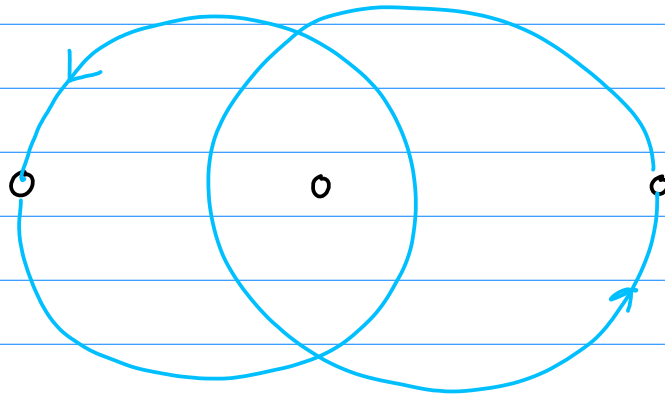
Trace = 3  $\Rightarrow$  Anosov on the torus (pseudo-Anosov on disk)

$$\text{Dilatation } \lambda = \frac{\tau + \sqrt{\tau^2 - 4}}{2} = \frac{3 + \sqrt{5}}{2} = \varphi^2$$

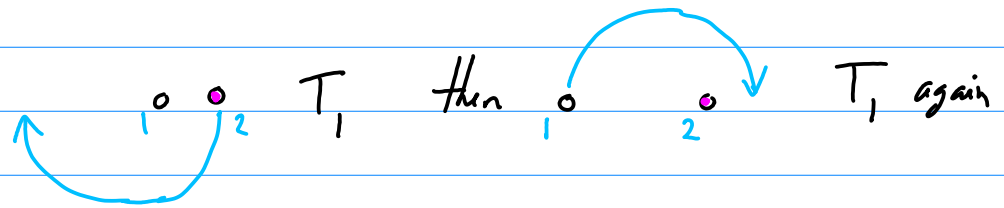
$$\varphi = \frac{1}{2}(1 + \sqrt{5}) \\ = \text{Golden ratio}$$

This tells you something about how "entangled" the fluid motion is.

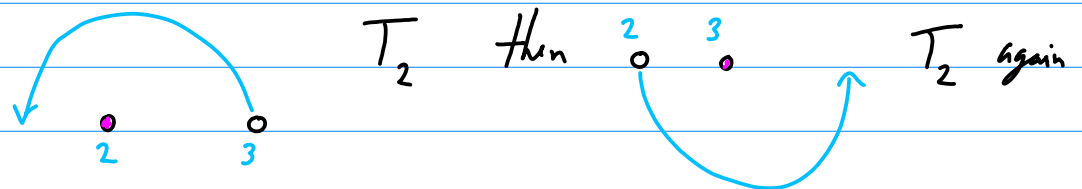
Taffy puller:



Look at movie:



Other side:



Hence, after all the rods return to same initial configuration, we're done

$$T_1^2 T_2^2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}^2 \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}^2 = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix}$$

$$\lambda = \frac{1}{2}(\tau + \sqrt{\tau^2 - 4}) = \frac{1}{2}(6 + \sqrt{32})$$

$$\lambda = 3 + 2\sqrt{2} = (1 + \sqrt{2})^2$$

Silver ratio!



## Lecture 24: The Thurston-Nielsen Classification

Our detailed look at the mapping class group of the torus will help us understand the much more general

### Thurston-Nielsen Classification Theorem.

Let  $\varphi$  be an orientation-preserving diffeomorphism of an orientable surface  $S$  of negative Euler characteristic.

Then  $\varphi$  is isotopic to a diffeomorphism  $\varphi'$  such that either:

1.  $\varphi'$  is finite-order;
2.  $\varphi'$  is reducible;
3.  $\varphi'$  is pseudo-Anosov.

$\varphi'$  is called the TN representative

We look at each case in turn.

1. Finite-order:  $\exists m > 0$  s.t.  $(\varphi')^m = \text{id}$ .

We have seen several examples of this for the torus ( $|a+d| < 2$  and  $[h] = \pm I$ .)

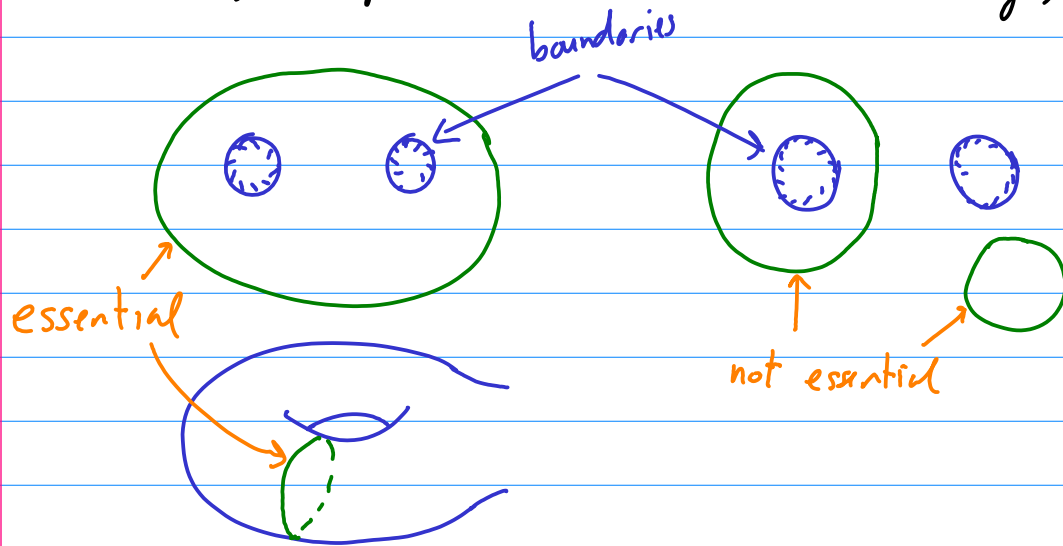
Note that, as for the torus, this does not imply that  $\varphi$  itself is finite-order.

$\Gamma_i \in \text{int}(S)$  Doesn't touch  
the boundary.  
Mixing

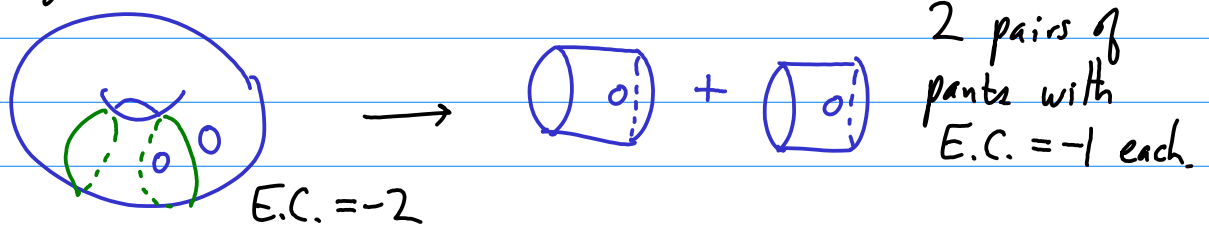
2. Reducible: In this case there is a system  $\Gamma$  of essential simple closed curves  $\{\Gamma_1, \dots, \Gamma_n\}$  ( $n \geq 1$ ) which are pairwise-disjoint with  $\varphi'(\Gamma_i) = \Gamma_i$ . (setwise)

Some clarification: "reducing curves"

- A simple closed curve does not intersect itself
- The  $\Gamma_i$  do not intersect each other (pairwise-disjoint)
- An essential curve  $\Gamma_i$  is not homotopic to a point, to a boundary component, or to each other  $\Gamma_j$ ,  $j \neq i$ .



Another way to capture the "essential" nature of  $\Gamma = \{\Gamma_1, \dots, \Gamma_n\}$  is to say that each connected component of  $S - \Gamma$  has negative Euler characteristic. (E.C.)



So the E.C. actually places a band on the number of reducing curves that can exist on a surface.

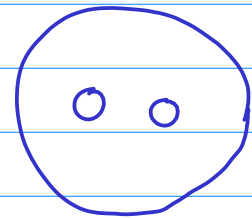
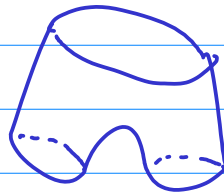
Reminder: Euler characteristic  $\chi(S_{g,b,0}) = 2 - 2g - b$

$$\chi(\text{sphere}) = 2$$

$$\chi(\text{disc}) = 1$$

$$\chi(\text{torus}) = 0$$

$$\chi(\text{pants}) = -1$$



↑  
genus

↑  
# of  
boundaries

Each extra boundary lowers  $\chi$  by 1:

$$\chi\left(\text{disc with pants and two holes}\right) = 0 - 1 - 1 = -2.$$

So the "first" surfaces to which the TN theorem applies are

$$S_{2,0,0}$$

$$S_{1,1,0}$$

$$S_{0,3,0}$$

$$\chi = -2$$

$$\chi = -1$$

$$\chi = -1$$

The other surfaces ( $S_{1,0,0}$ ,  $S_{0,0,0}$ ,  $S_{0,1,0}$ ,  $S_{0,2,0}$ ) have simple MCGs.

The components  $\Gamma_i$  of  $\Gamma = \{\Gamma_1, \dots, \Gamma_n\}$  can be permuted among themselves by the action of  $\varphi'$ .

The consequence of  $\varphi'$  being reducible is that we can cut  $S$  along  $\Gamma$ , and apply the TN theorem to the individual components. We can repeat this as necessary.

The reducible case "looks" like the case  $|a+d|=2$  for the torus, where the diffeo  $\begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}$ , for example, left an infinity of simple closed curves invariant. However, for the torus we do not have to cut along those curves to continue the classification.

This takes us to the most important case.

### 3. pseudo-Anosov (pA)

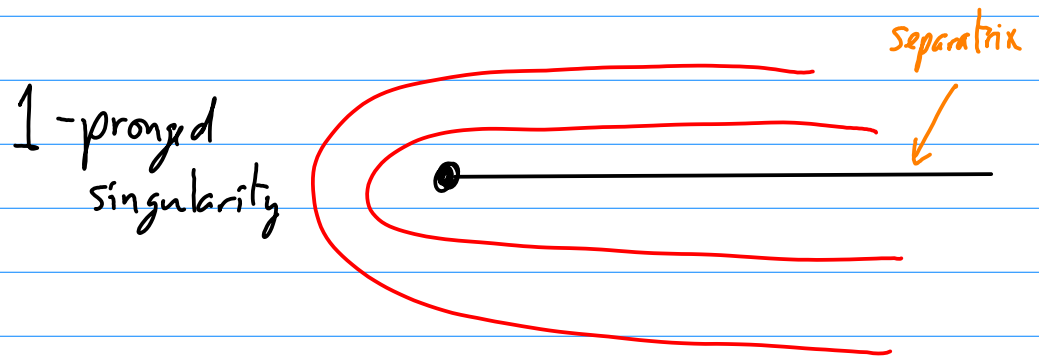
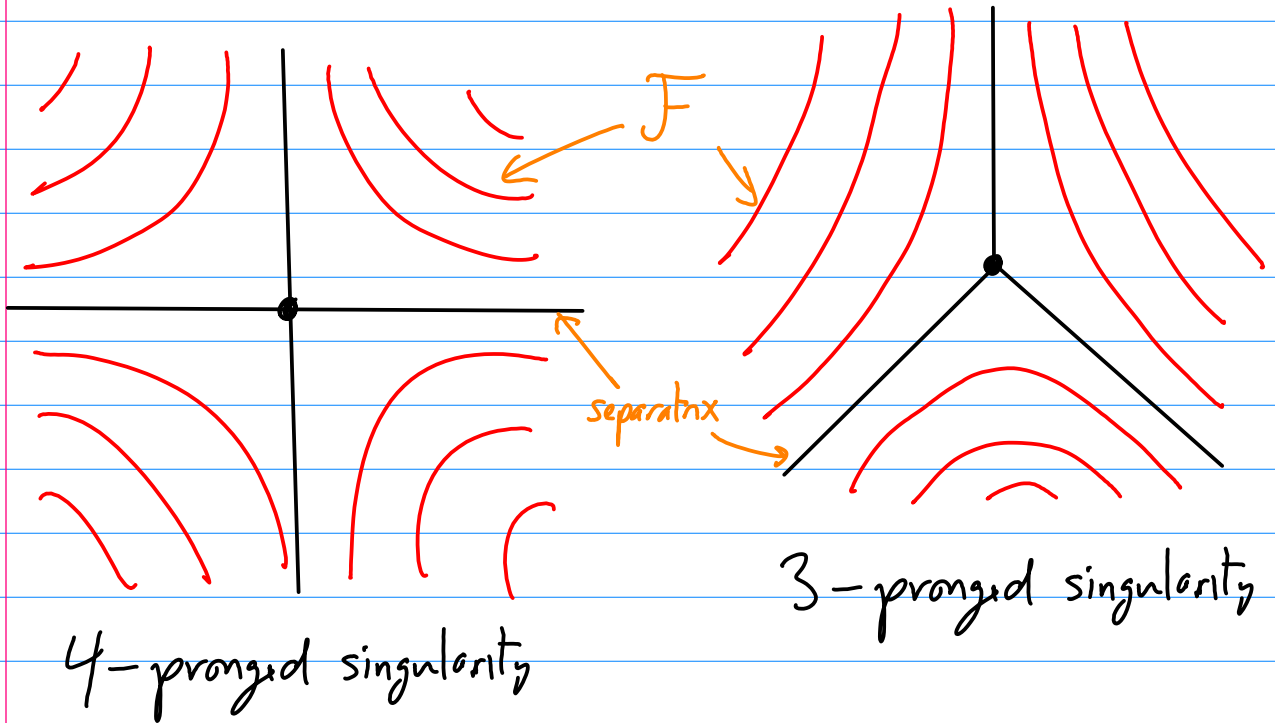
$\varphi$  is a pseudo-Anosov diffeomorphism if there exists a real number  $\lambda > 1$  (the dilatation or expansion constant) and a pair of transverse measured foliations  $(\mathcal{F}^u, \mu^u)$  and  $(\mathcal{F}^s, \mu^s)$  such that

$$\varphi(\mathcal{F}^u, \mu^u) = (\mathcal{F}^u, \lambda \mu^u)$$

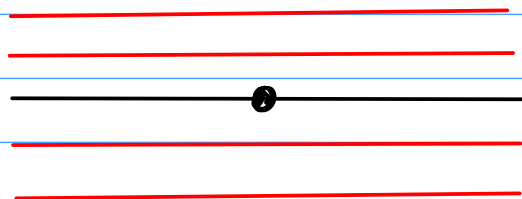
$$\varphi(\mathcal{F}^s, \mu^s) = (\mathcal{F}^s, \lambda^{-1} \mu^s)$$

Unlike the torus case,  $\mathcal{F}^u$  and  $\mathcal{F}^s$  have a finite number of singularities.

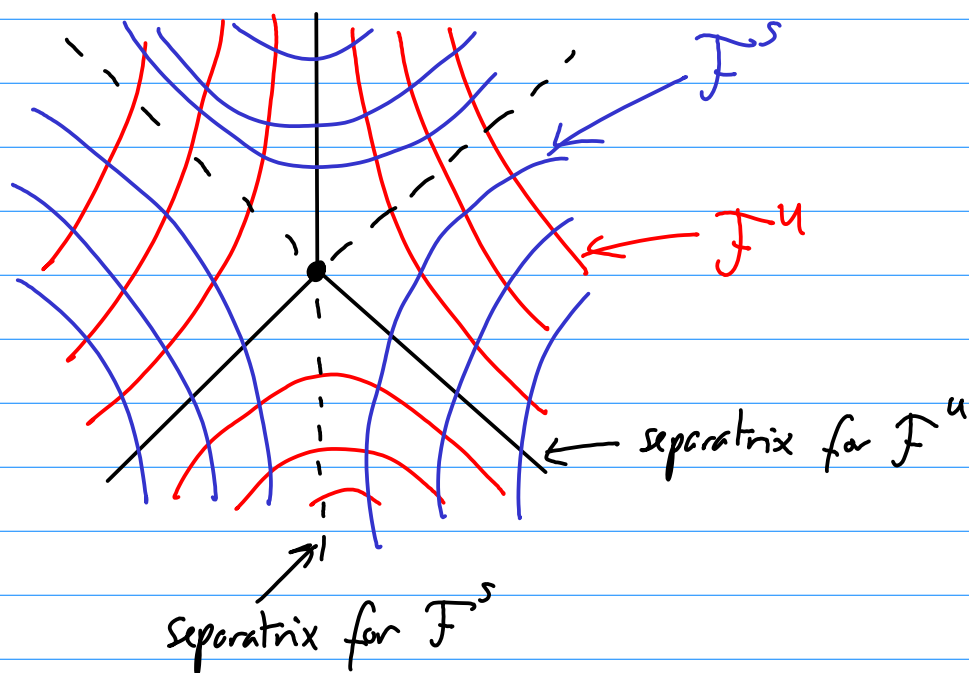
In the neighbourhood of a singularity, each  $\mathcal{F}$  consists of  $p$  separatrices, or prongs:



A 2-pronged singularity is special: it is a regular point, so it really isn't a singularity at all:



Near a singularity,  $F^u$  and  $F^s$  remain transverse, except at the singularity point itself:

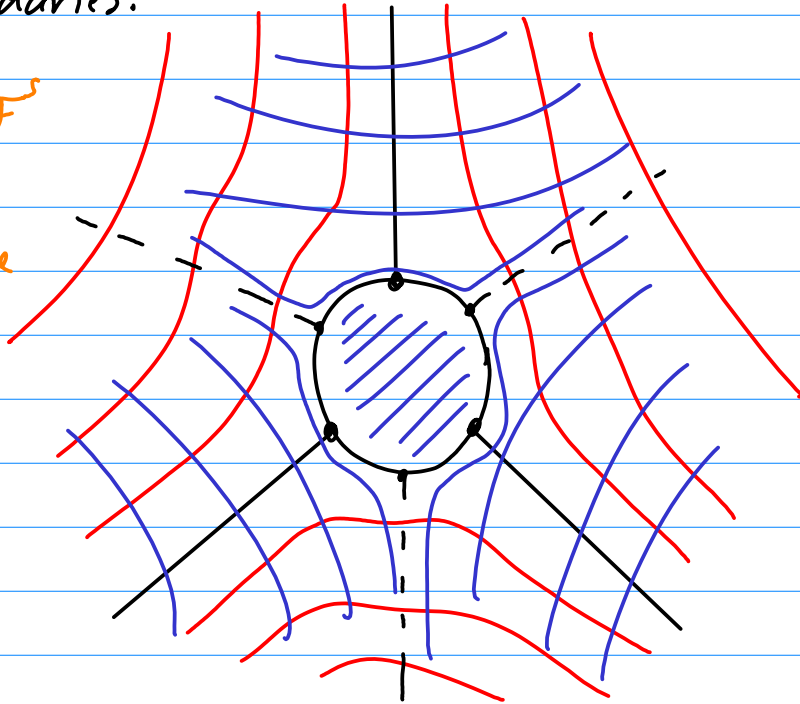


The separatrices are semi-infinite leaves of the foliation.

$F^u$  and  $F^s$  share singularities and they have the same number of prongs there. (Exception: singularities at boundaries — below).

## Near boundaries:

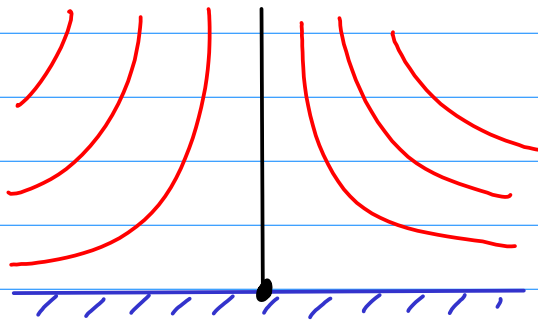
- Singularities of  $F^u$  and  $F^s$  do not coincide on boundaries, though some number of sing.
- Each boundary has at least one sing. for a pA.
- Boundary singularities are 3-prongs.



This is called the "standard model" of pA's (Fried - See Boyland '99)

Boundaries consist of leaves of the foliations, separated by singularities. The leaves of  $F^u$  and  $F^s$  are allowed to coincide at the boundaries. This is also the only place where singularities of  $F^u$  and  $F^s$  can have different number of prongs.

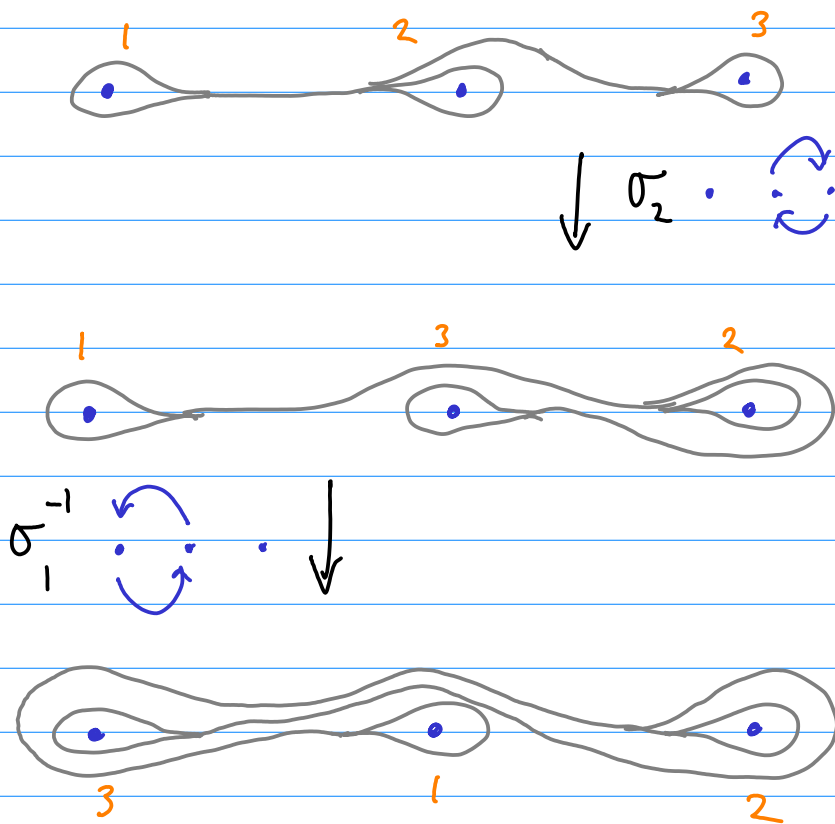
The separatrices of  $F^u$  attached to the boundary will hold special significance:



## Lecture 25: Computing with train tracks

Train tracks are akin to the foliation being crushed onto a one-dimensional graph. It is a branched manifold.

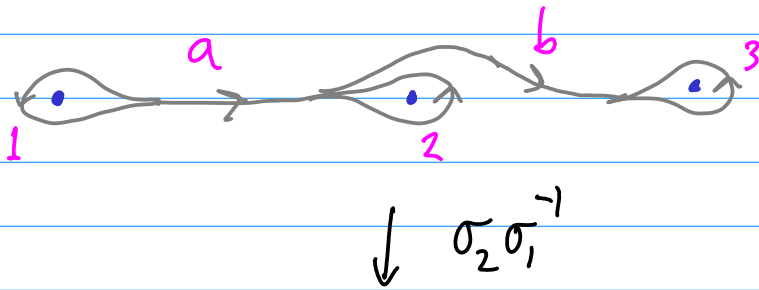
The train track must satisfy a similar invariance property to the foliation:



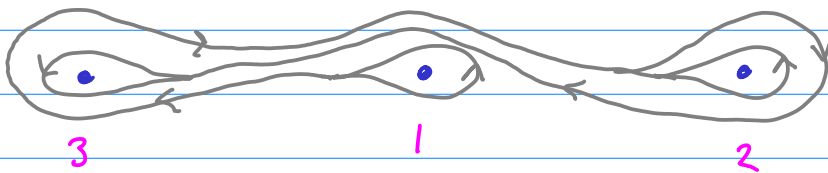
Notice how the final picture "looks" like the initial branched manifold.

We can use this to calculate  $\lambda$ . To do this, we label and orient the edges of the train track.





Label "main" edges by letters, and "loops" by numbers.



Now, much like Markov boxes, described the transformed edges in terms of edge paths, that is, a sequence of original edges.


Get the train track map:

$$\begin{array}{l}
 a \mapsto \bar{a} \bar{1} a b \\
 b \mapsto \bar{3} \bar{b} \bar{a} \\
 1 \mapsto 2 \\
 2 \mapsto 3 \\
 3 \mapsto 1
 \end{array}
 \quad
 \left.
 \begin{array}{l}
 \bar{a} \text{ means } a^{-1}, \text{ with respect to direction.} \\
 \\
 \\
 \\
 \end{array}
 \right\}
 \begin{array}{l}
 \text{The loops are} \\
 \text{simply permuted}
 \end{array}$$

If we find we cannot write the images of the edges in terms of original edge paths, we have the wrong train track.

Example:



Cannot be written in terms of 

We say that the train track supports the pA.

How to get  $\lambda$ : we Abelianize: only count the occurrences of each edge in the map:

$$\begin{pmatrix} a \\ b \\ 1 \\ 2 \\ 3 \end{pmatrix} \mapsto \begin{pmatrix} 2 & 1 & | & 1 & 0 & 0 \\ 1 & 1 & | & 0 & 0 & 1 \\ \hline 0 & 0 & | & 0 & 1 & 0 \\ 0 & 0 & | & 0 & 0 & 1 \\ 0 & 0 & | & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ 1 \\ 2 \\ 3 \end{pmatrix}$$

The spectral radius of the matrix is the spectral radius of  $\begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$ .

permutation matrix  
 $\rightarrow |\text{eigenvalues}| = 1$

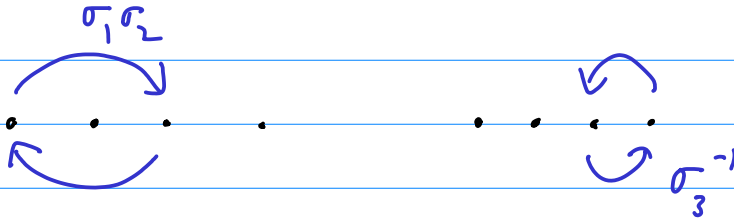
Since loops map only to loops, the transition matrix always has this block triangular form.

More generally, things are of course not so simple. Here are the steps:

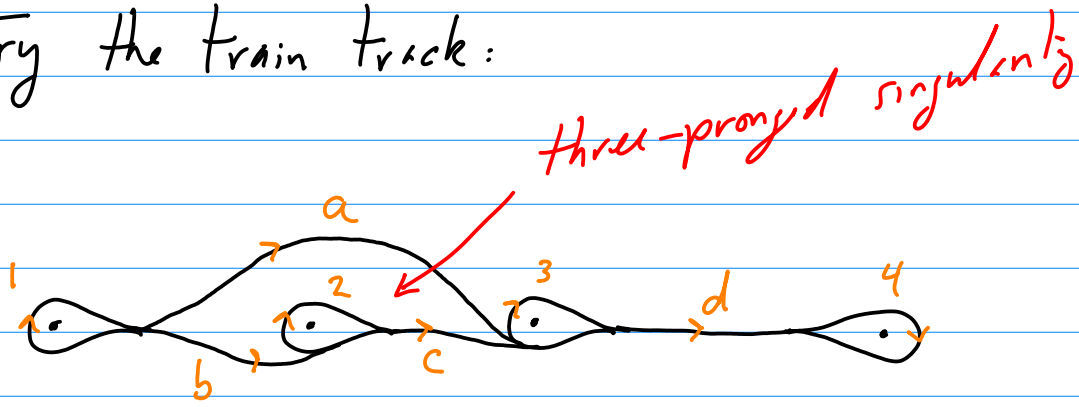
- 1) Find an invariant train track for the diffeo.
- 2) Verify that the TT map is efficient (no cancellations)
- 3) Verify that the transition matrix (the sub-block of main edges) is irreducible.

We will look at more complicated examples to illustrate these steps.

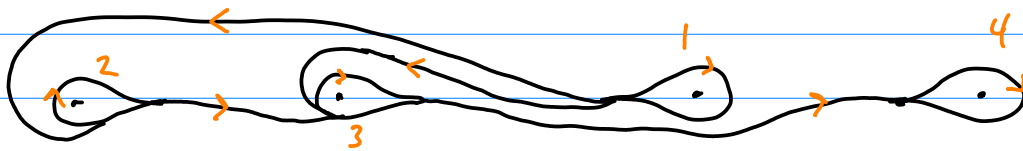
Let's do a more complicated example, for the braid  $\sigma_1 \sigma_2 \sigma_3^{-1}$ :

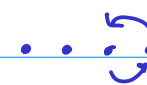


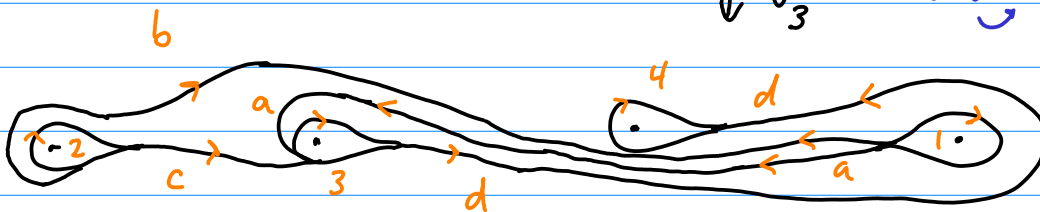
Try the train track:



$\downarrow \sigma_1 \sigma_2$   Skip over the 2 & 3.



$\downarrow \sigma_3^{-1}$  



Train track map:

$$a \mapsto \bar{d} \bar{c} \bar{2}$$

$$b \mapsto \bar{d} \bar{a} \bar{1}$$

$$c \mapsto b$$

$$d \mapsto c d \bar{4} \bar{d}$$

$$1 \mapsto 4$$

$$2 \mapsto 1$$

$$3 \mapsto 2$$

$$4 \mapsto 3$$

Incidence matrix:

$$M = \begin{pmatrix} a & b & c & d \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix} \begin{matrix} f(a) \\ f(b) \\ f(c) \\ f(d) \end{matrix}$$

Think of this as carrying the transformed measure!

Can we tell from the incidence matrix if this is a pseudo-Anosov?

We can, as long as the train track map is efficient (later).

$$M^2 = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 0 & 0 & 2 & 3 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 2 & 4 \end{pmatrix}, \quad M^3 = \begin{pmatrix} 1 & 1 & 2 & 5 \\ 0 & 2 & 3 & 6 \\ 0 & 0 & 2 & 3 \\ 1 & 2 & 4 & 9 \end{pmatrix}, \quad M^4 = \begin{pmatrix} 1 & 2 & 6 & 12 \\ 2 & 3 & 6 & 14 \\ \boxed{0} & 2 & 3 & 6 \\ 2 & 4 & 10 & 21 \end{pmatrix}$$

$$M^5 = \begin{pmatrix} 2 & 6 & 13 & 27 \\ 3 & 6 & 16 & 33 \\ 2 & 3 & 6 & 14 \\ 4 & 10 & 23 & 48 \end{pmatrix}$$

No more zeros!

A matrix  $A$  is reducible if there exists a permutation matrix  $P$  such that  $P^T A P$  is block-triangular. An irreducible matrix is not reducible. (d'uh)

Equivalently:  $A$  is irreducible if,  $\forall i, j$ , there exists  $k$  such that  $(A^k)_{ij} > 0$ .

Hence, the matrix  $M$  above is irreducible.

Perron-Frobenius theorem: Let  $A = (a_{ij})$  be a real  $n \times n$  matrix with  $a_{ij} \geq 0$ . Then:

1. The largest eigenvalue of  $A$  is the spectral radius,  $\lambda$ .  
(i.e., the spectral radius is real)

2. The corresponding eigenvector has nonnegative entries.

$$3. \min_i \sum_j a_{ij} \leq \lambda \leq \max_i \sum_j a_{ij}$$

[The theorem is slightly different for  $a_{ij} > 0$ . Then the largest eigenvalue is nondegenerate, and the eigenvector has entries  $> 0$ .]

3. implies that if we have an irreducible matrix with  $\lambda = 1$ , then it must be a permutation matrix. (apply 3 to  $A^n$ , for  $n$  large enough.)

For the matrix  $M$  above,  $\lambda \approx 2.2966 > 1$ , so it is indeed pseudo-Anosov.

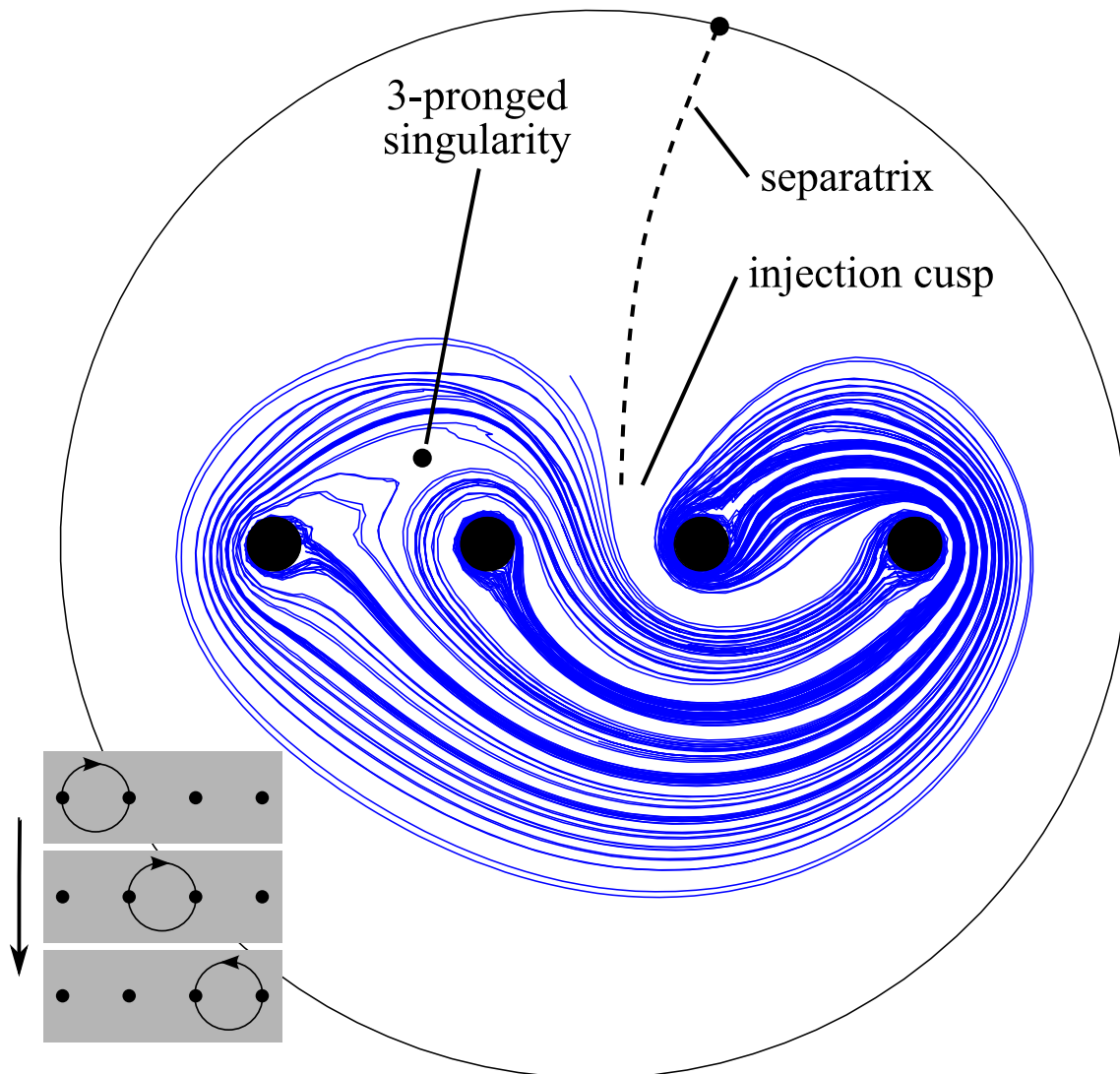
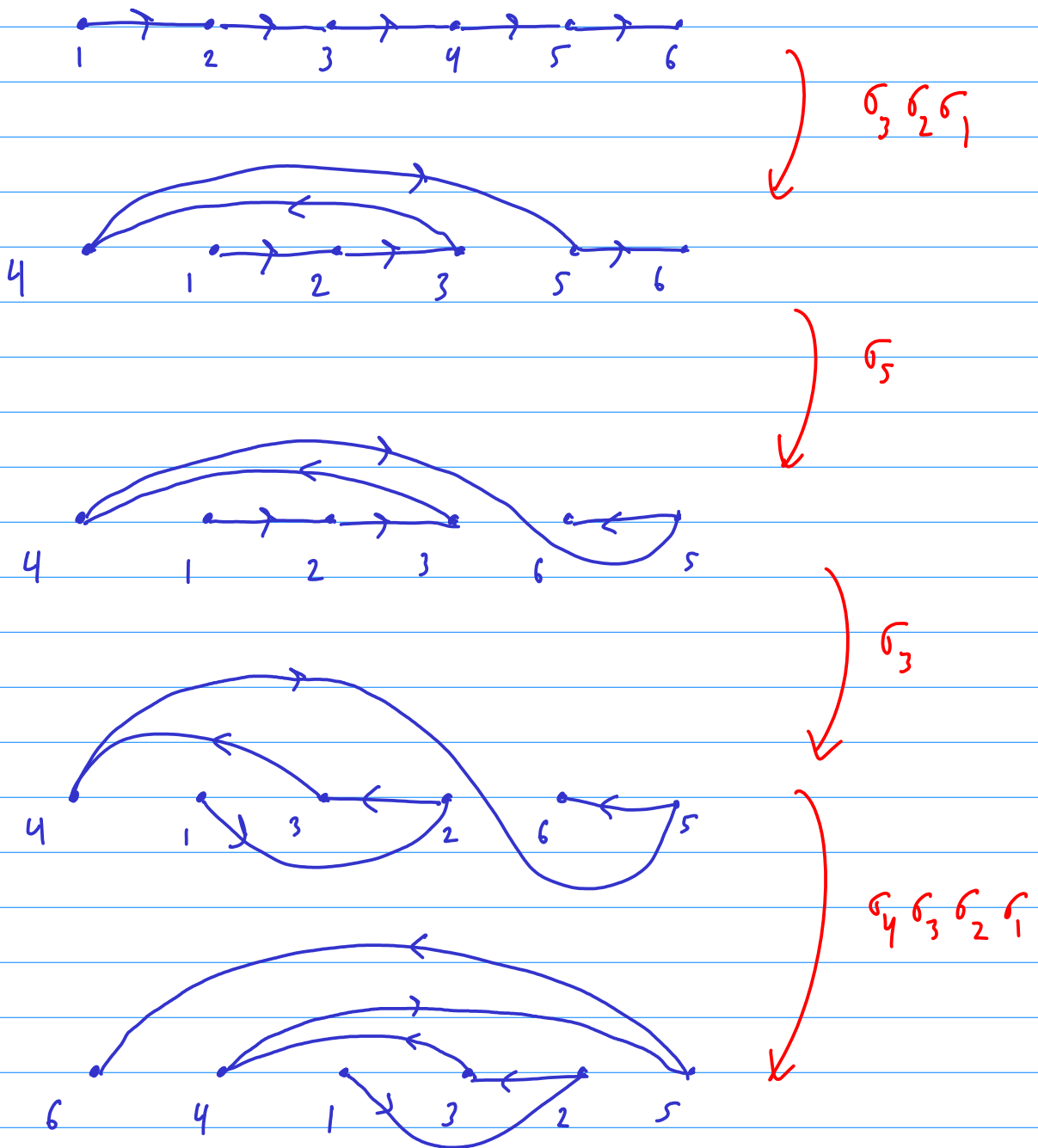


Figure 1: A fluid stirred with rod motions corresponding to the braid  $\sigma_1\sigma_2\sigma_3^{-1}$  (see J.-L. THIFFEAULT, M. D. FINN, E. GOUILLART, AND T. HALL, *Topology of chaotic mixing patterns*, *Chaos*, 18 (2008), p. 033123).

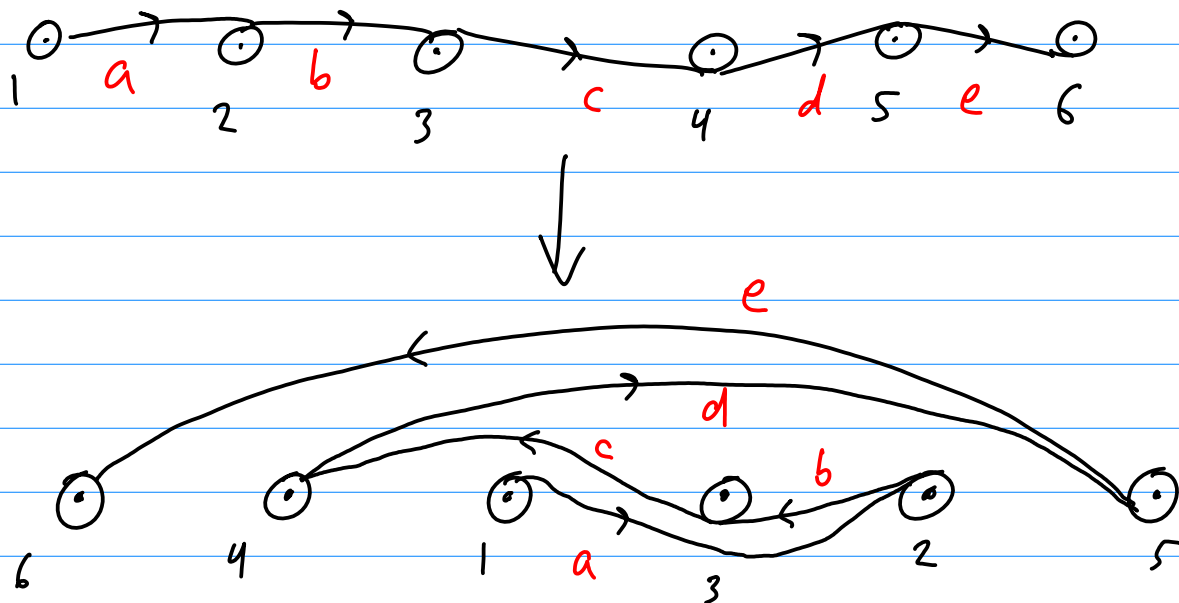
# Lecture 26: The Bestvina-Handel algorithm (from Toby Hall's notes)

Consider the braid:  $\sigma_3 \sigma_2 \sigma_1 \sigma_5 \sigma_3 \sigma_4 \sigma_3 \sigma_2 \sigma_1$   
( $\sigma$  defined counterclockwise for today)



all  
⊙

Redo this with peripheral edges:

Graph map:  $g: G \rightarrow G$ 

$$g(a) = cd$$

$$g(1) = 3$$

$$g(b) = \bar{d}$$

$$g(2) = 5$$

$$g(c) = \bar{c}\bar{b}$$

$$g(3) = 4$$

$$g(d) = bc4de$$

$$g(4) = 2$$

$$g(e) = \bar{e}\bar{d}\bar{4}\bar{c}\bar{b}\bar{a}$$

$$g(5) = 6$$

$$g(6) = 1$$

Note  $g(\bar{a}) = \bar{d}\bar{c}$ , etc.

No cancellations



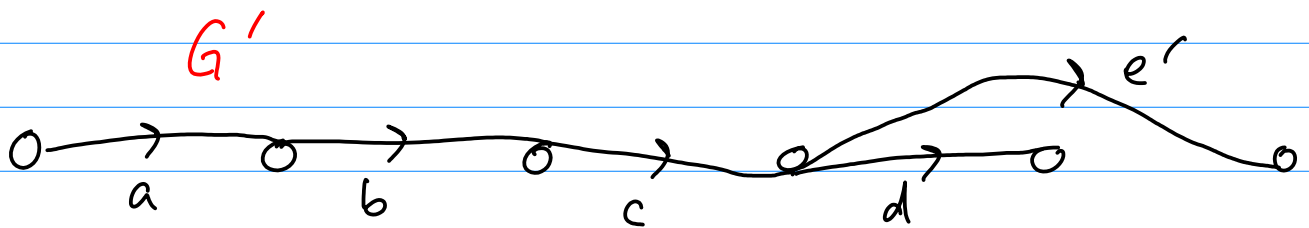
Now consider

$$\begin{aligned}
 g^2(d) &= g(bc4de) \\
 &= (\bar{d})(\bar{c}\bar{b})(2)(bc4de)(\bar{e}\bar{d}\bar{4}\bar{c}\bar{b}\bar{a}) \\
 &= \bar{d}\bar{c}\bar{b}2\bar{a}
 \end{aligned}$$

↓ cancellation!

Source of cancellation:  $g(d)$  contains  $de$ ,  
 and  $g(\bar{d})$  and  $g(e)$  start in the same way.  
( $\bar{e}\bar{d}\bar{4}\bar{c}\bar{b}$ )

To eliminate, identify  $\bar{d}$  with the initial  
 segment of  $e$  which has image  $\bar{e}\bar{d}\bar{4}\bar{c}\bar{b}$ .



$$e = \bar{d}e' \iff e' = de$$

This is called folding two edges

Let's see what this achieved:

$$g': G' \rightarrow G'$$



$$g''(a) = cd = c\bar{c}d' = d'$$

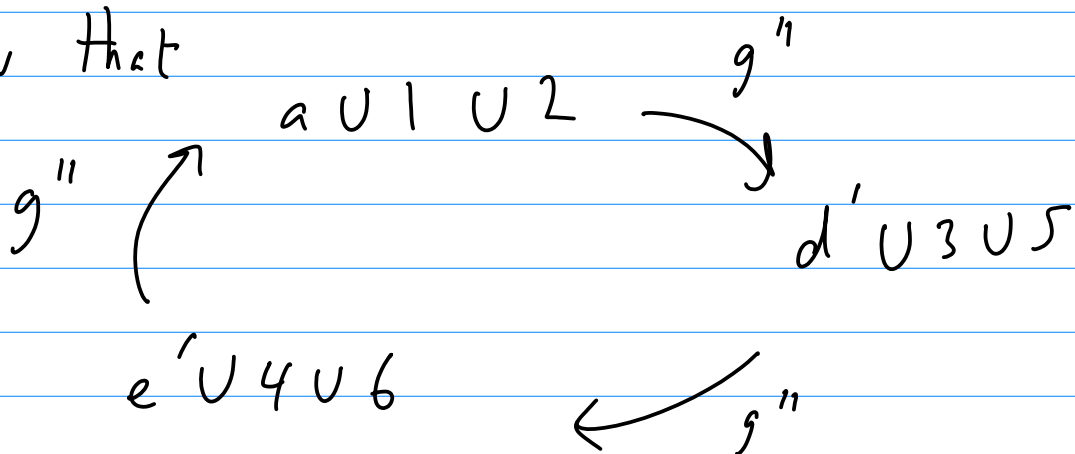
$$g''(b) = \bar{d} = \bar{d}'c$$

$$g''(c) = \bar{c}\bar{b}$$

$$g''(d') = 4e'$$

$$g''(e') = \bar{a}$$

Note now that



Check:

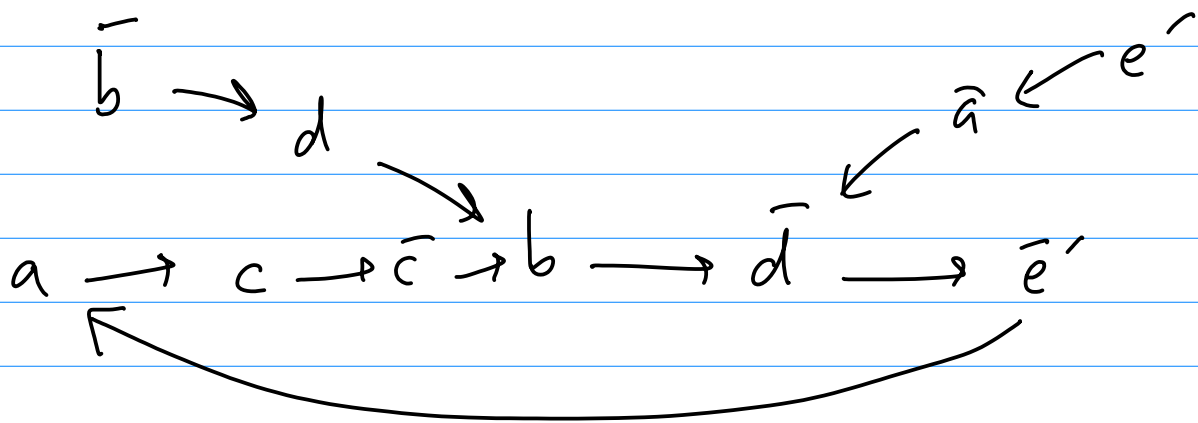
$$g''\{a, 1, 2\} = \{d', 3, 5\}$$

$$g''\{d', 3, 5\} = \{4e', 4, 6\}$$

$$g''\{e', 4, 6\} = \{\bar{a}, 2, 1\}$$



$$\begin{array}{lll}
 g'(a) = cd & Dg'(a) = c & Dg'(\bar{a}) = \bar{d} \\
 g'(b) = \bar{d} & Dg'(b) = \bar{d} & Dg'(\bar{b}) = d \\
 g'(c) = \bar{c}\bar{b} & Dg'(c) = \bar{c} & Dg'(\bar{c}) = b \\
 g'(d) = bc4e' & Dg'(d) = b & Dg'(\bar{d}) = \bar{e}' \\
 g'(e') = \bar{a} & Dg'(e') = \bar{a} & Dg'(\bar{e}') = a
 \end{array}$$



Two elements  $E_1$  and  $E_2$  are equivalent if

$$(Dg)^k(E_1) = (Dg)^k(E_2) \text{ for some } k \geq 0.$$

Equivalence classes are called gates.

Here the gates with  $> 1$  element are

$$\{d, \bar{c}, e'\}, \{\bar{b}, c\}, \{\bar{a}, b\}$$

Lemma: no cancellations in  $g^h(e)$   
 if  $g(e) = E_1 \cdots E_r$  is such that

$\bar{E}_i$  and  $E_{i+1}$  ( $i < r$ ) are in different gates.  
 (easy)

Hence, we construct a list of bad words  
 from the gates:

$\bar{d}\bar{c}$	$bc$	$ab$
$\bar{d}e'$	$\bar{c}\bar{b}$	$\bar{b}\bar{a}$
$cd$		
$\bar{c}e'$		
$e'd$		
$\bar{e}'\bar{c}$		

and check that none of these occurs in the map  
 (here  $cd$ ,  $bc$ , and  $\bar{c}\bar{b}$  do)

$$\begin{array}{lll}
 g''(a) = d' & Dg''(a) = d' & Dg''(\bar{a}) = \bar{d}' \\
 g''(b) = \bar{d}'c & Dg''(b) = \bar{d}' & Dg''(\bar{b}) = \bar{c} \\
 g''(c) = \bar{c}\bar{b} & Dg''(c) = \bar{c} & Dg''(\bar{c}) = b \\
 g''(d') = 4e' & Dg''(d') = 4 & Dg''(\bar{d}') = \bar{e}' \\
 g''(e') = \bar{a} & Dg''(e') = \bar{a} & Dg''(\bar{e}') = a
 \end{array}$$

$$\begin{array}{c}
 e' \rightarrow \bar{a} \\
 \downarrow \\
 c \rightarrow \bar{c} \rightarrow b \rightarrow \bar{d}' \rightarrow \bar{e}' \rightarrow a \rightarrow d' \rightarrow 4
 \end{array}$$

Nontrivial gates are  $\{\bar{c}, e'\}$ ,  $\{\bar{a}, b\}$

bad words:  $ce'$ ,  $\bar{e}'\bar{c}$ ,  $ab$ ,  $\bar{b}\bar{a}$

None of these appear above, so the map  $g''$  is efficient:

## Braids of entangled particle trajectories

Jean-Luc Thiffeault<sup>a)</sup>

*Department of Mathematics, University of Wisconsin, Madison, Wisconsin 53706, USA*

(Received 25 July 2009; accepted 16 October 2009; published online 5 January 2010)

In many applications, the two-dimensional trajectories of fluid particles are available, but little is known about the underlying flow. Oceanic floats are a clear example. To extract quantitative information from such data, one can measure single-particle dispersion coefficients, but this only uses one trajectory at a time, so much of the information on relative motion is lost. In some circumstances the trajectories happen to remain close long enough to measure finite-time Lyapunov exponents, but this is rare. We propose to use tools from braid theory and the topology of surface mappings to approximate the topological entropy of the underlying flow. The procedure uses all the trajectory data and is inherently global. The topological entropy is a measure of the entanglement of the trajectories, and converges to zero if they are not entangled in a complex manner (for instance, if the trajectories are all in a large vortex). We illustrate the techniques on some simple dynamical systems and on float data from the Labrador Sea. The method could eventually be used to identify Lagrangian coherent structures present in the flow. © 2010 American Institute of Physics. [doi:10.1063/1.3262494]

**Consider particles floating on top of a fluid. We can follow their trajectories, either with a camera or by computer simulation. If we then plot their position in a three-dimensional graph, with time as the vertical coordinate, we get a “spaghetti plot,” which contains information about how entangled the trajectories are. We discuss how to measure the level of entanglements in terms of topological entropy, and the interpretation of the results. This provides a straightforward method of estimating the level of chaos present in a system. This approach could also be used to determine if some trajectories remain together for a long time, and are thus part of a Lagrangian coherent structure.**

propose to examine the “braid” defined by the trajectories and to measure its degree of entanglement. (All these terms will be defined more precisely.) The number we get out of this is called the braid’s *topological entropy*. Figure 1(b) shows a measure of the entanglement of the ten floats as a function of time, with an exponential fit: The growth rate is the topological entropy. (For longer times, the curve levels off because floats start leaving the Labrador Sea, which is not really a closed system.) Much like a Lyapunov exponent, the topological entropy gives us a characteristic time for the entanglement of the floats in the Labrador Sea, here about  $1/0.02 \approx 50$  days.

### I. INTRODUCTION

#### A. Floats in the ocean: An example

Figure 1(a) shows the trajectories of ten floats released in the Labrador Sea, for a period of a few months. The principal reason to release such floats is the data they measure and transmit back—temperature, salinity, pressure, etc. However the actual trajectories of the floats are also important since they tell us something about large-scale transport in the ocean, a crucial component in understanding global circulation. From a single float, one can deduce the single-particle dispersion coefficient, a crude measure of how quickly a float wanders away from its release point. However, it is better to measure quantities that involve several floats.<sup>2</sup> For instance, if floats happen to start near each other, then we can see how quickly they separate and measure finite-time Lyapunov exponents,<sup>3</sup> which are linked to chaotic advection.<sup>4</sup> However if the floats are nowhere near each other, then a more global quantity is needed. In this paper we

#### B. Topology and trajectories

Since the original paper of Boyland *et al.*,<sup>5</sup> topological techniques in fluid dynamics have been applied to free point vortices,<sup>6</sup> fixed blinking vortex,<sup>7,8</sup> rod stirring devices,<sup>9–13</sup> and spatially periodic systems.<sup>14,15</sup> (See Ref. 16 for a brief review.) More recently, the emphasis has shifted to locating periodic orbits that play an important role in stirring<sup>9,12</sup>—so-called ghost rods—and even to the manufacture of such orbits.<sup>17</sup> Most of these authors study periodic motions of rods or particle orbits. For many practical applications, however, periodic motion is not directly observable since most such orbits are highly unstable. Hence, some authors examined *random braids*<sup>7,14,16,18,19</sup> composed of arbitrary chaotic trajectories. (There is also related literature from the knot theory perspective—see for example Ref. 20.)

The goal of the present paper is to give concrete techniques that can be used to obtain topological information from particle trajectories. The mathematical details are glossed over: The emphasis is on usability. Implementation details are discussed, and some sample MATLAB programs are presented in Appendix A. The hope is that this will make

<sup>a)</sup>Electronic mail: jeanluc@math.wisc.edu.



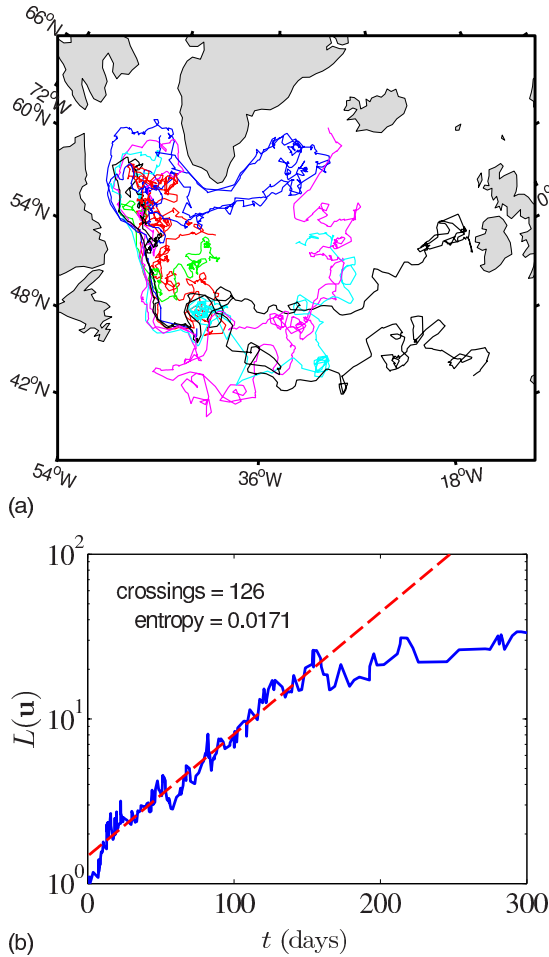


FIG. 1. (Color online) (a) Ten floats from Davis' Labrador Sea data (Ref. 1). (b) The growth of  $L(u)$  for the ten floats, with a fit of the topological entropy. For longer times, the floats leave the Labrador Sea and  $L(u)$  becomes constant. The details of the analysis are presented in Sec. IV.

these techniques more accessible to those with little or no background in braid theory and topology.

The principal measurement we extract from a braid is its topological entropy. This entropy is closely related to the traditional Lyapunov exponent, except that being a topological quantity it is not sensitive to the size of the sets on which chaos is occurring. This is both a weakness and a strength: It does not tell us everything we might like to know, but on the other hand the topological entropy is easy to compute from crude data. This is in contrast to Lyapunov exponents, which require at the very least detailed knowledge of particle trajectories that start close together, and at best the velocity field and its gradient. When dealing with, for example, data from oceanic floats (as we will later in this paper), being able to compute a Lyapunov exponent is a rarity.

There is also a philosophical point that bears some discussion. The viewpoint of the present paper is that given particle trajectory data, a useful thing to quantify is how “entangled” the particle trajectories are. This can be done from the particle data directly, without worrying about the underlying flow. By contrast, Lyapunov exponents are defined locally and are sensitive to the *smooth structure* of the flow. It is exactly the (presumed) smooth nature of the flow

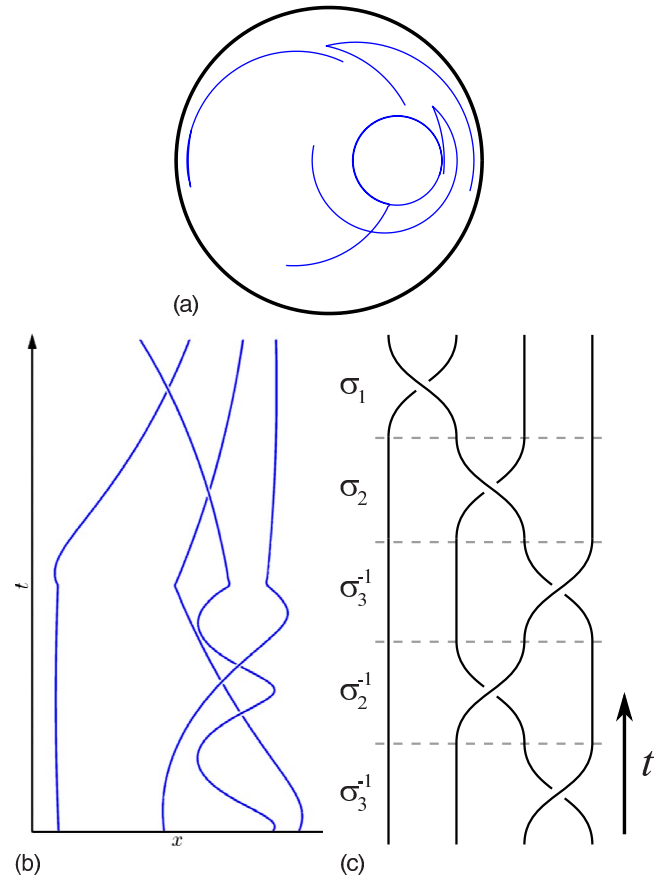


FIG. 2. (Color online) (a) The orbits of four particles in a circular two-dimensional domain. (b) The same orbits, lifted to a space-time diagram in three dimensions, with time flowing from bottom to top. (c) The standard braid diagram corresponding to (b).

that connects local information to a global quantity such as the Lyapunov exponent, but one is left wondering why we should care about the local picture at all in practical situations. The topological viewpoint presented here is an attempt to sidestep this and focus directly on global information.

We begin in Sec. II by a short introduction to braid theory, surface dynamics, and their connection to dynamical systems. In Sec. II B we show how to extract braids from particle trajectory data. Section III is devoted to topological entropy: In Sec. III A we discuss its connection to flows and in Sec. III B we show how to measure it from a braid (for a braid corresponding to periodic orbits). In Sec. IV we introduce random braids and again show how to measure entropies. As an application, we calculate the entropy for floats in the Labrador Sea, as presented in Fig. 1. We offer some concluding remarks in Sec. V.

## II. BRAIDS

### A. Physical and algebraic braids

First we describe intuitively how braids arise. Figure 2(a) shows the orbits of four particles in a circular two-dimensional domain. The particles might be fluid elements, solutions of ordinary differential equations, or physical particles at the surface of a fluid. Figure 2(b) shows the “world

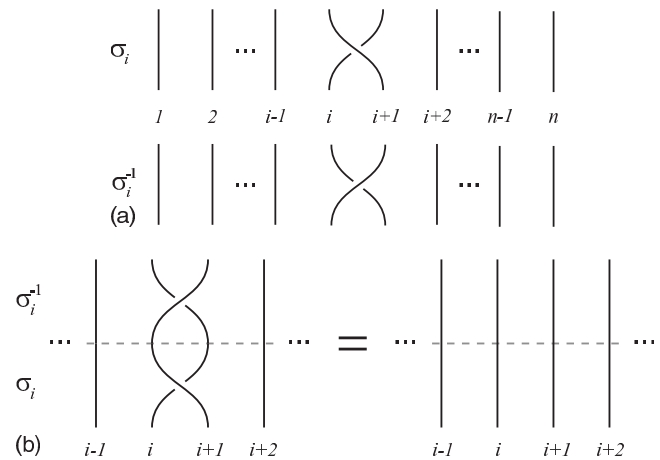


FIG. 3. (a) Braid group generator  $\sigma_i$ , corresponding to the clockwise interchange of the string at the  $i$ th and  $(i+1)$ th position, counted from left to right. Its inverse  $\sigma_i^{-1}$  involves their counterclockwise exchange. (b) The concatenation of  $\sigma_i$  and  $\sigma_i^{-1}$  gives the identity braid.

line” of the same orbits: They are plotted in a three-dimensional graph, with time flowing vertically upward. The diagram in Fig. 2(b) depicts a *physical braid* made up of four strands. No strand can go through another strand as a consequence of the deterministic motion of the particles (they never occupy the same point at the same time). Moreover, the mathematical definition of a braid requires that strands cannot “loop back:” Here this simply means that the particles cannot travel back in time. We will say that two braids are equivalent if they can be deformed into each other with no strand crossing other strands or boundaries. Throughout this paper, we will be interested in characterizing the level of “entanglement” of trajectories.

Since we can move the strands, it is convenient to draw braids in a normalized form, as shown in Fig. 2(c) for the braid in Fig. 2(b). Such a picture is called a *braid diagram*. The important thing is that we record when crossings occur, and which particle was behind and which was in front. It matters little how we define “behind” as long as we are consistent (see Sec. II B for practical considerations). In Fig. 2(c) the horizontal dashed lines also suggest that we can divide the braid into a sequence of elementary crossings, known as *generators*. Figure 3(a) shows the definition of  $\sigma_i$ , which denotes the clockwise interchange of the  $i$ th and  $(i+1)$ th strands, keeping all other strands fixed. Note that the index  $i$  is the position of the strand from left to right, *not* a label for the particular strand. For  $n$  strands, we have  $n-1$  distinct generators.

Figure 3(a) also shows the counterclockwise interchange of two strands, denoted by the operation  $\sigma_i^{-1}$ . The justification of the “inverse” notation is evident in Fig. 3(b): If we concatenate  $\sigma_i$  and  $\sigma_i^{-1}$ , then after pulling tight on the strands, we find that they are disentangled. We call the braid on the right in Fig. 3(b) the *identity braid*. In fact, the set of all braids on a given number  $n$  of strands forms a *group* in the mathematical sense: The group operation is given by concatenation of strands; the inverse by reversing the order and direction of crossings; the identity is as described above; and it is clear that concatenation is associative. This group is

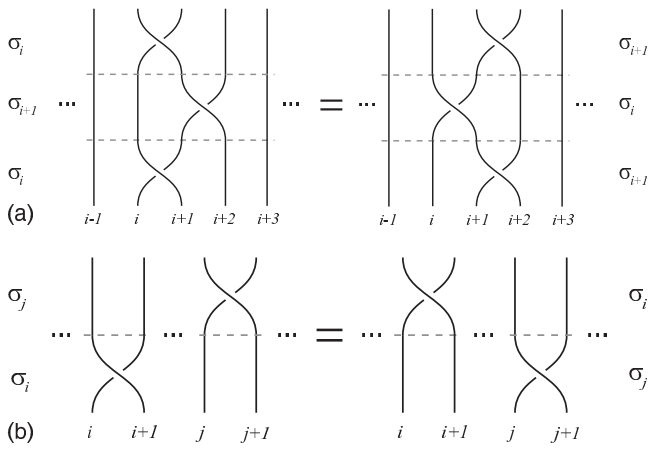


FIG. 4. Braid group relations [see Eq. (1)]: (a) relation for three adjacent strands; (b) commutation relation for generators that do not share a strand.

called  $B_n$ , the braid group on  $n$  strands, also known as the Artin braid group.

The braid group  $B_n$  is generated by the set  $\{\sigma_1, \dots, \sigma_{n-1}\}$ : This means that any braid in  $B_n$  can be written as a product (concatenation) of  $\sigma_i$ 's and their inverses. The braid group is *finitely generated* even though it is itself infinite: Only a finite number of generators give the whole group. To see that the braid group contains an infinite number of braids, simply consider  $\sigma_1^k$ , for  $k$  an arbitrary integer: No matter how large  $k$  gets, we always get a new braid out of this, consisting of increasingly twisted first and second strands.

We have now passed from physical braids, as depicted in Fig. 2(b), to *algebraic braids*. The algebraic braid corresponding to Fig. 2(c) is  $\sigma_3^{-1}\sigma_2^{-1}\sigma_3^{-1}\sigma_2\sigma_1$ , where we read generators from left to right in time (beware: conventions differ). In essence, an algebraic braid is simply a sequence of generators, which may or may not come from a physical braid. How can we guarantee that physical braids and algebraic braids describe the same group? We need to be mindful of *relations* among the generators that arise because of physical constraints. For example, Fig. 4(a) shows a relation among adjacent triplets of strands. Staring at the picture long enough and allowing for the deformation of strands without crossing, the reader can perhaps see that braids in Fig. 4(a) are indeed equal. Hence, the algebraic sequence  $\sigma_i\sigma_{i+1}\sigma_i$  must be equal to  $\sigma_{i+1}\sigma_i\sigma_{i+1}$  if the generators are to correspond to physical braids. Another, more intuitive relation is shown in Fig. 4(b): Generators commute if they do not share a strand. In summary, we have the relations

$$\sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j \quad \text{if } |i-j|=1, \tag{1a}$$

$$\sigma_i\sigma_j = \sigma_j\sigma_i \quad \text{if } |i-j|>1, \tag{1b}$$

among the generators. Artin<sup>21</sup> proved the surprising fact that there are no other relations satisfied by the generators  $\sigma_i$  except for those that can be derived from Eq. (1) by basic group operations (multiplication, inversion, etc.). The generators  $\{\sigma_1, \dots, \sigma_{n-1}\}$  together with relations (1) define the *algebraic braid group*, which we also denote  $B_n$ . With these relations, the groups of physical and algebraic braids are isomorphic.

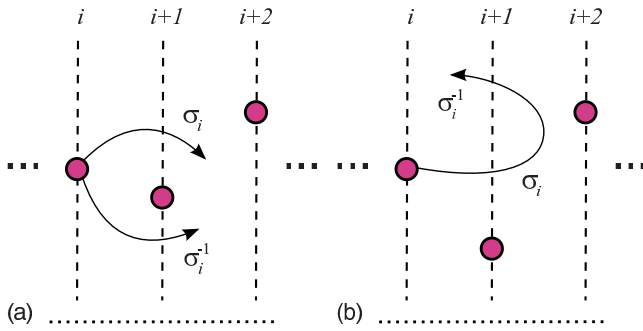


FIG. 5. (Color online) Detecting crossings: (a) Two possible particle paths that are associated with different braid group generators. (b) Two crossings that yield no net braiding. The projection line used to detect crossings is shown dotted at the bottom, and the perpendicular lines used to determine the braid generator are shown dashed.

A consequence of relations (1) is that it may not be immediately obvious that two algebraic braids are equal. For instance, the braids  $\sigma_1\sigma_2$  and  $\sigma_2\sigma_2\sigma_1\sigma_2\sigma_1^{-1}\sigma_1^{-1}$  are equal since  $\sigma_1\sigma_2 = (\sigma_1\sigma_2\sigma_1)\sigma_1^{-1} = (\sigma_2\sigma_1\sigma_2)\sigma_1^{-1} = \sigma_2(\sigma_1\sigma_2\sigma_1)\sigma_1^{-1}\sigma_1^{-1} = \sigma_2\sigma_2\sigma_1\sigma_2\sigma_1^{-1}\sigma_1^{-1}$ . This braid equality problem has seen many refinements: The original solution of Artin<sup>21</sup> has computational complexity exponential in the number of generators, but modern techniques can determine equality in a time quadratic in the braid length.<sup>22–24</sup>

## B. Extracting the braid from a flow

The first step in obtaining useful topological information from particle trajectories is to compute their associated braid, essentially going from the physical picture in Fig. 2(b) to the algebraic picture in Fig. 2(c). A simple method to do this was originally described in Ref. 7 but is also implicit in earlier work such as Refs. 18 and 25 (see also Ref. 26 for a related technique).

We start with trajectory information for  $n$  particles over some time. We first project the position of the particles onto any fixed *projection line* (which we choose to be the horizontal axis), and label the particles by  $i=1, 2, \dots, n$  in increasing order of their projection. A crossing occurs whenever two particles interchange their positions on the projection line. A crossing can occur as an “over-” or “underbraid,” which for us means a clockwise or counterclockwise interchange. These interchanges correspond to the braid group generators introduced in Sec. II A.

Assuming that a crossing has occurred between the  $i$ th and  $(i+1)$ th particles, we need to determine if the corresponding braid generator is  $\sigma_i$  or  $\sigma_i^{-1}$ . We look at the projection of the  $i$ th and  $(i+1)$ th particles in the direction perpendicular to the projection line (the vertical axis in our case). If the  $i$ th particle is *above* the  $(i+1)$ th at the time of crossing, then the interchange involves the group generator  $\sigma_i$  (we define above as having a greater value of projection along the perpendicular direction). Conversely, if the  $i$ th particle is *below* the  $(i+1)$ th at the time of crossing, then the interchange involves the group generator  $\sigma_i^{-1}$ . Figure 5(a) depicts these two situations.

The net result is that from the  $n$  particle trajectories we obtain a time-ordered sequence of the generators  $\sigma_i$  and  $\sigma_i^{-1}$ ,

$i=1, \dots, n-1$ . We call this sequence the braid of the trajectories. We also record the times at which crossings occur so each generator in the sequence has a time associated with it.

*Remarks:*

- The method just described might seem to detect spurious crossings if two well-separated particles just happen to interchange position on the projection line several times in a row, as shown in Fig. 5(b). However, this would imply a sequence of  $\sigma_i$  and  $\sigma_i^{-1}$  braid generators since which particle is labeled  $i$  changes at each crossing. When composed together the generators for these crossings cancel.
- We give a simple MATLAB implementation of the method in Appendix A 1. The program GENCROSS detects crossings in trajectory data; it makes an effort to resolve multiple simultaneous crossings (up to triple crossings) but will complain if it gets confused. More sophisticated code can be written that reinterpolates the trajectory as needed to detect crossings.
- When the system has symmetries, such as when several periodic orbits lie on the same line, there are ‘bad’ choices of projection line where it is impossible to resolve the order of crossings since orbits cross at exactly the same time. Displacing the projection line a little cures this.
- If the braid is truly periodic, that is, if all the particles return to their initial configuration, then changing the projection line changes the braid, but only by *conjugation*, which does not affect the entropy<sup>6</sup> (Sec. III).
- If the trajectories are not periodic, then the method does not define a braid in the traditional sense where all the strands return to the same initial configuration. This is inconsequential to our purposes: all that matters is the order along the projection line (see also Ref. 25). The choice of projection line changes the braid beyond simple conjugation, but this only creates an error in a small, finite number of generators, which is not important when considering long braids and does not asymptotically affect the entropy (Sec. III).
- If the braid is generated from chaotic trajectories, then missing a few crossings (due to, say, gaps in the data) is fine as long as the trajectories are long enough.

## III. TOPOLOGICAL ENTROPY

In Sec. II we described how a set of trajectories in a two-dimensional dynamical system can be described as a braid in a three-dimensional space-time diagram. In this section we will describe further how this braid relates to topological information for the underlying flow.

It is worth noting that braids are not always interpreted in terms of trajectories: They arose first and are still studied as independent geometrical and algebraic objects. The reason they take center stage in the present study is through their connection to mappings of surfaces (mapping class groups). The Thurston–Nielsen theory<sup>27–31</sup> classifies mappings of surfaces according to whether they can be “deformed” to each other in a topological sense. Braids provide a convenient way of labeling the *isotopy classes* that result. So even

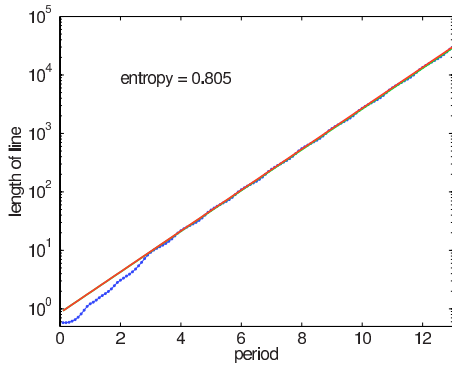


FIG. 6. (Color online) The length of a material line as it is advected by a flow for many periods. The exponential growth rate is very well defined (fitted line), even though there are identical small oscillations at each period.

though we will often speak here of the braid as being the primary object of interest, we are really using techniques that apply to the class of mappings labeled by a braid.

**A. Entropy of a flow**

Ultimately, we want to measure the topological entropy of a system directly from a braid of trajectories. Before we do this, we discuss the meaning of topological entropy of a flow or map. The topological entropy of a dynamical system measures the loss of information under the dynamics. It is closely related to the Lyapunov exponent, which measures the time-asymptotic rate of separation of neighboring trajectories. However it is in some sense a cruder quantity since it does not require a notion of distance. A positive entropy is associated with chaos although it tells us nothing about the size of the chaotic region. The topological entropy is an upper bound on the largest Lyapunov exponent of a flow. The two are equal only for very simple systems where stretching is uniform.

Although there are more fundamental ways to define it, we shall take our working definition of entropy to be the asymptotic growth rate of material lines.<sup>32</sup> It is fairly straightforward to measure this numerically, given a sufficiently accurate velocity field. We simply choose an initial material line and follow it for some time, interpolating new points as the line gets longer. Figure 6 shows such a line for a numerical simulation of a stirred viscous flow. Note how the exponential growth rate is very sharply defined. The topological entropy is the supremum of the growth rate over all such loops, but in practice almost any nontrivial loop (i.e., that spans the domain) will grow exponentially at a rate  $h_{\text{flow}}$ .

In practical applications we often do not have access to an accurate representation of the velocity field. This is where braids come in, as a way of approximating the topological entropy. As we will see in Sec. III B, the braid provides a lower bound on the flow’s topological entropy.

**B. Entropy of a braid**

Figure 7 illustrates how the motion of  $n=3$  point particles can be used to set a lower bound on the topological entropy, defined here as the growth rate of material lines or loops. Here, the point particles undergo a motion described

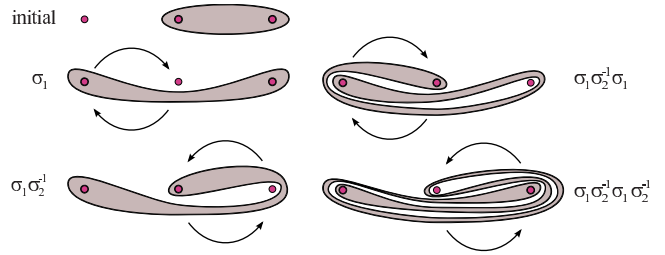


FIG. 7. (Color online) For  $n=3$  particles, a loop is initially wrapped around the second and third particles. The generator  $\sigma_1$  is applied, interchanging clockwise the first and second particles, followed by  $\sigma_2^{-1}$ , which interchanges the second and third particles counterclockwise. The loop is forced to stretch under this application. The final loop on the lower right underwent two applications of the braid  $\sigma_1 \sigma_2^{-1}$ . If we keep applying this braid, the length of the loop will grow exponentially.

by the braid  $\sigma_1 \sigma_2^{-1}$ . Two full periods are shown. Notice that an initial loop that is “caught” on the particles is forced to follow along since determinism implies that it cannot occupy the same point in phase space as the particles. In fact, a straightforward calculation<sup>5</sup> shows that for this braid the total length of the loop must grow exponentially at least at a rate  $\log((3+\sqrt{5})/2)$  per period. We call this rate the topological entropy of the braid,  $h_{\text{braid}}^{(n)}$ , to distinguish it from the true topological entropy of the flow,  $h_{\text{flow}}$ , as defined in Sec. III A. We have

$$h_{\text{flow}} \geq h_{\text{braid}}^{(n)} \tag{2}$$

for any braid obtained from the motion of  $n$  particles in the flow. Typically, the more particles are included in the braid, the closer  $h_{\text{braid}}^{(n)}$  is to  $h_{\text{flow}}$ .<sup>14</sup> Note that  $h_{\text{braid}}^{(1)}$  and  $h_{\text{braid}}^{(2)}$  are always zero.

An essential property of  $h_{\text{braid}}^{(n)}$  is that the growth rate of the loop is independent of specific details: For instance, if the particles are not equally spaced, or if the loop is “tightened” around the particles, then the length will change, but the asymptotic growth rate will not because all these changes amount to an additive constant in the logarithm, which gets divided by a large time.

We are now faced with a task: Given a sequence of generators  $\sigma_i$ , measured in some way or obtained numerically from a flow, what is  $h_{\text{braid}}^{(n)}$ ? The method used in Ref. 7, based on a matrix representation of the braid group, only provides a lower bound on the braid entropy. An accurate and efficient computation has since become a lot simpler due to a new algorithm by Moussafir,<sup>33</sup> who used a set of coordinates to encode a loop. We describe this briefly below; for more details see Refs. 33–35. The reader who is mostly interested in using the method can skip to the end of the section to procedure 1.

The basic idea is simple: Consider the closed loop in Fig. 8, which is wrapped around  $n=5$  particles. The loop does not intersect itself, so in two dimensions the allowable paths it can follow around the particles are far from arbitrary. The amazing fact is that we can reconstruct the entire loop, or at least the way it is threaded around the particles, by counting how many times it intersects the vertical lines in Fig. 8.

In Fig. 9 we give specific labels to the crossing numbers.

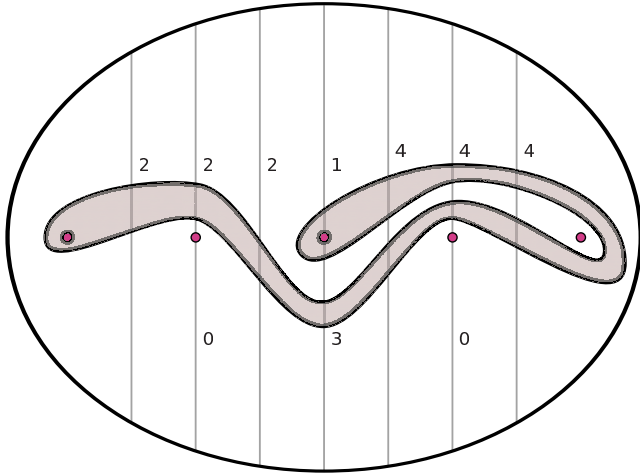


FIG. 8. (Color online) A nonintersecting closed loop wrapped around  $n=5$  particles. Up to trivial deformations, the loop can be reconstructed by counting intersections with the vertical lines. In terms of the crossing numbers defined in Fig. 9, this loop has  $\nu_1=\nu_2=2$ ,  $\nu_3=\nu_4=4$ ,  $\mu_1=2$ ,  $\mu_3=1$ ,  $\mu_5=4$ ,  $\mu_2=\mu_6=0$ ,  $\mu_4=3$ , and Dynnikov coordinate vector  $\mathbf{u}=(-1, 1, -2, 0, -1, 0)$  [see Eq. (4)].

For  $n$  particles,  $\mu_{2i-3}$  (odd index) gives the number of crossings of a loop above the  $i$ th particle, and  $\mu_{2i-2}$  (even index) below the same particle. The number  $\nu_i$  counts the crossings between particle  $i$  and  $i+1$ . We have a total of  $3n-5$  crossing numbers.

This set of crossing numbers (which are all non-negative) can be reduced further: Define

$$a_i = \frac{1}{2}(\mu_{2i} - \mu_{2i-1}), \quad b_i = \frac{1}{2}(\nu_i - \nu_{i+1}) \quad (3)$$

for  $i=1, \dots, n-2$ . The vector of length  $(2n-4)$ ,

$$\mathbf{u} = (a_1, \dots, a_{n-2}, b_1, \dots, b_{n-2}), \quad (4)$$

is called the *Dynnikov coordinates* of a loop. (As far as we know, this specific encoding was originally introduced by Dynnikov,<sup>34</sup> but it is implicit in earlier work of Thurston, Dehn, and others.) The components  $a_i$  and  $b_i$  are signed integers. They can be used to exactly reconstruct the loop,<sup>35</sup> but we shall not need to do this here. This set of coordinates is minimal: It is not possible to achieve the same reconstruction with fewer numbers.

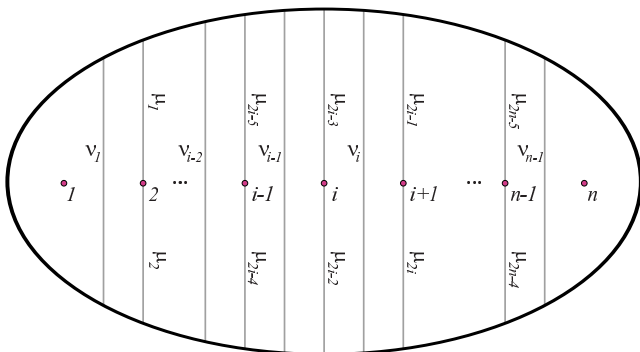


FIG. 9. (Color online) Definition of the crossing numbers  $\mu_i$  and  $\nu_i$ . The  $\mu_i$  for  $i$  odd count crossings above a particle, and below a particle for  $i$  even. The  $\nu_i$  count crossings between particles.

We can also obtain the minimum number of intersections  $L(\mathbf{u})$  of the loop with the horizontal line through the particles,<sup>33</sup>

$$L(\mathbf{u}) = |a_1| + |a_{n-2}| + \sum_{i=1}^{n-3} |a_{i+1} - a_i| + \sum_{i=0}^{n-1} |b_i|, \quad (5)$$

where  $b_0$  and  $b_{n-1}$  can be obtained from the other coordinates as<sup>35</sup>

$$b_0 = - \max_{1 \leq i \leq n-2} \left( |a_i| + b_i^+ + \sum_{j=1}^{i-1} b_j \right), \quad b_{n-1} = -b_0 - \sum_{j=1}^{n-2} b_j. \quad (6)$$

The formula for  $L(\mathbf{u})$  is encoded in the MATLAB function LOOPINTER in Appendix A 2. For example, the loop in Fig. 8 intersects the horizontal axis (the line through all the particles) 12 times. The crucial observation, which will allow a simple computation of  $h_{\text{braid}}^{(n)}$ , is that if the length of the loop grows exponentially, then  $L(\mathbf{u})$  also grows exponentially at the same rate.<sup>27,33</sup>

Now that we have a way of encoding any loop, we need to find how the loop is transformed by a braid. What makes all this work is that there is a very efficient way of doing this: Given a loop encoded by  $\mathbf{u}$  as in Eq. (4), each generator of the braid group  $\sigma_i$  simply transforms these coordinates in a predetermined manner. (Mathematically, this defines an *action* of the braid group on the set of Dynnikov coordinates.) We call these transformations the *update rules* for a generator.

The update rules are straightforward to code on a computer. (See Appendix A 2 for a MATLAB implementation.) To express them succinctly,<sup>36</sup> first define for a quantity  $f$  the operators,

$$f^+ := \max(f, 0), \quad f^- := \min(f, 0). \quad (7)$$

After we define

$$c_{i-1} = a_{i-1} - a_i - b_i^+ + b_{i-1}^-, \quad (8)$$

we can express the update rules for the  $\sigma_i$  acting on  $\mathbf{u} = (a_1, \dots, a_{n-2}, b_1, \dots, b_{n-2})$  as

$$a'_{i-1} = a_{i-1} - b_{i-1}^+ - (b_i^+ + c_{i-1})^+, \quad (9a)$$

$$b'_{i-1} = b_i + c_{i-1}^-, \quad (9b)$$

$$a'_i = a_i - b_i^- - (b_{i-1}^- - c_{i-1})^-, \quad (9c)$$

$$b'_i = b_{i-1} - c_{i-1}^- \quad (9d)$$

for  $i=2, \dots, n-2$ . For this and the following update rules, all the other unlisted components of  $\mathbf{u}$  are unchanged under the action of  $\sigma_i$  or  $\sigma_i^{-1}$ . The leftmost ( $i=1$ ) and rightmost ( $i=n-1$ ) generators require special treatment, having update rules

$$a'_1 = -b_1 + (a_1 + b_1^+)^+, \quad (10a)$$

$$b'_1 = a_1 + b_1^+ \quad (10b)$$

for  $\sigma_1$ , and



$$a'_{n-2} = -b_{n-2} + (a_{n-2} + b_{n-2})^-, \tag{11a}$$

$$b'_{n-2} = a_{n-2} + b_{n-2}^- \tag{11b}$$

for  $\sigma_{n-1}$ .

We need to give separate update rules for the generators  $\sigma_i^{-1}$ . With the definition

$$d_{i-1} = a_{i-1} - a_i + b_i^+ - b_{i-1}^-, \tag{12}$$

the update rules for the  $\sigma_i^{-1}$  are

$$a'_{i-1} = a_{i-1} + b_{i-1}^+ + (b_i^+ - d_{i-1})^+, \tag{13a}$$

$$b'_{i-1} = b_i - d_{i-1}^+, \tag{13b}$$

$$a'_i = a_i + b_i^- + (b_{i-1}^- + d_{i-1})^-, \tag{13c}$$

$$b'_i = b_{i-1} + d_{i-1}^+ \tag{13d}$$

for  $i=2, \dots, n-2$ . We also have

$$a'_1 = b_1 - (b_1^+ - a_1)^+, \tag{14a}$$

$$b'_1 = b_1^+ - a_1 \tag{14b}$$

for  $\sigma_1^{-1}$ , and

$$a'_{n-2} = b_{n-2} - (b_{n-2}^- - a_{n-2})^-, \tag{15a}$$

$$b'_{n-2} = b_{n-2}^- - a_{n-2} \tag{15b}$$

for  $\sigma_{n-1}^{-1}$ .

Update rules of this form are known as *piecewise linear*: once the minima and maxima are resolved, what is left is simply a linear operation. However, the minima and maxima are what keep this from being a simple linear algebra problem and make the braid dynamics so rich.

Here then is a recipe for computing  $h_{\text{braid}}^{(n)}$ , the topological entropy of a braid of  $n$  particle trajectories:

**Procedure 1.** (Entropy of periodic braid)

- (1) Start with an arbitrary initial loop, encoded as a vector  $\mathbf{u}$  [Eq. (4)]; set  $m$  to 0;
- (2) for each generator in the braid, use the appropriate update rule from (8)–(15) to modify  $\mathbf{u}$ ;
- (3) compute the intersection number  $L(\mathbf{u})$  using Eq. (5);
- (4) repeat steps (2) and (3) for all generators in the braid;
- (5) add 1 to  $m$ ; calculate  $h_{\text{braid}}^{(n)} = m^{-1} \log L(\mathbf{u})$ ;
- (6) repeat steps (2)–(5) until  $h_{\text{braid}}^{(n)}$  converges in step (5).

*Remarks:*

- The procedure above assumes that the braid is *periodic*, i.e., is obtained from periodic orbits of the flow. In Sec. IV we will discuss how the method differs for random braids obtained from sampling arbitrary trajectories (which do not necessarily repeat).
- The dimension of  $h_{\text{braid}}^{(n)}$  is inverse time, where the unit of time is the period over which the braid is repeated.
- Even though the discussion so far has described  $\mathbf{u}$  as a vector of integers, the initial condition for  $\mathbf{u}$  in step (1) can in practice be chosen to be a random set of real numbers.

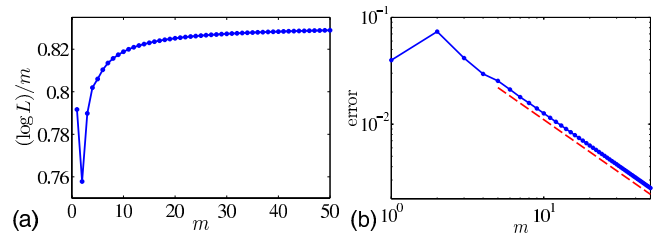


FIG. 10. (Color online) For the braid  $\sigma_3^{-1}\sigma_2^{-1}\sigma_3^{-1}\sigma_2\sigma_1$  in Fig. 2. (a) Entropy estimate  $m^{-1} \log L(\mathbf{u})$  as a function of period  $m$  using procedure 1. (b) Error (deviation from the true entropy  $\approx 0.83144$ ), showing the  $1/m$  convergence (dashed line).

(This is called a *projectivized* version of the coordinates.<sup>35</sup>)

- As is typical of such exponential growth calculation, it is possible that the components of  $\mathbf{u}$  becomes so large that they overflow double-precision arithmetic. In that case standard “renormalization” techniques can be used: Divide  $\mathbf{u}$  by a large constant  $L_{\text{overflow}}$ , but keep track of how many times this division was done. Then add that multiple of  $\log L_{\text{overflow}}$  to the logarithm in step (5). Another option is to use real or integer arbitrary precision arithmetic, but this slows down the calculation.
- We stress that Moussaïf’s technique for the computation of a braid’s entropy is extremely rapid compared to previous methods, which typically use *train tracks* and the Bestvina–Handel algorithm,<sup>37</sup> or combinatorial methods.<sup>26</sup> The rapidity arises from the fact that the algorithm keeps a bare minimum of information (the vector  $\mathbf{u}$ ) to express the topology of an arbitrarily long curve. The Bestvina–Handel algorithm, however, gives more information about the braid (such as the existence of invariant curves—see Sec. V).

The speed of convergence of this procedure is discussed in Refs. 14 and 33. As an example, Fig. 10(a) shows the result of applying the procedure to the braid in Fig. 2, and Fig. 10(b) shows the convergence rate to the exact entropy.

**IV. RANDOM BRAIDS**

From the point of view of data analysis, looking at periodic braid is not general enough. Most periodic orbits in a dynamical system are unstable, and thus they cannot be detected directly. The trajectories we have access to are typically chaotic. Nevertheless, we can ask what the braid corresponding to a set of orbits tells us about a dynamical system. The answer is that its entropy approximates the “true” topological entropy of the flow, and the approximation gets better as more particles are added.

There are two ways to analyze random braids generated by chaotic trajectories: without and with ensemble averaging. “Without averaging” means that we have a single realization to study, say  $n$  trajectories integrated or measured up to some final time. Unless the final time is extremely long, this is not very accurate. “With averaging” means that we have the luxury of repeating the experiment several times, following the same number of trajectories at each realization (assuming the flow is the same for each realization, at least in

a statistical sense). We then average over the total number of realizations of the experiment, in the manner described below.

### A. Entropy without averaging

Let us first describe the procedure without averaging: We assume that we have obtained a sequence of generators from the trajectories of  $n$  particles, as well as the time at which each crossing occurs (see Sec. II B). In the examples presented here, those trajectories were either computed from randomly selected initial conditions, or they were obtained from measured data (the oceanic floats).

**Procedure 2.** (Entropy of random braid, without averaging)

- (1) Start with an arbitrary initial loop, encoded as a vector  $\mathbf{u}$  [Eq. (4)];
- (2) for each generator in the braid, use the appropriate update rule from (8)–(15) to modify  $\mathbf{u}$ ;
- (3) compute the intersection number  $L(\mathbf{u})$  using Eq. (5);
- (4) plot  $\log L(\mathbf{u})$  versus  $t$ , where  $t$  is a vector of times when each crossing occurs;
- (5) repeat steps (2)–(4) until we can fit a line in step (4).

Remarks:

- Since the braid is random, we must keep track of the time when crossings occur.
- Most of the remarks from the periodic procedure 1 of Sec. III B still apply.

Figure 11 shows an example of applying procedure 2 to  $n=3$  particles advected by a blinking vortex flow<sup>4,7,8</sup> in the regular regime [Fig. 11(a)] and chaotic regime [Fig. 11(b)]. In the first case, the growth of  $L(\mathbf{u})$  is roughly linear so the entropy is zero. In the second case the growth is exponential. Note that the integration time is quite long, and in Fig. 11(b)  $L(\mathbf{u})$  becomes enormous. For such long integration time, the fit for the entropy is good.

### B. Entropy with averaging

To get a more accurate measurement of  $h_{\text{braid}}^{(n)}$  for random braids, ensemble averaging is desirable if we have that luxury. To implement this, we integrate a set of  $n$  trajectories  $N_{\text{real}}$  times, randomizing the initial condition for each realization. We obtain a list of  $n \times N_{\text{real}}$  trajectories for times  $0 \leq t \leq t_{\text{max}}$ , from which we compute  $N_{\text{real}}$  braids and vectors of crossing times. To do the averaging, we need to be able to compare  $L(\mathbf{u})$  at the same times for each braid, but since crossings occur at different times we cannot do this directly. We instead break up the total time interval into equal subintervals of length  $\Delta t$ , and for each subinterval and each realization we record  $L(\mathbf{u})$  up to time  $q\Delta t$ , where  $q$  is an integer with  $0 \leq q\Delta t \leq t_{\text{max}}$ . We finally obtain  $N_{\text{real}}$  lists of  $[t_{\text{max}}/\Delta t]$  (square brackets denote the integer part) intersection numbers  $L(\mathbf{u})$ , all sampled at the same times corresponding to each subinterval. The whole procedure is summarized as follows.

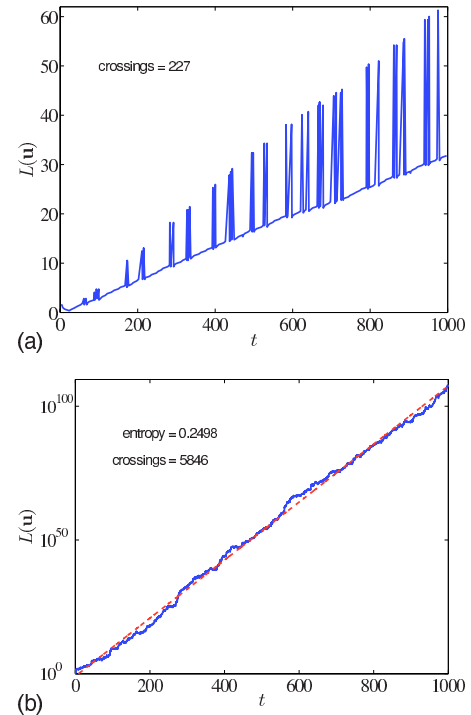


FIG. 11. (Color online) (a) The number of crossings of a loop vs time for the blinking vortex flow in the regular regime (circulation  $\Gamma=0.5$ , corotating vortices; see Ref. 7). The growth of  $L(\mathbf{u})$  is linear. (b) Same as in (a) but in the chaotic regime (circulation  $\Gamma=16.5$ , counter-rotating vortices; see Ref. 7). The vertical axis is now on a log scale; the slope of the line gives the braid entropy (procedure 2).

**Procedure 3.** (Entropy of random braid, with averaging)

- (1) Start with  $N_{\text{real}}$  lists of intersection numbers  $L(\mathbf{u})$  and their crossing times, generated following procedure 2, steps (1)–(3);
- (2) for each realization, record the intersection numbers up to fixed times  $q\Delta t$ ,  $0 \leq q\Delta t \leq t_{\text{max}}$ ;
- (3) at each time  $q\Delta t$ , compute the average  $\langle \log L(\mathbf{u}) \rangle$  over all  $N_{\text{real}}$  realizations;
- (4) plot  $\langle \log L(\mathbf{u}) \rangle$  versus  $q\Delta t$ ,  $0 \leq q\Delta t \leq t_{\text{max}}$ , and fit a line to get  $h_{\text{braid}}^{(n)}$ .

Remarks:

- We average  $\log L(\mathbf{u})$  rather than  $L(\mathbf{u})$ : Not only does this ensure that procedures 2 and 3 give the same entropy but it also leads to smaller fluctuations.
- For best results, the number of subintervals  $[t_{\text{max}}/\Delta t]$  has to be large enough to get a good fit, but small enough that there are several crossings within each subinterval of length  $\Delta t$ .

Figure 12 shows an example of applying procedure 3 with  $N_{\text{real}}=50$  realization of  $n=3$  particles advected by the same blinking vortex flow as for Fig. 11(b). Notice that the fit is much better, even though the integration time is shorter. We used  $[t_{\text{max}}/\Delta t]=10$  time subintervals of length  $\Delta t=10$ . An explicit example in MATLAB (for the Duffing oscillator) is given in Appendix A 3 (Fig. 13).

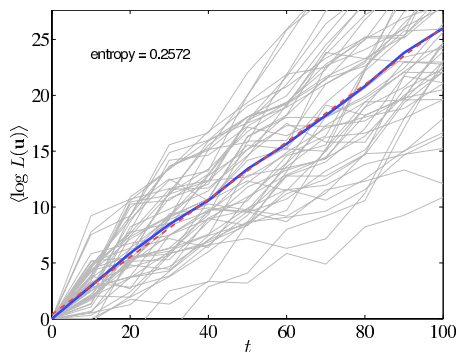


FIG. 12. (Color online) Similar plot to Fig. 11(b) but after averaging  $\langle \log L(\mathbf{u}) \rangle$  over  $N_{\text{real}}=50$  shorter trajectories ( $t_{\text{max}}=100$  time units rather than 1000). The average is plotted at each  $\Delta t=10$  time units (see procedure 3). The individual trajectories are shown in the background.

### C. Oceanic floats

As a more practical application of random braids, we consider data for oceanic floats in the Labrador Sea (North Atlantic),<sup>1</sup> discussed in Sec. I A. The position of ten floats for a few months is shown in Fig. 1(a). Note that the float trajectories seem more entangled while they are confined to the Labrador Sea (between Greenland and Labrador), and some eventually escape. To compute the braid, we linearly interpolate the float positions to determine when crossings occur between the  $n=10$  floats. We then use procedure 2 to compute the entropy, as shown in Fig. 1(b), since ensemble averaging is not available here (we only have data from one experiment). We see in Fig. 1(b) that  $L(\mathbf{u})$  has a convincing exponential regime for about 150 days, after which floats tend to escape the Labrador Sea and  $L(\mathbf{u})$  reaches a plateau. The entropy gives us a time scale for the entanglement of floats in the Labrador Sea, here about  $1/0.02 \approx 50$  days. This number is easy to obtain from the raw data: There is no need for a model of the velocity field. However, the trajectories need to be long enough for a significant number of crossings to occur, and localized enough for particles to actually braid.

More contexts will be needed to fully understand what it means to say that the time scale for entanglement is 50 days. For instance, the method could be benchmarked by following

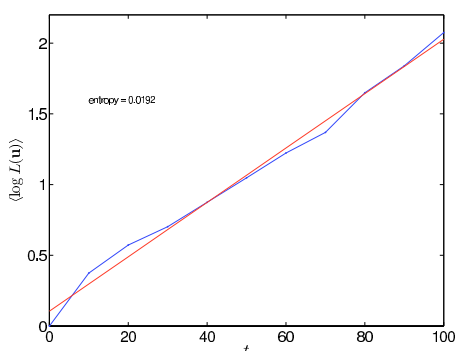


FIG. 13. (Color online) Output of PROC3\_EXAMPLE (Appendix A 3).

tracers in simulations of flows comparable to the Labrador Sea, in which braids can be easily computed. The measure is also useful for comparing different regions of the ocean.

Note that there has been previous work on bounding the topological entropy of experimental data. Amon and Lefranc<sup>38</sup> obtained lower bounds on entropy and evidence of chaotic behavior in a nonstationary optical system. See also the review by Gilmore.<sup>39</sup>

### V. DISCUSSION

There are two ways to interpret the data obtained from braids of particles. The first is to accumulate data for enough particle trajectories that a good approximation to the topological entropy  $h_{\text{flow}}$  is obtained. The drawback to this is that the convergence of  $h_{\text{braid}}^{(n)}$  to  $h_{\text{flow}}$  as  $n$  gets larger appears to be fairly slow,<sup>14</sup> although more work is needed to determine this convergence rate. In this interpretation, the braid approach is seen as a practical way of measuring  $h_{\text{flow}}$ . This interpretation is in the same spirit as a Lyapunov exponent. Its main advantage is that a single number is easy to comprehend and compare; its main drawback is that a single number does not capture the subtleties of a particular system.

The second interpretation is to regard  $h_{\text{braid}}^{(n)}$  as the “ $n$ -particle braiding time,” in a similar fashion that  $n$ -particle correlation functions are measured. Thus, the behavior of  $h_{\text{braid}}^{(n)}$  with  $n$  carries real information, as it tells us the typical time for three particle trajectories to become entangled, then four particles, etc. We might call this the “spectrum of braid entropies” for a dynamical system. The drawback of this approach is that it requires a more careful analysis of the data.

The method presented in this paper is limited to two-dimensional flows. Indeed, a four-dimensional braid of three-dimensional particle trajectories is not very useful, as this and all higher-dimensional braid groups are trivial (strands can always be disentangled without crossing<sup>31</sup>). The best alternative is to lift the trajectories of material lines to sheets in four dimensions, but this presents some daunting visualization challenges, and there is little developed theory (but see Ref. 40).

Finally, the topological entropy  $h_{\text{braid}}^{(n)}$  is only the crudest piece of information that can be extracted from a braid. Many other types of invariants can be computed.<sup>41</sup> However, those invariants do not necessarily have a clear interpretation in terms of dynamics. A promising avenue for obtaining much more precise information on a dynamical system is to find the isotopy class of the random braid (see Sec. III). This would tell us, for instance, whether some floats merely orbit each other and thus behave as one “trajectory” from the point of view of braiding and entropy. This is akin to making a braid out of thick rope: Even though each rope is made up of tiny strands, these contribute to the braid as one large strand. This sort of approach could help to identify Lagrangian coherent structures from particle trajectory data, by looking for decomposable braids. However, the tools available to do this, such as the Bestvina–Handel algorithm,<sup>37</sup> are still slow and



difficult to use on large braids. A promising approach was recently used in Ref. 35 but needs to be developed further.

## ACKNOWLEDGMENTS

The author thanks Karen Daniels, Tom Solomon, Matt Finn, Matt Harrington, and Philip Boyland for their help and

comments. This work was supported by the Division of Mathematical Sciences of the U.S. National Science Foundation, under Grant No. DMS-0806821.

## APPENDIX A: MATLAB EXAMPLE PROGRAMS

The source code of the MATLAB files in this appendix is available online.<sup>42</sup>

### 1. GENCROSS and INTERPCROSS

The function GENCROSS computes the generators and crossing times for particle trajectories (see Sec. II B). The function INTERPCROSS is a helper function to GENCROSS that interpolates crossings. Both these functions are simple implementations with few bells and whistles: GENCROSS deals with two or three adjacent particles crossing between successive time steps, but it does not attempt to refine the trajectory (by interpolation or integration) to resolve crossings. If it gets confused because too many crossings are occurring between two successive time steps, there is no other option but to refine the data further.

```
function [varargout]=gencross(t,X,Y)
%GENCROSS Find braid generators from crossings of trajectories.
% G=GENCROSS(T,X,Y) finds the braid group generators associated with
% crossings of particle trajectories. Here T is a column vector of times,
% and X and Y are coordinates of particles at those times. X and Y have
% the same number of rows as T, and N columns, where N is the number of
% particles. A projection on the X axis is used to define crossings.
%
% [G,TC]=GENCROSS(T,X,Y) also returns a vector of times TC when the
% crossings occurred.

% Find the permutation at each time.
[Xperm,Iperm]=sort(X,2);
dperm=diff(Iperm,1); % Crossings occur when the permutation changes.
icr=find(any(dperm,2)); % Index of crossings.
gen=[ ]; tcr=[ ];

for i=1:length(icr)
    % Order (from left to right) of particles involved in crossing.
    igen=find(dperm(icr(i),:));
    j=1;
    while j<length(igen)
        if ~sum(dperm(icr(i),igen(j:j+1)))
            %
            % Crossing involves a pair of particles.
            %
            P=Iperm(icr(i),igen(j:j+1)); % The two particles involved in crossing.
            [tt,dY]=interpcross(t,X,Y,icr(i),p(1),p(2));
            tcr=[tcr;tt]; gen=[gen;igen(j)*dY];
            j=j+2;
        elseif ~sum(dperm(icr(i),igen(j:j+2)))
            %
            % Crossing involves a triplet of particles.
            % Two cases are possible:
            %
            if Iperm(icr(i),igen(j))=Iperm(icr(i)+1,igen(j)+1)
                % Case 1: ABC->CAB

                % Particles B&C cross first
                P=Iperm(icr(i),igen([j+1j+2]));
                [tt,dY]=interpcross(t,X,Y,icr(i),p(1),p(2));
                tcr=[tcr;tt]; gen=[gen;igen(j+1)*dY];

                % Particles A&C cross second
                P=Iperm(icr(i),igen([jj+2]));
```

```

[tt,dY]=intercross(t,X,Y,icr(i),p(1),p(2));
tcr=[tcr;tt]; gen=[gen;igen(j)*dY];
elseif Iperm(icr(i),igen(j))=Iperm(icr(i)+1,igen(j)+2)
% Case 2: ABC- >BCA

% Particles A&B cross first
P=Iperm(icr(i),igen([j+1]));
[tt,dY]=intercross(t,X,Y,icr(i),p(1),p(2));
tcr=[tcr;tt]; gen=[gen;igen(j)*dY];

% Particles A&C cross second
P=Iperm(icr(i),igen([j+2]));
[tt,dY]=intercross(t,X,Y,icr(i),p(1),p(2));
tcr=[tcr;tt]; gen=[gen;igen(j+1)*dY];
else
error('something's wrong with triple crossing—increase resolution')
end
j=j+3;
else
error('too many simultaneous crossings—increase resolution')
end
end
end
varargout{1}=gen;
if nargout > 1, varargout{2}=tcr; end

```

---

```

function [tc,dY]=intercross(t,X,Y,icr,p1,p2)
%INTERPCROSS Interpolate a crossing.
% [TC,DY]=INTERPCROSS(T,X,Y,ITC,P1,P2) is a helper function for
% GENCROSS. The input is the data T,X,Y (described in the help for
% GENCROSS); the index ITC of the time of crossing (i.e., the particles
% cross between T(ITC) and T(ITC+1)); and the indices P1 and P2 of the two
% particles that are crossing. INTERPCROSS returns the interpolated
% crossing time TC, as well as DY (the sign of the difference in Y
% coordinates) which determines the sign of the generator.

% Refine crossing time and position (linear interpolation).
dt=t(itc+1)-t(itc); % Time interval.
% Particle velocities in that interval.
U1=(X(itc+1,p1)-X(itc,p1))/dt;
V1=(Y(itc+1,p1)-Y(itc,p1))/dt;
U2=(X(itc+1,p2)-X(itc,p2))/dt;
V2=(Y(itc+1,p2)-Y(itc,p2))/dt;
% Interpolated crossing time and Y coordinates.
dtc=-(X(itc,p2)-X(itc,p1))/(U2-U1);
tc=t(itc)+dtc;
Y1c=Y(itc,p1)+dtc*V1;
Y2c=Y(itc,p2)+dtc*V2;
dY=sign(Y1c-Y2c);
% The sign of Y1c-Y2c determines if the crossing is g or g^-1
if dY=0,
error('can't resolve sign of generator—increase resolution?');
end

```

---

## 2. LOOPSIGMA and LOOPINTER

The function `LOOPSIGMA` applies a sequence of braid group generators to a loop [Sec. III B, Eqs. (8)–(15)]. The function `LOOPINTER` computes  $L(\mathbf{u})$ , the minimum number of intersections of a loop with the horizontal axis [Sec. III B, Eq. (5)].

```

function up=loopsigma(ii,u)
%LOOPSIGMA Act on a loop with a braid group generator sigma.
% UP=LOOPSIGMA(J,U) acts on the loop U (encoded in Dynnikov coordinates)
% with the braid generator sigma_J, and returns the new loop UP. J can be
% a positive or negative integer (inverse generator), and can be specified
% as a vector, in which case all the generators are applied to the loop
% sequentially from left to right.

```

---

```

n=length(u)/2+2;
a=u(1:n-2); b=u((n-1):end);
ap=a; bp=b;
pos=@(x)max(x,0); neg=@(x)min(x,0);
for j=1:length(ii)
    i=abs(ii(j));
    if ii(j)>0
        switch(i)
            case 1
                bp(1)=a(1)+pos(b(1));
                ap(1)=-b(1)+pos(bp(1));
            case n-1
                bp(n-2)=a(n-2)+neg(b(n-2));
                ap(n-2)=-b(n-2)+neg(bp(n-2));
            otherwise
                c=a(i-1)-a(i)-pos(b(i))+neg(b(i-1));
                ap(i-1)=a(i-1)-pos(b(i-1))-pos(pos(b(i))+c);
                bp(i-1)=b(i)+neg(c);
                ap(i)=a(i)-neg(b(i))-neg(neg(b(i-1))-c);
                bp(i)=b(i-1)-neg(c);
            end
        end
    elseif ii(j)<0
        switch(i)
            case 1
                bp(1)=-a(1)+pos(b(1));
                ap(1)=b(1)-pos(bp(1));
            case n-1
                bp(n-2)=-a(n-2)+neg(b(n-2));
                ap(n-2)=b(n-2)-neg(bp(n-2));
            otherwise
                d=a(i-1)-a(i)+pos(b(i))-neg(b(i-1));
                ap(i-1)=a(i-1)+pos(b(i-1))+pos(pos(b(i))-d);
                bp(i-1)=b(i)-pos(d);
                ap(i)=a(i)+neg(b(i))+neg(neg(b(i-1))+d);
                bp(i)=b(i-1)+pos(d);
            end
        end
    end
    a=ap; b=bp;
end
up=[ap bp];

```

---

```

function int=loopinter(u)
%LOOPINTER The number of intersections of a loop with the real axis.
% I=LOOPINTER(U) computes the minimum number of intersections of a loop
% (encoded in Dynnikov coordinates) with the real axis. (See Moussafir
% (2006), Proposition 4.4.) U is either a row-vector, or a matrix of
% row-vectors, in which case the function acts vectorially on each row.

n=size(u,2)/2+2;
a=u(:,1:n-2); b=u(:,(n-1):end);
cumb=[zeros(size(u,1),1)cumsum(b,2)];
% The number of intersections before/after the first and last punctures.
% See Hall & Yurttas (2009).
b0=-max(abs(a)+max(b,0)+cumb(:,1:end-1),[],2); bn=-b0-sum(b,2);
int=sum(abs(b,2)+sum(abs(a(:,2:end)-a(:,1:end-1)),2)...
+abs(a(:,1))+abs(a(:,end))+abs(b0)+abs(bn));

```

---

### 3. PROC3\_EXAMPLE

The function PROC3\_EXAMPLE computes  $L(\mathbf{u})$  for a random braid of four particles advected by the Duffing oscillator, using averaging over 100 realizations (see Sec. IV, procedure 3).

```

function proc3_example

% Procedure 3 example:
% 4 particles advected by the Duffing oscillator, average over realizations.
N=4; Nreal=100;
tmax=100; dt=10; npts=3000;
X=zeros(npts,n); Y=zeros(npts,n);
Ll=zeros(Nreal,tmax/dt); tl=dt*(1:tmax/dt);
rand('twister',2);
for r=1:Nreal
    fprintf(1,'Realization %d',r)
    % Compute n particle trajectories, from random initial conditions.
    for i=1:n
        [t,xy]=ode45(@duffing,linspace(0,tmax,npts)',4*rand(1,2)-2);
        X(:,i)=xy(:,1); Y(:,i)=xy(:,2);
    end
    [gen,tc]=gencross(t,X,Y); % Find generators and crossing times.
    % Act with the generators on a random initial loop.
    up=rand(1,2*(n-2)); up=up/loopinter(up);
    for i=1:length(gen), up=[up;loopsigma(gen(i),up(end,:))]; end
    % Find the number of intersections with the real axis.
    L{r}=loopinter(up);
    % Keep intersections at a list of fixed time intervals dt, for averaging.
    for q=1:(tmax/dt)
        idx=find(tc <= tl(q));
        Ll(r,q)=L{r}(idx(end));
    end
end
logLavg=[0 mean(log(Ll),1)]; tl=[0 tl];
m=polyfit(tl,logLavg,1); % Fit a line
figure 2, plot(tl,logLavg,'b.-'), hold on
text(10,1.6,sprintf('entropy=%0.4f',m(1)))
plot(tl,m(1)*tl+m(2), 'r-'), hold off
xlabel('t'), ylabel('<log L>')

% -----
function yt=duffing(t,y)

delta=1; gamma=4; omega=2;
yt=zeros(size(y));
yt(1,:)=y(2,:);
yt(2,:)=y(1,:).*(1-y(1,:)).*y(1,:)+gamma*cos(omega*t)-delta*y(2,:);

```

<sup>1</sup>WOCE subsurface float data assembly center, 2004, <http://wfdac.whoi.edu>.

<sup>2</sup>J. H. LaCasce, in *Transport and Mixing in Geophysical Flows*, Lecture Notes in Physics Vol. 744, edited by J. B. Weiss and A. Provenzale (Springer, Berlin, 2008), pp. 165–218.

<sup>3</sup>A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, *Physica D* **16**, 285 (1985).

<sup>4</sup>H. Aref, *J. Fluid Mech.* **143**, 1 (1984).

<sup>5</sup>P. L. Boyland, H. Aref, and M. A. Stremler, *J. Fluid Mech.* **403**, 277 (2000).

<sup>6</sup>P. L. Boyland, M. A. Stremler, and H. Aref, *Physica D* **175**, 69 (2003).

<sup>7</sup>J.-L. Thiffeault, *Phys. Rev. Lett.* **94**, 084502 (2005).

<sup>8</sup>E. Kin and T. Sakajo, *Chaos* **15**, 023111 (2005).

<sup>9</sup>E. Gouillart, M. D. Finn, and J.-L. Thiffeault, *Phys. Rev. E* **73**, 036311 (2006).

<sup>10</sup>J.-L. Thiffeault, M. D. Finn, E. Gouillart, and T. Hall, *Chaos* **18**, 033123 (2008).

<sup>11</sup>M. D. Finn, S. M. Cox, and H. M. Byrne, *J. Fluid Mech.* **493**, 345 (2003).

<sup>12</sup>B. J. Binder and S. M. Cox, *Fluid Dyn. Res.* **40**, 34 (2008).

<sup>13</sup>A. Vikhansky, *Chaos* **14**, 14 (2004).

<sup>14</sup>M. D. Finn and J.-L. Thiffeault, *SIAM J. Appl. Dyn. Syst.* **6**, 79 (2007).

<sup>15</sup>M. D. Finn, J.-L. Thiffeault, and E. Gouillart, *Physica D* **221**, 92 (2006).

<sup>16</sup>J.-L. Thiffeault and M. D. Finn, *Philos. Trans. R. Soc. London, Ser. A* **364**, 3251 (2006).

<sup>17</sup>M. A. Stremler and J. Chen, *Phys. Fluids* **19**, 103602 (2007).

<sup>18</sup>A. Vikhansky, *Phys. Fluids* **15**, 1830 (2003).

<sup>19</sup>M. A. Berger, in *Topological Fluid Mechanics*, edited by H. K. Moffatt and A. Tsinober (Cambridge University Press, Cambridge, 1990), pp. 440–448.

<sup>20</sup>S. K. Nechaev, *Statistics of Knots and Entangled Random Walks* (World Scientific, Singapore, 1996).

<sup>21</sup>E. Artin, *Ann. Math.* **48**, 101 (1947).

<sup>22</sup>J. S. Birman, K. H. Ko, and S. J. Lee, *Adv. Math.* **139**, 322 (1998).

<sup>23</sup>J. S. Birman and T. E. Brendle, in *Handbook of Knot Theory*, edited by W. Menasco and M. Thistlethwaite (Elsevier, Amsterdam, 2005).

<sup>24</sup>I. Dynnikov and B. Wiest, *J. Eur. Math. Soc.* **9**, 801 (2007).

<sup>25</sup>J.-M. Gambaudo and E. E. Pécou, *Ergod. Theory Dyn. Syst.* **19**, 627 (1999).

<sup>26</sup>M. Lefranc, *Phys. Rev. E* **74**, 035202(R) (2006).

<sup>27</sup>A. Fathi, F. Laudenbach, and V. Poénaru, *Asterisque* **66–67**, 1 (1979).

<sup>28</sup>W. P. Thurston, *Bull., New Ser., Am. Math. Soc.* **19**, 417 (1988).

<sup>29</sup>A. J. Casson and S. A. Bleiler, *Automorphisms of Surfaces After Nielsen*

- and Thurston, *London Mathematical Society Student Texts* (Cambridge University Press, Cambridge, 1988), Vol. 9.
- <sup>30</sup>P. L. Boyland, *Topol. Appl.* **58**, 223 (1994).
- <sup>31</sup>J. S. Birman, *Braids, Links, and Mapping Class Groups, Annals of Mathematics Studies* (Princeton University Press, Princeton, NJ, 1975).
- <sup>32</sup>S. E. Newhouse and T. Pignataro, *J. Stat. Phys.* **72**, 1331 (1993).
- <sup>33</sup>J.-O. Moussafir, *Func. Anal. Other Math.* **1**, 37 (2006).
- <sup>34</sup>I. A. Dynnikov, *Russ. Math. Surveys* **57**, 592 (2002).
- <sup>35</sup>T. Hall and S. Ö. Yurttaş, *Topol. Appl.* **156**, 1554 (2009).
- <sup>36</sup>We are using the numbering scheme of Ref. 35 but the notation of Ref. 33.
- Also, we define generators  $\sigma_i$  as clockwise interchanges rather than counterclockwise.
- <sup>37</sup>M. Bestvina and M. Handel, *Topology* **34**, 109 (1995).
- <sup>38</sup>A. Amon and M. Lefranc, *Phys. Rev. Lett.* **92**, 094101 (2004).
- <sup>39</sup>R. Gilmore, *Rev. Mod. Phys.* **70**, 1455 (1998).
- <sup>40</sup>S. Kamada, *Braid and Knot Theory in Dimension Four, Mathematical Surveys & Monographs* (American Mathematical Society, Providence, 2002).
- <sup>41</sup>M. A. Berger, *Lett. Math. Phys.* **55**, 181 (2001).
- <sup>42</sup>See supplementary material at <http://dx.doi.org/10.1063/1.3262494> for the source code of the MATLAB files in this appendix.

## Lecture 28: The braidlab Matlab package

### 1 Installing braidlab

The package `braidlab` is defined inside a Matlab namespace, which are specified as subfolders beginning with a '+' character. The Matlab path must contain the folder that contains the subfolder `+braidlab`, and not the `+braidlab` folder itself:

```
>> addpath 'path to folder containing +braidlab'
```

To execute a `braidlab` *function*, either call it using the syntax `braidlab.function`, or import the whole namespace:

```
>> import braidlab.*
```

This allows invoking *function* by itself, without the `braidlab` prefix. For the remainder of this document, we assume this has been done and omit the `braidlab` prefix. The `addpath` and `import` commands can be added to `startup.m` to ensure they are executed at the start of every Matlab session.

### 2 A tour of braidlab

#### 2.1 The braid class

`braidlab` defines a number of classes, most importantly `braid` and `loop`. The braid  $\sigma_1\sigma_2^{-1}$  is defined by

```
>> a = braid([1 -2]) % defaults to 3 strings
a = < 1 -2 >
```

which defaults to the minimum required strings, 3. The same braid on 4 strings is defined by

```
> a4 = braid([1 -2],4) % force 4 strings
a4 = < 1 -2 >
```

Two braids can be multiplied:

```
>> a = braid([1 -2]); b = braid([1 2]);
>> a*b, b*a

ans = < 1 -2 1 2 >

ans = < 1 2 1 -2 >
```

Powers can also be taken, including the inverse:

```
>> a^5, inv(a), a*a^-1

ans = < 1 -2 1 -2 1 -2 1 -2 1 -2 >

ans = < 2 -1 >

ans = < 1 -2 2 -1 >
```

Note that this last expression is the identity braid, but is not simplified. The method `compact` attempts to simplify the braid:

```
>> compact(a*a^-1)

ans = < e >
```

The method `compact` is based on the heuristic algorithm of Bangert *et al.* (2002), since finding the braid of minimum length in the standard generators is in general difficult (Paterson & Razborov, 1991).

The number of strings is

```
>> a.n

ans = 3
```

Note that

```
>> help braid
```

describes the class `braid`. To get more information on the `braid` constructor, invoke

```
>> help braid.braid
```

which refers to the method `braid` within the class `braid`. (Use `methods(braid)` to list all the methods in the class.) There are other ways to construct a `braid`, such as using random generators, here a braid with 5 strings and 10 random generators:

```
>> braid('random',5,10)

ans = < 1  4 -4  2  4 -1 -2  4  4  4 >
```

The constructor can also build some standard braids:

```
>> braid('halftwist',5)

ans = < 4  3  2  1  4  3  2  4  3  4 >
```

In Section 2.2 we will also show how to construct a braid from a trajectory data set.

The `braid` class also handles equality of braids:

```
>> a = braid([1 -2]); b = braid([1 -2 2 1 2 -1 -2 -1]);
>> a == b

ans = 1
```

These are the same braid. Equality is determined efficiently by acting on loop (Dynnikov) coordinates (Dynnikov, 2002), as described by Dehornoy (2008). See Sections 2.3–2.4 for more details.

We can extract a subbraid by choosing specific strings: for example, if we take the 4-string braid  $\sigma_1\sigma_2\sigma_3^{-1}$  and discard the third string, we obtain  $\sigma_1\sigma_2^{-1}$ :

```
>> a = braid([1 2 -3]);
>> subbraid(a,[1 2 4]) % subbraid using strings 1,2,4

ans = < 1 -2 >
```

There are a few methods that exploit the connection between braids and homeomorphisms of the punctured disk. Braids label *isotopy classes* of homeomorphisms, so we can assign a topological entropy to a braid:

```
>> entropy(braid([1 2 -3]))

ans = 0.8314
```



The entropy is computed by iterated action on a loop (Moussafir, 2006). This can fail if the braid is finite-order or has very low entropy:

```
>> entropy(braid([1 2]))
Warning: Failed to converge to requested tolerance; braid is
likely finite-order or has low entropy.
> In braid.entropy at 89

ans = 0
```

To force the entropy to be computed using the Bestvina–Handel train track algorithm Bestvina & Handel (1995), we add an optional parameter:

```
>> entropy(braid([1 2]), 'trains')

ans = 0
```

Note that for large braids the Bestvina–Handel algorithm is impractical. But when applicable it can also determine the Thurston–Nielsen type of the braid (Fathi *et al.*, 1979; Thurston, 1988; Casson & Bleiler, 1988; Boyland, 1994):

```
>> tntype(braid([1 2 -3]))

ans = pseudo-Anosov
>> tntype(braid([1 2]))

ans = finite-order
>> tntype(braid([1 2],4)) % reducing curve around 1,2,3

ans = reducible
```

`braidlab` uses Toby Hall’s implementation of the Bestvina–Handel algorithm (Hall, 2012).

Finally, we can also find the Burau matrix representation (Burau, 1936; Birman, 1975) of a braid:

```
>> burau(braid([1 -2]), -1)

ans = 1    -1
      -1    2
```

where the last argument ( $-1$ ) is the value of the parameter  $t$  in the Laurent polynomials that appear in the entries of the Burau matrices.

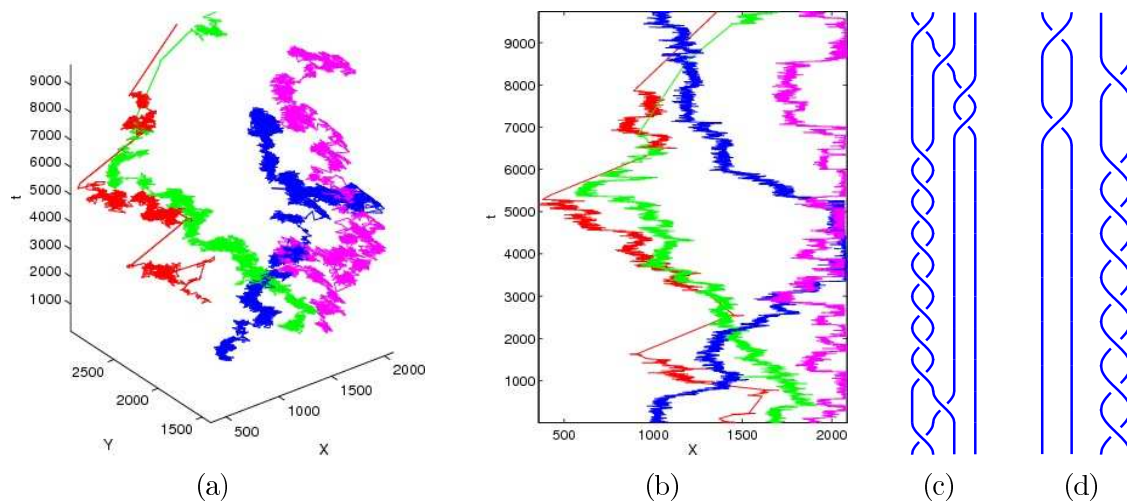


Figure 1: (a) A dataset of four trajectories, (b) projected along the  $X$  axis. (c) The compacted braid  $\sigma_1^{-1}\sigma_2^{-1}\sigma_1^{-8}\sigma_3^2\sigma_2\sigma_1$  corresponding to the  $X$  projection in (b). (d) The compacted braid  $\sigma_3^{-7}\sigma_1\sigma_3^{-1}\sigma_1$  corresponding to the  $Y$  projection, with closure enforced. The braids in (c) and (d) are conjugate.

## 2.2 Constructing a braid from data

One of the main purposes of `braidlab` is to analyze two-dimensional trajectory data using braids. We can assign a braid to trajectory data by looking for *crossings* along a projection line (Thiffeault, 2005, 2010). The `braid` constructor allows us to do this easily.

The folder `testing` contains a dataset of trajectories, from laboratory data for granular media Puckett *et al.* (2012). From the `testing` folder, we load the data:

```
>> clear; load testdata
>> whos
  Name          Size          Bytes  Class    Attributes
  XY            9740x2x4      623360 double
  ti            1x9740        77920  double
```

Here `ti` is the vector of times, and `XY` is a three-dimensional array: its first component specifies the timestep, its second specifies the  $X$  or  $Y$  coordinate, and its third specifies one of the 4 particles. Figure 1(a) shows the  $X$  and  $Y$  coordinates of these four trajectories, with time plotted vertically. Figure 1(b) shows the same data, but

projected along the  $X$  direction. To construct a braid from this data, we simply execute

```
>> b = braid(XY);
>> b.length

ans = 894
```

This is a very long braid! But Figure 1(b) suggests that this is misleading: many of the crossings are ‘wiggles’ that cancel each other out. Indeed, if we attempt to shorten the braid:

```
>> b = compact(b)

b = < -1 -2 -1 -1 -1 -1 -1 -1 -1 -1 3 3 2 1 >
>> b.length

ans = 14
```

we find the number of generators (the length) has dropped to 14! We can then plot this shortened braid as a braid diagram using `plot(b)` to produce Figure 1(c). The braid diagram allows us to see topological information clearly, such as the fact that the second and third particles undergo a large number of twists around each other; we can check this by creating a subbraid with only those two strings:

```
>> subbraid(bX,[2 3])

ans = < -1 -1 -1 -1 -1 -1 -1 -1 >
```

which shows that the winding number between these two strings is  $-4$ .

The braid was constructed from the data by assuming a projection along the  $X$  axis (the default). We can choose a different projection by specifying an optional angle for the projection line; for instance, to project along the  $Y$  axis we invoke

```
>> b = braid(XY,pi/2); % project onto Y axis
>> b.length

ans = 673
>> b.compact

ans = < -3 -3 -3 -3 -3 -3 -3 1 -3 >
```

In general, a change of projection line only changes the braid by conjugation (Boylan, 1994; Thiffeault, 2010). We can test for conjugacy:

```
>> bX = compact(braid(XY,0)); bY = compact(braid(XY,pi/2));
>> conjtest(bX,bY) % test for conjugacy of braids
```

```
ans = 0
```

The braids are not conjugate. This is because our trajectories do not form a ‘true’ braid: the final points do not correspond exactly with the initial points, as a set. If we truly want a rotationally-conjugate braid out of our data, we need to enforce a closure method:

```
>> XY = closure(XY); % close braid and avoid new crossings
>> bX = compact(braid(XY,0)), bY = compact(braid(XY,pi/2))
```

```
bX = < -1 -2 -1 -1 -1 -1 -1 -1 -1 -1 3 3 2 1 >
```

```
bY = < -3 -3 -3 -3 -3 -3 -3 1 -3 1 >
```

This default closure simply draws line segments from the final points to the initial points in such a way that no new crossings are created in the  $X$  projection. Hence, the  $X$ -projected braid  $\mathbf{bX}$  is unchanged by the closure, but here the  $Y$ -projected braid  $\mathbf{bY}$  is longer by one generator ( $\mathbf{bY}$  is plotted in Figure 1(d)). This is enough to make the braids conjugate:

```
>> [~,c] = conjtest(bX,bY) % ~ means discard first return arg
```

```
c = < 3 2 >
```

where the optional second argument  $c$  is the conjugating braid, as we can verify:

```
>> bX == c*bY*c^-1
```

```
ans = 1
```

There are other ways to enforce closure of a braid (see `help closure`), in particular `closure(XY, 'mindist')`, which minimizes the total distance between the initial and final points.

Note that `conjtest` uses the library *CBraid* (Cha, 2011) to first convert the braids to Garside canonical form (Birman & Brendle, 2005), then to determine conjugacy. This is very inefficient, so is impractical for large braids.

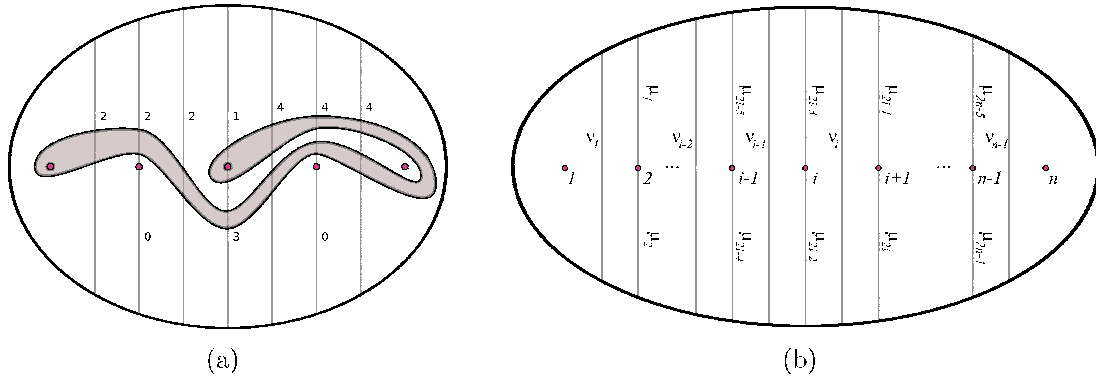


Figure 2: (a) A simple close loop in a disk with  $n = 5$  punctures. (b) Definition of intersection numbers  $\mu_i$  and  $\nu_i$ . [From Thiffeault (2010).]

### 2.3 The loop class

A simple closed loop on a disk with 5 punctures is shown in Figure 2(a). We consider equivalence classes of such loops under homotopies relative to the punctures. In particular, the loops are *essential*, meaning that they are not null-homotopic or homotopic to the boundary or a puncture. The *intersection numbers* are also shown in Figure 2(a): these count the minimum number of intersctions of an equivalence class of loops with the fixed vertical lines shown. For  $n$  punctures, we define the intersection numbers  $\mu_i$  and  $\nu_i$  in Figure 2(b).

Any given loop will lead to a unique set of intersection numbers, but a general collection of intersection numbers do not typically correspond to a loop. It is therefore more convenient to define

$$a_i = \frac{1}{2} (\mu_{2i} - \mu_{2i-1}), \quad b_i = \frac{1}{2} (\nu_i - \nu_{i+1}), \quad i = 1, \dots, n-2. \quad (1)$$

We then combine these in a vector of length  $(2n-4)$ ,

$$\mathbf{u} = (a_1, \dots, a_{n-2}, b_1, \dots, b_{n-2}), \quad (2)$$

which gives the *loop coordinates* (or *Dynnikov coordinates*) for the loop. (Some authors such as Dehornoy (2008) give the coordinates as  $(a_1, b_1, \dots, a_{n-2}, b_{n-2})$ .) There is now a bijection between  $\mathbb{Z}^{2n-4}$  and essential simple closed loops (Dynnikov, 2002; Moussafir, 2006; Hall & Yurttas, 2009; Thiffeault, 2010). Actually, *multiloops*: loop coordinates can describe unions of disjoint loops.

Let's create the loop in Figure 2(a) as a `loop` object:

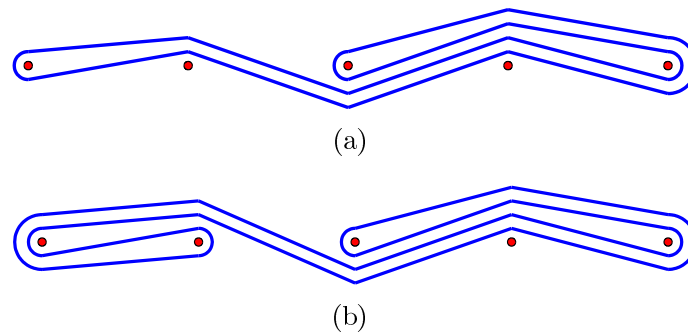


Figure 3:

```
>> l = loop([-1 1 -2 0 -1 0])
l = (( -1 1 -2 0 -1 0 ))
```

Figure 3(a) shows the output of the `plot(l)` command. Now we can act on this loop with braids. For example, we define the braid `b` to be  $\sigma_1^{-1}$  with 5 strings, corresponding to the 5 punctures, and then act on the loop `l` by using the multiplication operator:

```
>> b = braid([-1],5); % one generator with 5 strings
>> b*l % act on a loop with a braid

ans = (( -1 1 -2 1 -1 0 ))
```

Figure 3(b) shows `plot(b*l)`. The first and second punctures were interchanged counterclockwise (the action of  $\sigma_1^{-1}$ ), dragging the loop along.

The minimum length of an equivalence class of loops is determined by assuming the punctures are one unit of length apart and have zero size. After pulling tight the loop on the punctures, it is thus made up of unit-length segments. The minimum length is thus an integer. For the loop in Figure 3(a),

```
>> minlength(l)

ans = 12
```

The `entropy` method computes the topological entropy of a braid by repeatedly acting on a loop, and monitoring the growth rate of the loop.

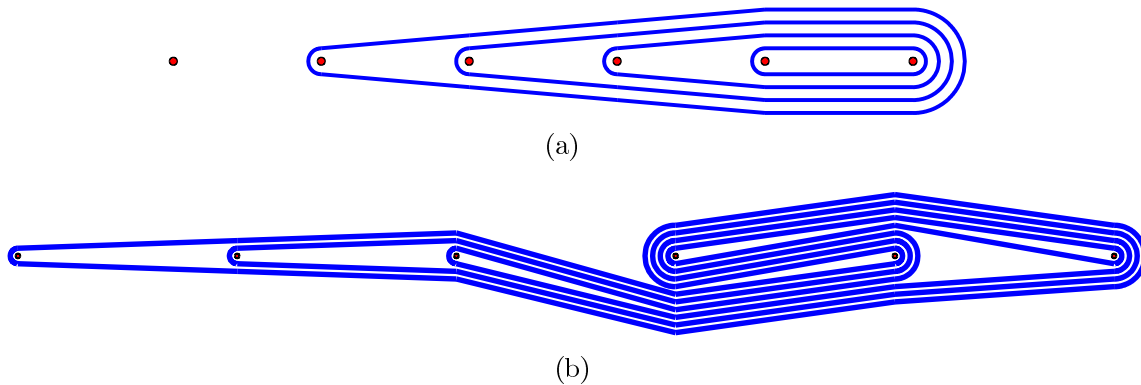


Figure 4: (a) The multiloop created by `loop(5)`. (b) The multiloop `b*loop(5)`, where `b` is the braid  $\sigma_1\sigma_2\sigma_3\sigma_4^{-1}$ .

```
>> b = braid([1 2 3 -4]);
% apply braid 100 times to l, then compute growth of length
>> log(minlength(b^100*1)/minlength(1)) / 100

ans = 0.7637
>> entropy(b)

ans = 0.7672
```

The entropy value returned by `entropy(b)` is more precise, since that method monitors convergence and adjusts the number of iterations accordingly.

## 2.4 Loop coordinates for a braid

The loop coordinates allow us to define a unique normal form for braids. Consider the multiloop depicted in Figure 4(a), which is the output of `plot(loop(5))`. Notice that `loop(5)` defaulted to a loop on a disk with 6 punctures. The reason is that this default multiloop is used to define loop coordinates for braids. The extra puncture is regarded as the outer boundary of the disk, and the loops form a generating set for the fundamental group of the disk with 5 punctures. The canonical loops coordinates for braids exploit the fact that two braids are equal if and only if they act the same way on the fundamental group of the disk. Hence, if we take a braid and act on `loop(5)`,

```
>> b = braid([1 2 3 -4]);
```

```
>> b*loop(5)
ans = (( 0  0  3 -1 -1 -1 -4  3 ))
```

then the set of numbers  $(( 0 0 3 -1 -1 -1 -4 3 ))$  can be thought of as *uniquely* characterizing the braid. It is this property that is used to rapidly determine equality of braids (Dehornoy, 2008). (The loop `b*loop(5)` is plotted in Figure 4(b).) The same loop coordinates for the braid can be obtained without creating an intermediate loop with

```
>> loopcoords(b)
ans = (( 0  0  3 -1 -1 -1 -4  3 ))
```

## Acknowledgments

The development of `braidlab` was supported by the US National Science Foundation, under grants DMS-0806821 and CMMI-1233935. The author thanks Michael Allshouse for extensive testing, comments, and for contributing some of the code. James Puckett and Karen Daniels provided the test data from their granular medium experiments (Puckett *et al.*, 2012). `braidlab` uses Toby Hall’s *Train* (Hall, 2012); Jae Choon Cha’s *CBraid* (Cha, 2011); Juan González-Meneses’s *Braiding* (González-Meneses, 2011); and Markus Buerhen’s *assignmentoptimal* (Buerhen, 2011).

## References

- BANGERT, P. D., BERGER, M. A. & PRANDI, R. 2002 In search of minimal random braid configurations. *J. Phys. A* **35** (1), 43–59.
- BESTVINA, M. & HANDEL, M. 1995 Train-tracks for surface homeomorphisms. *Topology* **34** (1), 109–140.
- BIRMAN, J. S. 1975 *Braids, Links, and Mapping Class Groups*. *Annals of Mathematics Studies* 82. Princeton, NJ: Princeton University Press.
- BIRMAN, J. S. & BRENDLE, T. E. 2005 Braids: A survey. In *Handbook of Knot Theory* (ed. W. Menasco & M. Thistlethwaite), pp. 19–104. Amsterdam: Elsevier, available at <http://arXiv.org/abs/math.GT/0409205>.



- BOYLAND, P. L. 1994 Topological methods in surface dynamics. *Topology Appl.* **58**, 223–298.
- BUERHEN, M. 2011 Functions for the rectangular assignment problem. <http://www.mathworks.com/matlabcentral/fileexchange/6543>.
- BURAU, W. 1936 Über Zopfgruppen und gleichsinnig verdrehte Verkettungen. *Abh. Math. Semin. Hamburg Univ.* **11**, 171–178.
- CASSON, A. J. & BLEILER, S. A. 1988 *Automorphisms of surfaces after Nielsen and Thurston*, London Mathematical Society Student Texts, vol. 9. Cambridge: Cambridge University Press.
- CHA, J. C. 2011 *CBraid: A C++ library for computations in braid groups*. <http://code.google.com/p/cbraid>.
- DEHORNOY, P. 2008 Efficient solutions to the braid isotopy problem. *Discr. Applied Math.* **156**, 3091–3112.
- DYNNIKOV, I. A. 2002 On a Yang–Baxter map and the Dehornoy ordering. *Russian Math. Surveys* **57** (3), 592–594.
- FATHI, A., LAUNDENBACH, F. & POÉNARU, V. 1979 Travaux de Thurston sur les surfaces. *Astérisque* **66-67**, 1–284.
- GONZÁLEZ-MENESES, J. 2011 *Braiding: A computer program for handling braids*. The version used is distributed with *CBraid*: <http://code.google.com/p/cbraid>.
- HALL, T. 2012 *Train: A C++ program for computing train tracks of surface homeomorphisms*. [http://www.liv.ac.uk/~tobyhall/T\\_Hall.html](http://www.liv.ac.uk/~tobyhall/T_Hall.html).
- HALL, T. & YURTTAŞ, S. Ö. 2009 On the topological entropy of families of braids. *Topology Appl.* **156** (8), 1554–1564.
- MOUSSAFIR, J.-O. 2006 On computing the entropy of braids. *Func. Anal. and Other Math.* **1** (1), 37–46.
- PATERSON, M. S. & RAZBOROV, A. A. 1991 The set of minimal braids is co-NP complete. *J. Algorithm* **12**, 393–408.

- PUCKETT, J. G., LECHENAULT, F., DANIELS, K. E. & THIFFEAULT, J.-L. 2012 Trajectory entanglement in dense granular materials. *Journal of Statistical Mechanics: Theory and Experiment* **2012** (6), P06008.
- THIFFEAULT, J.-L. 2005 Measuring topological chaos. *Phys. Rev. Lett.* **94** (8), 084502.
- THIFFEAULT, J.-L. 2010 Braids of entangled particle trajectories. *Chaos* **20**, 017516.
- THURSTON, W. P. 1988 On the geometry and dynamics of diffeomorphisms of surfaces. *Bull. Am. Math. Soc.* **19**, 417–431.

## Lecture 29: Mix-norms\*

### 1 Norms

In this lecture we define a measure of mixing that does not necessarily require diffusion to measure the amount of homogenization that occurs during the mixing process. Recall the advection-diffusion equation

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = \kappa \nabla^2 \theta, \quad (1)$$

where  $\theta$  is a concentration field in a finite domain  $\Omega$ , with no-net-flux boundary conditions. We assume without loss of generality that

$$\int_{\Omega} \theta \, d\Omega = 0, \quad (2)$$

and define the  $L^2$ -norm, or variance, as

$$\|\theta\|_2^2 = \int_{\Omega} \theta^2 \, d\Omega. \quad (3)$$

Recall from Lecture 1 that the variance evolves according to

$$\frac{d}{dt} \|\theta\|_2^2 = -2\kappa \|\nabla \theta\|_2^2, \quad (4)$$

and decays in time as the system mixes. The variance indicates the extent to which the concentration has homogenized and is thus a good measure of the amount of mixing that has occurred. However, the variance requires knowledge of small scales in  $\theta$ , which we are not necessarily interested in. A measure of how well-mixed the concentration is does not necessarily require knowledge of how much homogenization has occurred due to diffusion at small scales. This is more in keeping with the definition of mixing in the sense of ergodic theory [2]. In this regard, we proceed to consider the pure advection equation

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = 0. \quad (5)$$

Note that in this case equation (4) predicts that the variance satisfies

$$\frac{d}{dt} \|\theta\|_2^2 = 0, \quad (6)$$

---

\*Notes by Sam Pegler and Woosok Moon of lectures delivered at the 2010 *Woods Hole Summer Program in Geophysical Fluid Dynamics*.

and cannot therefore be used as a measure of mixing.

The advection equation (5) takes us closer to the ergodic sense of mixing in which we think of the advection due to the velocity field as a time-dependent operator  $S^t : \Omega \rightarrow \Omega$  that moves an initial patch of dye according to

$$\theta_0(\mathbf{x}) \mapsto \theta(\mathbf{x}, t) = S^t \theta_0(\mathbf{x}). \quad (7)$$

If we consider a region  $A$  of uniform concentration defined by

$$\theta_0(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in A, \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

then the volume of the patch

$$\text{Vol}[\theta(\mathbf{x}, t)] = \text{Vol}(A), \quad (9)$$

remains constant in time by incompressibility. We can associate the volume of the patch with the Lebesgue measure and, because of (9),  $S^t$  is measure-preserving.

We define mixing in the sense of ergodic theory by

$$\lim_{t \rightarrow \infty} \text{Vol}[A \cap S^t(B)] = \text{Vol}(A)\text{Vol}(B), \quad (10)$$

for all patches  $A, B \in \Omega$ . This definition follows our intuition for what good mixing is. Referring to figure 1, when the system is well-mixed the intersection of  $A$  and  $S^t B$  is proportional to both  $\text{Vol}(A)$  and  $\text{Vol}(B)$ . Thus, if the condition (10) holds then  $S^t$  must spread any initial patch throughout the domain. This condition is referred to as *strong mixing* and can be shown to imply ergodicity.

The intersection of the advected patch  $B$  with the reference patch  $A$  is analogous to projection onto  $L^2$  functions. This motivates the following *weak convergence* condition

$$\lim_{t \rightarrow \infty} \langle \theta(\mathbf{x}, t), g \rangle = 0, \quad (11)$$

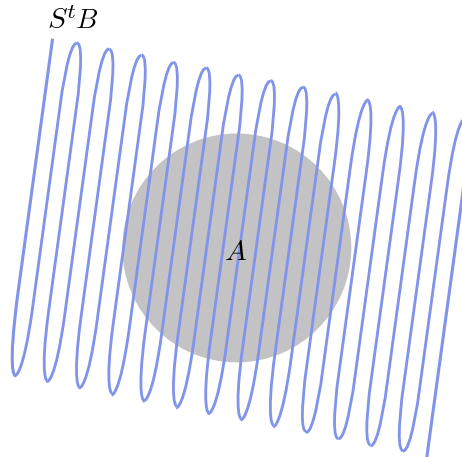


Figure 1: An advected patch  $S^t B$  that has undergone strong mixing. At late times the patch covers an arbitrary reference patch  $A$ .

for all functions  $g \in L^2(\Omega)$ , where the inner product is defined by

$$\langle f, g \rangle = \int_{\Omega} f(\mathbf{x})g(\mathbf{x}) \, d\Omega, \tag{12}$$

and  $f \in L^2(\Omega)$  if  $\int_{\Omega} |f|^2 \, d\Omega < \infty$ . Weak convergence is equivalent to mixing as a consequence of the Riemann–Lebesgue lemma. The equivalent conditions (10) and (11) require computing over all patches  $A$  or functions  $g$ , respectively. Thus, neither of these conditions is useful in practice. However, we proceed to describe a theorem that shows there is a simpler way to determine whether or not weak convergence is satisfied.

Mathew, Mezic and Petzold [5] introduced the mix-norm, which for mean-zero functions is equivalent to

$$\|\theta\|_{\dot{H}^{-1/2}} := \|\nabla^{-1/2}\theta\|_2. \tag{13}$$

Doering and Thiffeault [1] and Lin, Thiffeault and Doering [3] generalized the mix-norm to

$$\|\theta\|_{\dot{H}^q} := \|\nabla^q\theta\|_2, \quad q < 0, \tag{14}$$

which is a negative homogeneous Sobolev norm. This norm can be interpreted for negative  $q$  via eigenfunctions of the Laplacian operator. For example, in a periodic domain, we have

$$\|\theta\|_{\dot{H}^q}^2 = \sum_{\mathbf{k}} |\mathbf{k}|^{2q} |\hat{\theta}_{\mathbf{k}}|^2, \tag{15}$$

from which we see that, for  $q < 0$ ,  $\|\theta\|_{\dot{H}^q}^q$  smooths  $\theta$  before taking the  $L^2$  norm. The theorem

$$\lim_{t \rightarrow \infty} \|\theta\|_{\dot{H}^q} = 0, \quad q < 0 \iff \theta \text{ converges weakly to } 0, \tag{16}$$

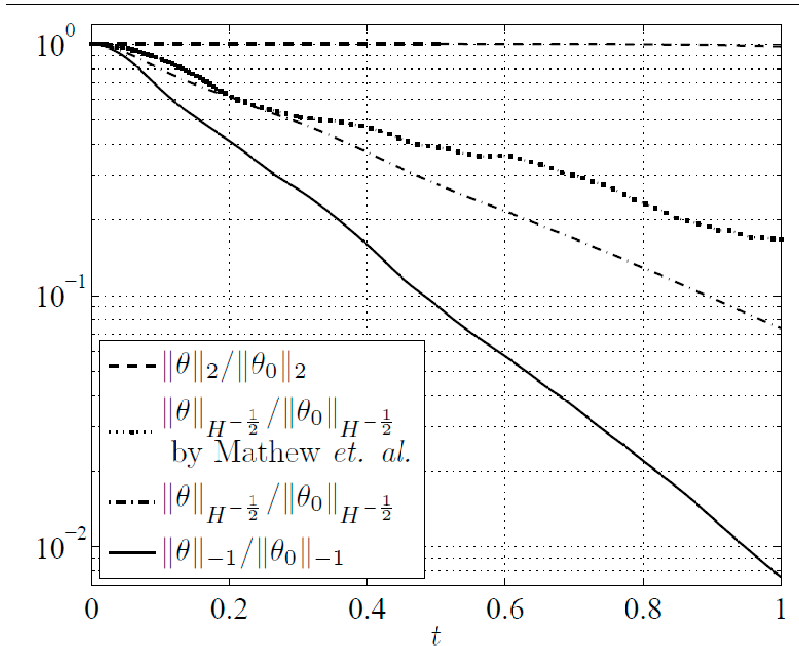


Figure 2: Comparison of the mix-norms for a flow optimized using the separate methods of optimal control and optimal instantaneous decay. Figure from Lin *et al.* [3].

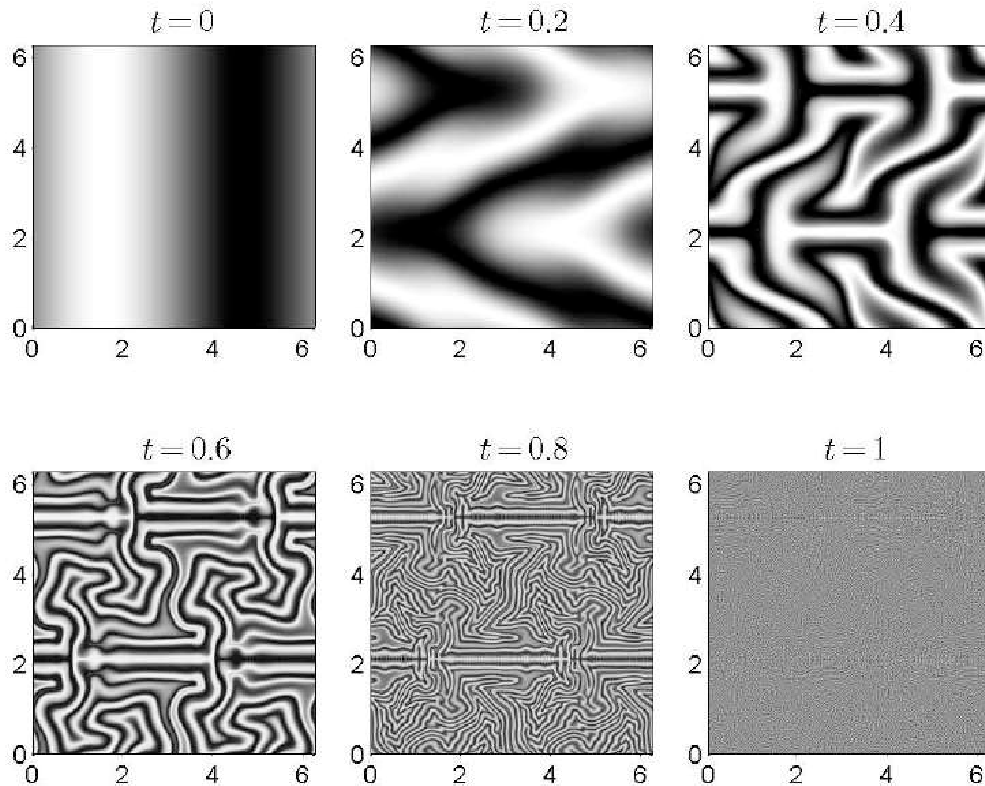


Figure 3: Evolution of the concentration field for the flow optimized in the case  $q = -1$  as computed by Lin *et al.* [3].

due to Mathew, Mezic and Petzold [5] and Doering, Lin and Thiffeault [3] shows that we can track any mix-norm to determine whether a system is mixing (in the weak sense). The existence of this quadratic norm facilitates optimization of the velocity field to achieve good mixing. Mathew, Mezic, Grivopoulos, Vaidya and Petzold [4] have used optimal control to optimize the decay of the  $q = -1/2$  mix-norm. Lin, Doering and Thiffeault [3] have optimized the instantaneous decay rate of the  $q = -1$  norm using the method of steepest descent, which is easier to compute numerically but yields suboptimal, but nevertheless very effective, stirring velocity fields. A comparison of the methods for optimized mixing is shown in figure 2. The solid line decays faster, but this is merely because the  $\dot{H}^{-1}$  cannot be compared directly with  $\dot{H}^{-1/2}$ . The corresponding evolution of the concentration field for the case  $q = -1$  from Lin *et al.* [3] is shown in figure 3.

## References

- [1] C. R. DOERING AND J.-L. THIFFEAULT, *Multiscale mixing efficiencies for steady sources*, Phys. Rev. E, 74 (2006), p. 025301(R).

- [2] A. LASOTA AND M. C. MACKEY, *Chaos, Fractals, and Noise*, Springer-Verlag, New York, 1994.
- [3] Z. LIN, C. R. DOERING, AND J.-L. THIFFEAULT, *Optimal stirring strategies for passive scalar mixing*, J. Fluid Mech., 675 (2011), pp. 465–476.
- [4] G. MATHEW, I. MEZIĆ, S. GRIVOPOULOS, U. VAIDYA, AND L. PETZOLD, *Optimal control of mixing in Stokes fluid flows*, J. Fluid Mech., 580 (2007), pp. 261–281.
- [5] G. MATHEW, I. MEZIĆ, AND L. PETZOLD, *A multiscale measure for mixing*, Physica D, 211 (2005), pp. 23–46.
- [6] J.-L. THIFFEAULT, *Using multiscale norms to quantify mixing and transport*, Nonlinearity, 25 (2012), pp. R1–R44.

## Lecture 30: Mixing in the presence of sources and sinks\*

### 1 Sources and Sinks

Consider the situation with a sources-sink term  $s(\mathbf{x}, t)$ ,

$$\begin{aligned}\partial_t \theta + \mathbf{u} \cdot \nabla \theta &= \kappa \nabla^2 \theta + s(\mathbf{x}, t), \\ \nabla \cdot \mathbf{u} &= 0.\end{aligned}\tag{1}$$

For simplicity, assume that  $\int_{\Omega} s(\mathbf{x}, t) d\Omega = 0$ . Otherwise, we can subtract the mean of  $\theta$ . It is convenient to think of sources and sinks as hot and cold regions.

Let us also assume that our sources and sinks are time-independent. Then, the system eventually achieves a steady state  $\theta(\mathbf{x})$  that satisfies

$$\mathbf{u} \cdot \nabla \theta = \kappa \nabla^2 \theta + s.\tag{2}$$

We define the operator

$$\mathcal{L} \equiv \mathbf{u} \cdot \nabla - \kappa \nabla^2\tag{3}$$

so that (2) can be written

$$\mathcal{L}\theta = s.\tag{4}$$

The steady solution is then

$$\theta = \mathcal{L}^{-1}s\tag{5}$$

where the mean-zero condition on  $\theta$  makes this unique. Note that  $\kappa \neq 0$  is needed to achieve a steady state. So, assuming the system has reached a steady-state, we have to determine how we measure the quality of mixing. One of the possible ways is to look at the norms  $\|\theta\|_{\dot{H}^q}$ , where  $q = 0$  represents standard derivation. But we have to decide what we will compare to. One possibility is  $\|\theta\|_{\dot{H}^q} / \|s\|_{\dot{H}^q}$ . This ratio is a reasonable choice, but has units of inverse time. It is preferable to use a dimensionless quantity for measuring the quality of mixing. In this spirit, we define *mixing enhancement factors*:

$$\varepsilon_q = \frac{\|\tilde{\theta}\|_{\dot{H}^q}}{\|\theta\|_{\dot{H}^q}},\tag{6}$$

where  $\tilde{\theta}$  is the purely-diffusive solution which satisfies

$$\tilde{\mathcal{L}}\tilde{\theta} = s.$$

---

\*Notes by Sam Pegler and Woosok Moon of lectures delivered at the 2010 *Woods Hole Summer Program in Geophysical Fluid Dynamics*.



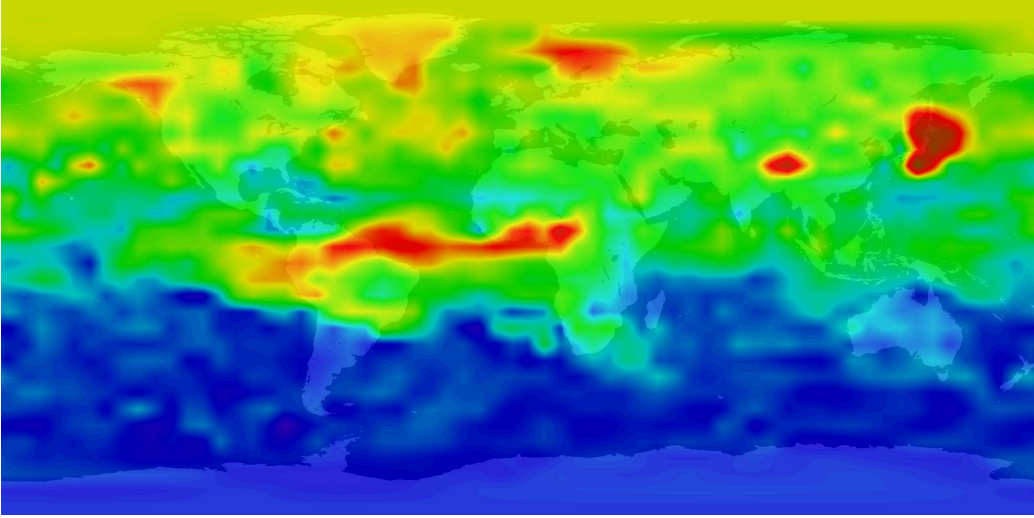


Figure 1: Sources and sinks: CO in the atmosphere. Red corresponds to high levels of CO (450 parts per billion) and blue to low levels (50 ppb). Note the immense clouds due to grassland and forest fires in Africa and South America. (Photo NASA/NCAR/CSA.)

Here,  $\tilde{\mathcal{L}} = -\kappa\nabla^2$  is the pure diffusive operator, so  $\tilde{\theta}$  can be interpreted as the solution in the absence of stirring. Since  $\|\theta\|_{\dot{H}^q}$  is usually decreased by stirring,  $\varepsilon_q$  measures the enhancement over the pure-diffusion state. Several properties are given in Doering and Thiffeault [1], Shaw, Thiffeault and Doering [3], and Thiffeault and Pavliotis [5]; see also the review by Thiffeault [4]. We interpret a large  $\varepsilon_q$  as ‘good stirring,’ since it in that case the norm is decreased by stirring.

A natural question is whether  $\varepsilon_q$  can ever be less than unity, that is, if stirring can ever be worse than not stirring. Let’s consider

$$\varepsilon_1 = \frac{\|\nabla\tilde{\theta}\|_2}{\|\nabla\theta\|_2}$$

Here,

$$\tilde{\theta} = \tilde{\mathcal{L}}^{-1}s = (-\kappa\nabla^2)^{-1}s = -\kappa^{-1}\nabla^{-2}s \Rightarrow \nabla\tilde{\theta} = -\kappa^{-1}\nabla^{-1}s$$

Also, from  $\mathcal{L}\theta = s$ , we can multiply  $\theta$  on both sides and take spatial average and then get

$$\langle\theta\mathcal{L}\theta\rangle = \langle s\theta\rangle,$$

where  $\langle\cdot\rangle = \int_{\Omega}\cdot d\Omega$ . We expand the left-hand side:

$$\begin{aligned}\langle\theta\mathcal{L}\theta\rangle &= \langle\theta u\cdot\nabla\theta\rangle - \kappa\langle\theta\nabla^2\theta\rangle \\ &= \langle\nabla\cdot(u\theta^2/2)\rangle - \kappa\langle\theta\nabla^2\theta\rangle \\ &= -\kappa\langle\theta\nabla^2\theta\rangle = \kappa\langle|\nabla\theta|^2\rangle.\end{aligned}$$

As for the right-hand side, it can be written as

$$\langle\theta s\rangle = \langle\theta\nabla\cdot\nabla^{-1}s\rangle = -\langle\nabla\theta\cdot\nabla^{-1}s\rangle = \kappa\langle\nabla\theta\cdot\nabla\tilde{\theta}\rangle,$$

where we used

$$\begin{aligned}\tilde{\theta} &= \tilde{\mathcal{L}}^{-1}s = (-\kappa\nabla^2)^{-1}s = -\kappa^{-1}\nabla^{-2}s \\ &\iff \nabla\tilde{\theta} = -\kappa^{-1}\nabla^{-1}s.\end{aligned}$$

Recall that  $\langle |\nabla\theta|^2 \rangle = \|\theta\|_{\dot{H}^1}^2$ . Therefore,

$$\|\theta\|_{\dot{H}^1}^2 = \langle \nabla\theta \cdot \nabla\tilde{\theta} \rangle \leq \|\nabla\theta\|_2 \|\nabla\tilde{\theta}\|_2 = \|\theta\|_{\dot{H}^1} \|\tilde{\theta}\|_{\dot{H}^1}.$$

We conclude that

$$\|\theta\|_{\dot{H}^1} \leq \|\tilde{\theta}\|_{\dot{H}^1} \iff \varepsilon_1 \leq 1. \quad (7)$$

This is somewhat counter-intuitive because gradients are usually increased by stirring. However, the gradients in a steady-state have been affected by diffusion.

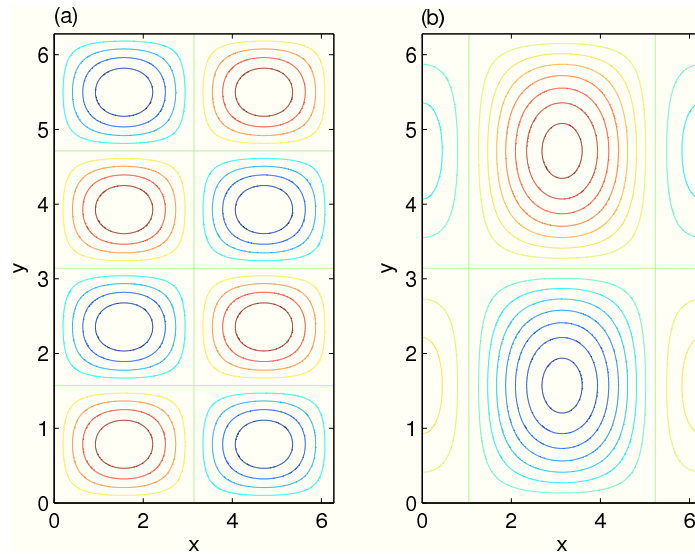


Figure 2: The pattern of velocity field and source for the ‘unmixing’ flow and source distribution (8).

What about  $\varepsilon_q$  for values of  $q$  other than 1? We tried and failed to prove  $\varepsilon_q \leq 1$ , simply because it is not true. Following a challenge by Charlie Doering at a workshop at the IMA in 2010, Jeff Weiss came up with something like:

$$u = (2 \sin x \cos 2y, -\cos x \sin 2y), \quad (8a)$$

$$s = (\cos x - \frac{1}{2}) \sin y. \quad (8b)$$

This velocity field manages to concentrate the source and sink distribution more than diffusion alone. Streamlines of  $u$  and level sets of  $s$  are shown in figure 2. In this example, we could get  $\varepsilon_0 \simeq 0.978$  and  $\varepsilon_{-1} \simeq 0.945$ , which are slightly less than 1. It is an open problem to characterize such ‘unmixing’ flows.

## 2 Optimization

We defined the mixing enhancement factors based on Sobolev norms. Large mixing enhancement factor indicates good mixing for a given source and sink pattern. One of the relevant questions in this step is what kinds of flow give the largest  $\varepsilon_q$  given source and sink distribution  $s(\mathbf{x})$ .

Here is a simple but surprising example. The source and sink distribution is given by  $s(x) = \sin x$  with periodic boundary conditions on  $\theta$ . The optimal solution for this source and sink distribution is  $\mathbf{u} = U\hat{x}$ , which is constant flow from the hot region to the cold region [2,3,5] (figure 3). This example demonstrates that, with body sources, the best stirring often has more to do with transport than with creation of small scales.

More generally, we have to solve the optimization problem numerically. Figure 4 shows contours of the streamfunction for the optimal stirring velocity (lines) for a source  $s(\mathbf{x}) = \sin x \sin y$ , for  $q = 0$  and  $q = -1$ . The optimal velocity fields are identical for the two values of  $q$ , because the source is an eigenfunction of the Laplacian.

Contrast this to the optimal solutions in figure 5, for the source distribution  $s(\mathbf{x}) = \cos x \cos y + \cos 3y + (1/4)\sin 3y$ . This source is not an eigenfunction of the Laplacian, and we expect optimal solutions to depend on  $q$ . Comparing the left ( $q = 0$ ) and right ( $q = -1$ ) figures, we see this is indeed the case, though the difference in this case is fairly small.

Finally, given an optimization code, it is simple to turn it around to anti-optimize, that is, find the *worst* stirring velocity for a given source distribution. Figure 6 shows this for the source (8b) and  $q = 0$ . Note how the velocity field seems to work to concentrate the source sink, thereby increasing the variance. The efficiency for this anti-optimal solution is  $\varepsilon_0 = 0.9736$ , which is not much lower than Jeff Weiss's unoptimized flow (8a), which had  $\varepsilon_0 \simeq 0.978$ .

To reproduce the 5 figures in this section run the program `example(n)` in the Appendix, where  $n$  is a number from 1 to 5.

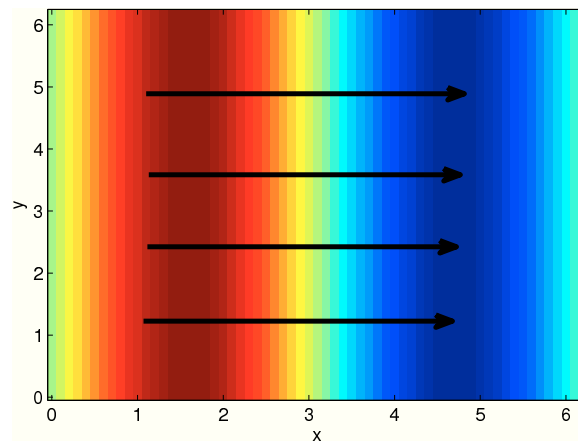


Figure 3: The optimal velocity field (solid arrows) for the source distribution  $s(x) = \sin x$ .

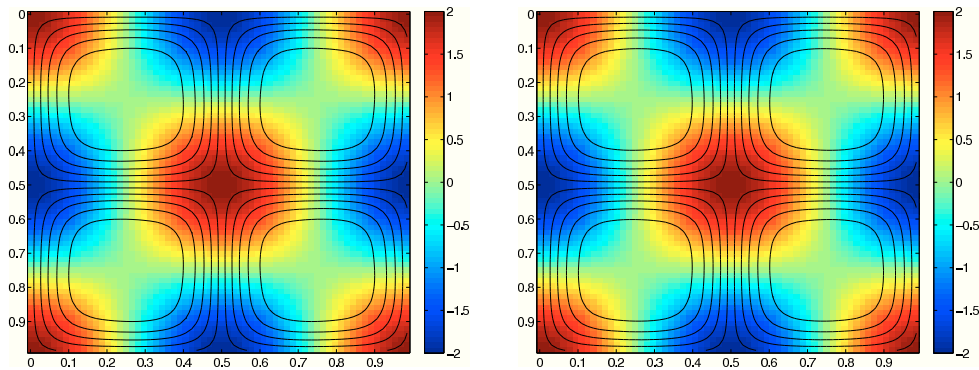


Figure 4: Optimal stirring velocity field (solid lines) for the source  $s(\mathbf{x}) = \sin x \sin y$  (colored background), for  $q = 0$  (left) and  $q = -1$  (right). The optimal velocity is the same in both cases because the source is an eigenfunction of the Laplacian. (Matlab programs `example(1)` and `example(2)` in the Appendix.)

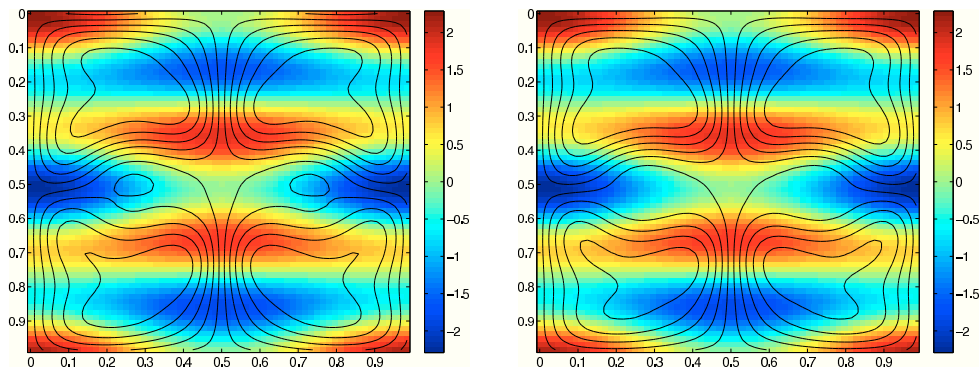


Figure 5: Optimal stirring velocity field (solid lines) for the source  $\cos x \cos y + \cos 3y + (1/4) \sin 3y$  (colored background), for  $q = 0$  (left) and  $q = -1$  (right). The optimal velocities are different since the source is not an eigenfunction of the Laplacian. (Matlab programs `example(3)` and `example(4)` in the Appendix.)

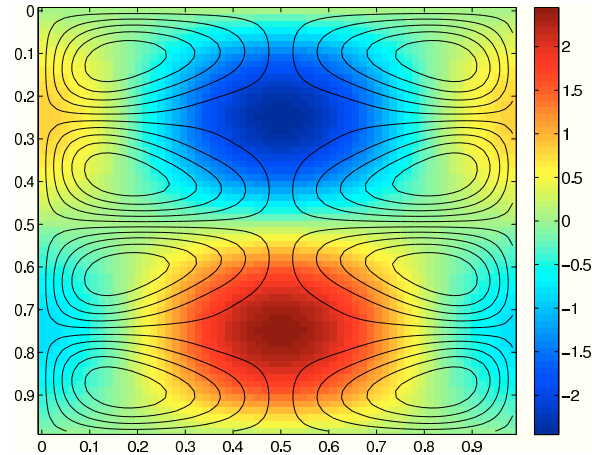


Figure 6: Optimal ‘unmixing’ solution for the source  $(\cos x - \frac{1}{2}) \sin y$ , with mixing efficiency  $\varepsilon_0 = 0.9736$ . (Matlab program `example(5)` in the Appendix.)

## Appendix: Matlab code

### 1 Program file `example.m`

```
function example(ex)

if nargin < 1, ex = 1; end

N = 11; % Number of gridpoints
L = 1; k1 = 2*pi/L; % Physical size of domain
x = L*(0:N-1)/N; y = x'; [xx,yy] = meshgrid(x,y);

switch ex
% Examples 1 and 2 use the same cellular source, for q=0 and q=-1.
% Since the source is an eigenfuntion of the Laplacian, the optimal
% flow is the same for q=-1.
case {1,2}
    src = cos(k1*xx).*cos(k1*yy) * 2/L;
    psi0 = sin(k1*xx).*sin(k1*yy) * 1/sqrt(2)/pi;
    kappa = .1; q = 1-ex;
% Examples 3 and 4 use the same two-mode source, for q=0 and q=-1.
% Since the source is not an eigenfuntion of the Laplacian, the optimal
% flow is different for q=-1.
case {3,4}
    src = (cos(k1*xx).*cos(k1*yy)+cos(3*k1*yy)+.25*sin(3*k1*yy)) * 4*sqrt(2)/5/L;
    psi0 = sin(k1*xx).*sin(k1*yy) * 1/sqrt(2)/pi;
    kappa = .1; q = 3-ex;
case 5
    % Unmixing solution
    src = (cos(k1*xx) - .5).*sin(k1*yy) * 2*sqrt(2/3)/L;
    psi0 = sin(k1*xx).*sin(2*k1*yy) * 1/sqrt(5)/pi;
    scalefac = -N^2; % Set scale factor negative to minimize instead
    kappa = 1/4; q = 0;
end

if ~exist('scalefac'), scalefac = N^2; end

[psi, Effq] = velopt(psi0,src,kappa,q,L,scalefac);
```

```

fprintf(1,'Eff_%.d=%.f\n',q,Effq)

figure(1)
Nplot = 64; % Interpolate solution for plotting
psir = refine2(psi,Nplot); srcr = refine2(src,Nplot);
xplot = L*(0:Nplot-1)/Nplot; yplot = xplot';
imagesc(xplot,yplot,srcr), colorbar, hold on
contour(xplot,yplot,psir,10,'EdgeColor','k'), hold off

```

## 2 Program file velopt.m

```

function [psi,Effq] = velopt(psi0,src,kappa,q,L,scalefac)

% Problem parameters for Matlab's optimizer fmincon.
psi0 = psi0(:); problem.x0 = psi0(2:end);
problem.objective = @(x) normHq2(x,src,kappa,q,L,scalefac);
problem.nonlcon = @(x) nonlcon(x,src,kappa,q,L,scalefac);
problem.solver = 'fmincon';
problem.options = optimset('Display','iter','TolFun',1e-10,...
    'GradObj','on','GradConstr','on',...
    'algorithm','interior-point');

[psi,Hq2] = fmincon(problem);

% Mixing efficiency: call normHq2 with no flow to get pure-conduction solution.
Effq = sqrt(normHq2(zeros(size(psi)),src,kappa,q,L,scalefac) / Hq2);

psi = reshape([0;psi],size(src)); % Convert psi back into a square grid

%=====
function [varargout] = normHq2(psi,src,kappa,q,L,scalefac)

N = size(src,1); src = src(:);

% 2D Differentiation matrices and negative-Laplacian
[Dx,Dy,Dxx,Dyy] = Diffmat2(N,L); mlap = -(Dxx+Dyy);
if q ~= 0 && q ~= -1, error('This code only supports q = 0 or -1.');
```

```

end

psi = [0;psi]; ux = Dy*psi; uy = -Dx*psi;
ugradop = diag(sparse(ux))*Dx + diag(sparse(uy))*Dy;

if q == 0
    Aop2 = (-ugradop + kappa*mlap);
elseif q == -1
    Aop2 = mlap*(-ugradop + kappa*mlap);
end
Aop1 = (ugradop + kappa*mlap)*Aop2;
% Solve for chi, dropping corner point to fix normalisation.
chi = [0; Aop1(2:end,2:end) \ src(2:end)];
theta = Aop2*chi;

% The squared H^q norm of theta.
varargout{1} = L^2*sum(theta.^2)/N^2 * scalefac;

if nargout > 1
    % Gradient of squared-norm Hq2.
    gradHq2 = 2*((Dx*theta).(Dy*chi) - (Dy*theta).(Dx*chi));
    varargout{2} = gradHq2(2:end) / N^2 * scalefac;
end

%=====
function [c,ceq,gc,gceq] = nonlcon(psi,src,kappa,q,L,scalefac)

```

```

psi = [0;psi]; N = size(src,1);
c = []; gc = [];

[Dx,Dy,Dxx,Dyy] = Diffmat2(N,L); % 2D Differentiation matrices
U2 = L^2*(sum((Dx*psi).^2 + (Dy*psi).^2)/N^2);
ceq(1) = (U2-1) * scalefac;

if nargout > 2
    % Gradient of constraints
    mlappsi = -(Dxx+Dyy)*psi;
    gceq(:,1) = 2*mlappsi(2:end) / N^2 * scalefac;
end

```

## References

- [1] C. R. DOERING AND J.-L. THIFFEAULT, *Multiscale mixing efficiencies for steady sources*, Phys. Rev. E, 74 (2006), p. 025301(R).
- [2] S. PLASTING AND W. R. YOUNG, *A bound on scalar variance for the advection-diffusion equation*, J. Fluid Mech., 552 (2006), pp. 289–298.
- [3] T. A. SHAW, J.-L. THIFFEAULT, AND C. R. DOERING, *Stirring up trouble: Multi-scale mixing measures for steady scalar sources*, Physica D, 231 (2007), pp. 143–164.
- [4] J.-L. THIFFEAULT, *Using multiscale norms to quantify mixing and transport*, Nonlinearity, 25 (2012), pp. R1–R44.
- [5] J.-L. THIFFEAULT AND G. A. PAVLIOTIS, *Optimizing the source distribution in fluid mixing*, Physica D, 237 (2008), pp. 918–929.

## Lecture 31: Shuffling cards

[Aigner & Ziegler]

Many similarities with mixing. But exactly what is the connection?

Goal: defining shuffling method. How long before deck is shuffled "enough"?

Two warmup problems, which at first appear unrelated.

Birthday paradox:  $n$  people

prob. that they all have different birthdays?

(365 days a year, no seasonal effects)

two people:  $p(2) = 1 - \frac{1}{365}$

three people:  $p(3) = \left(1 - \frac{1}{365}\right)\left(1 - \frac{2}{365}\right)$

⋮

$$p(n) = \prod_{i=1}^{n-1} \left(1 - \frac{i}{365}\right)$$

We have  $p(n) < \frac{1}{2}$  for  $n = 23$ .



$n$  balls placed independently in  $K$  boxes.  
 prob that no box has  $> 1$  ball is

$$p(n, K) = \prod_{i=1}^{n-1} \left(1 - \frac{i}{K}\right)$$

### Coupon collector:

Take balls from a bowl with  $n$  distinguishable balls, put back each time

How many draws expected until you've drawn each ball at least once?

If you've drawn  $k$  distinct balls, then prob. of not getting a new one in next draw is  $\frac{k}{n}$ .

So prob. to need exactly  $s$  drawings for the next ball is:

$$\left(\frac{k}{n}\right)^{s-1} \left(1 - \frac{k}{n}\right)$$

Expected number of drawings for next ball is

$$\sum_{s \geq 1} \left(\frac{k}{n}\right)^{s-1} \left(1 - \frac{k}{n}\right) s$$

We can evaluate this:  $x = k/n$

$$\sum_{s \geq 1} x^{s-1} (1-x) s = \sum_{s \geq 1} x^{s-1} s - \sum_{s \geq 1} x^s s$$

$$= \sum_{s \geq 0} x^s (s+1) - \sum_{s \geq 0} x^s s$$

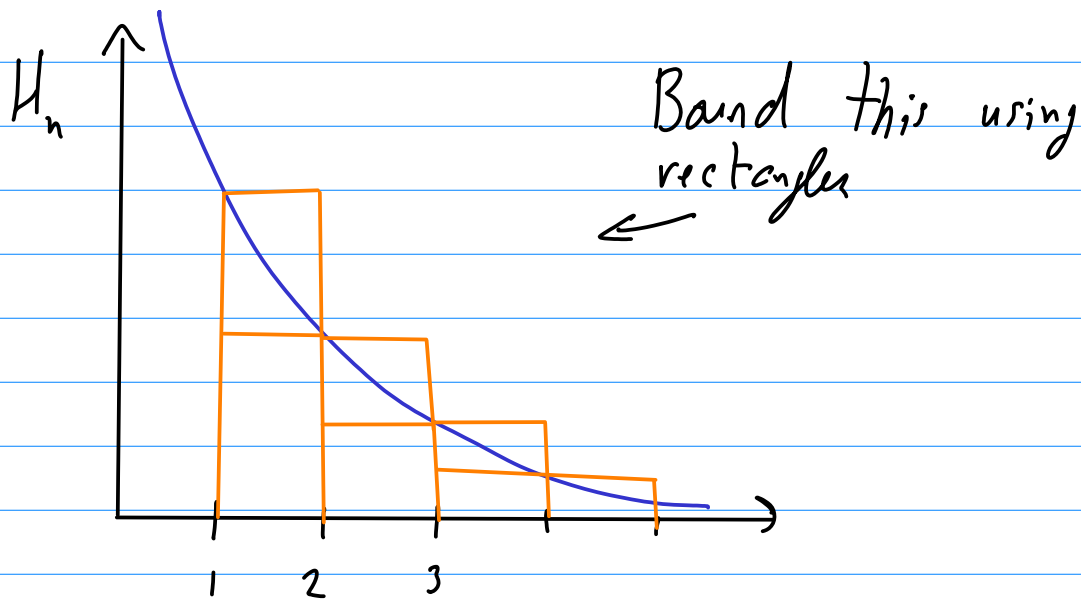
$$= \sum_{s \geq 0} x^s = \frac{1}{1-x}$$

Hence, 
$$\sum_{s \geq 1} \left(\frac{k}{n}\right)^{s-1} \left(1 - \frac{k}{n}\right) s = \frac{1}{1 - k/n}$$

The expected number of draws until we have drawn each of the  $n$  different balls is then

$$\begin{aligned} \sum_{k=0}^{n-1} \frac{1}{1 - \frac{k}{n}} &= \frac{n}{n} + \frac{n}{n-1} + \dots + \frac{n}{2} + \frac{n}{1} \\ &= n H_n \end{aligned}$$

$$H_n = \sum_{k=1}^n \frac{1}{k} \quad \text{Harmonic series}$$



$$H_n - 1 = \sum_{k=2}^n \frac{1}{k} < \int_1^n \frac{dt}{t} = \log n$$

$$H_n - \frac{1}{n} = \sum_{k=1}^{n-1} \frac{1}{k} > \int_1^n \frac{dt}{t} = \log n$$

$$\log n + \frac{1}{n} < H_n < \log n + 1$$

So  $H_n \approx \log n$ ,  $n \gg 1$ .

So the number of draws to get all the balls is

$$\approx n \log n$$

$V_n$  = number of drawings needed to get all  $n$  balls

$$E[V_n] \approx n \log n$$

$A_{i,m} = \{\text{ball } i \text{ not drawn in first } m \text{ drawings}\}$

$$\text{Prob}[V_n > m] = \text{Prob}\left[\bigcup_{i=1}^n A_{i,m}\right]$$

$$\leq \sum_{i=1}^n \text{Prob } A_{i,m} \quad \leftarrow \text{these are all equal}$$

$$= n \left(1 - \frac{1}{n}\right)^m \quad \leftarrow \text{we replace, so always } n \text{ balls}$$

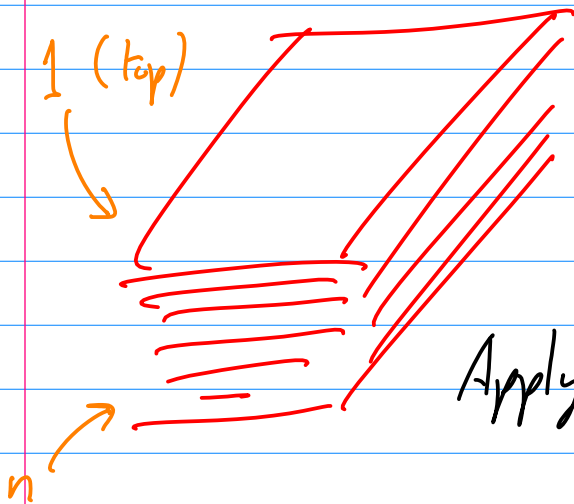
$$< n e^{-m/n}$$

[since  $(1 - \frac{1}{n})^n < \frac{1}{e}$  for all  $n \geq 1$ ]

let  $m = \lceil n \log n + cn \rceil$ . Then:

$$\text{Prob}[V_n > m] \leq e^{-c} \quad \leftarrow \text{probability that it takes "long"}$$

Deck of cards: number 1 to  $n$  in the order they first appear in the deck.  
(all different)



$S_n$  = symmetric (permutation) group.

What does it mean to shuffle?

Apply random permutations to deck.

Say we pick  $\pi \in S_n$  with equal prob.  $\frac{1}{n!}$ .  
Order after one shuffle is  $(\pi(1), \pi(2), \dots, \pi(n))$ .

Perfectly random!

But that's not how it's really done.

Only "certain" permutations occur.

example: top-in-at-random shuffle  
(not very effective, but easy to analyze)

Take the top card, insert at random in one of  $n$  places, with prob  $1/n$ .

So one of

$$\tau_i = (2, 3, \dots, i, 1, i+1, \dots, n) \quad 1 \leq i \leq n$$

is applied, each with prob  $1/n$ .

Expect this to take a long time, but how do we measure this?

"variation distance":

We look at the probability distribution of the  $n!$  different orderings of our deck.

Starting distribution  $I$ :

$$I(\text{id}) = 1$$

$$I(\pi) = 0, \text{ otherwise}$$

$$(\pi \in S_n)$$

Since the cards are unmixed

At the other extreme, we have the Uniform distribution:

$$U(\pi) = \frac{1}{n!}, \quad \pi \in S_n$$

Our goal is to see, after a certain number of shuffles, how "close" we are to the uniform distribution.

The variation distance between two probability distributions  $Q_1$  and  $Q_2$  is

$$\|Q_1 - Q_2\| = \frac{1}{2} \sum_{\pi \in S_n} |Q_1(\pi) - Q_2(\pi)|$$

## Lecture 32: Strong uniform stopping rules

The variation distance between two probability dists  $Q_1$  and  $Q_2$  is

$$\|Q_1 - Q_2\| = \frac{1}{2} \sum_{\pi \in S_n} |Q_1(\pi) - Q_2(\pi)|$$

Write  $Q_i(S) = \sum_{\pi \in S} Q_i(\pi)$ .

We have:  $\|Q_1 - Q_2\| = \max_{S \subseteq S_n} |Q_1(S) - Q_2(S)|$

proof: Let  $B = \{\pi \in S_n : Q_1(\pi) > Q_2(\pi)\}$ .

$$Q_1(A) - Q_2(A) \leq Q_1(A \cap B) - Q_2(A \cap B), \quad A \subseteq S_n$$

because  $\pi \in A \cap B^c$  satisfies  $Q_1(\pi) - Q_2(\pi) < 0$ , so difference in prob cannot decrease when such elements are eliminated. Then

$$Q_1(A) - Q_2(A) \leq Q_1(B) - Q_2(B)$$

since including more elements from  $B$  cannot decrease the prob. difference.



But by the same reasoning,

$$Q_2(A) - Q_1(A) \leq Q_2(B^c) - Q_1(B^c)$$

Difference in upper bounds:

$$\begin{aligned} & (Q_1(B) - Q_2(B)) - (Q_2(B^c) - Q_1(B^c)) \\ &= Q_1(B + B^c) - Q_2(B + B^c) = 1 - 1 = 0! \end{aligned}$$

Hence,  $Q_1(A) - Q_2(A) \leq C$

$$Q_2(A) - Q_1(A) \leq C$$

Now take  $A = B$ :  $Q_1(A) - Q_2(A) \leq C$  OK

Hence,  $A = B$  saturates the upper bound,

(Same for  $A = B^c$  for second inequality.)

Hence,

$$\begin{aligned} \max_{A \subseteq S_n} |Q_1(A) - Q_2(A)| &= \frac{1}{2} \left( \underbrace{Q_1(B) - Q_2(B)}_{>0} + \underbrace{Q_2(B^c) - Q_1(B^c)}_{>0} \right) \\ &= \frac{1}{2} \left[ \sum_{\pi \in B} (Q_1(\pi) - Q_2(\pi)) + \sum_{\pi \in B^c} |Q_1(\pi) - Q_2(\pi)| \right] \end{aligned}$$

since these are the same

$$= \frac{1}{2} \sum_{\pi \in S_n} |Q_1(\pi) - Q_2(\pi)|.$$



Recall the initial distribution  $I$  and the uniform distribution  $U$ .

At the start, we have:

$$\begin{aligned} \|I - U\| &= \frac{1}{2} \sum_{\pi \in S_n} |I(\pi) - U(\pi)| \\ &= \frac{1}{2} \underbrace{|I(\text{id})|}_1 - \underbrace{U(\text{id})}_{\frac{1}{n!}} + \frac{1}{2} \sum_{\substack{\pi \in S_n \\ \pi \neq \text{id}}} |U(\pi)| \end{aligned}$$

$$= \frac{1}{2} \left( 1 - \frac{1}{n!} \right) + \frac{1}{2} (n! - 1) \frac{1}{n!}$$

$$= 1 - \frac{1}{n!} \quad \leftarrow \text{very close to 1 (max distance)}$$

Now perform one top-in-at-random shuffle.  
We have:

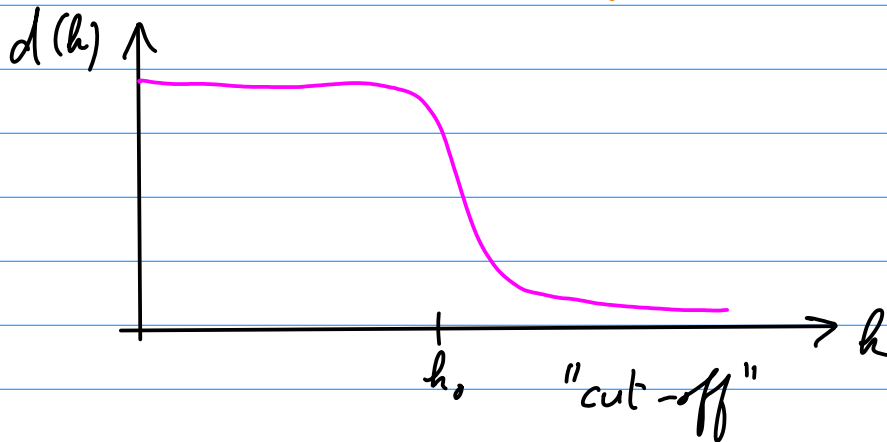
$$\text{Top}(\tau_i) = \frac{1}{n}, \text{ zero otherwise}$$

$$\tau_i = (2, 3, \dots, i, 1, i+1, \dots, n) \quad 1 \leq i \leq n,$$

$$\begin{aligned}
\|T_{\text{top}} - U\| &= \frac{1}{2} \sum_{\tau_i} \left| \frac{1}{n} - \frac{1}{n!} \right| + \frac{1}{2} \sum_{\pi \neq \tau_i} \frac{1}{n!} \\
&= \frac{1}{2} n \left( \frac{1}{n} - \frac{1}{n!} \right) + \frac{1}{2} (n! - n) \frac{1}{n!} \\
&= 1 - \frac{1}{(n-1)!} \quad \text{Not a big improvement!}
\end{aligned}$$

Let  $d(k) := \|T_{\text{top}}^{*k} - U\|$

$\uparrow$   $k$  times



Strong uniform stopping rules: [Aldous & Diaconis]

Observe permutations, then yell STOP! using some stopping rule. Strong uniform if  $\forall k \geq 0$

If the process is stopped after exactly  $k$  steps, then the resulting permutations of the deck have uniform distribution.

Let  $T$  be the number of steps performed until the stopping rule applies. ( $T$  is a random var.)

Ordering of deck after  $k$  shuffles is  $X_k \in S_n$ .

Stopping rule is strong uniform if

$$\text{Prob}[X_k = \pi \mid T = k] = \frac{1}{n!}, \quad \forall \pi \in S_n$$

example: Top-in-at-random.

STOP after the original bottom card ( $n$ ) is first inserted back into the deck.

is a strong uniform stopping rule.

This works because all the cards below card  $n$  are necessarily uniformly distributed.

Now let  $T_i = \#$  of shuffles until for the first time  $i$  cards lie below card  $n$ .

We have to determine the distribution of

$$T = T_1 + (T_2 - T_1) + \dots + (T_{n-1} - T_{n-2}) + (T - T_{n-1})$$

$T_i - T_{i-1}$  = time until top card is inserted at one of the  $i$  possible places below card  $n$ .

But this is the same as the time a canyon collector takes from canyon  $(n-i)$  to  $(n-i+1)$ !

Think of it as reverse canyon collecting: at first it will be unlikely to insert the top card below the  $n$ <sup>th</sup> one, which is like the collector trying to get his/her last canyon.

Thus  $T$  has the same distribution as  $V_n$ , and

$$\text{Prob}[T > h] \leq e^{-c}, \quad h = \lceil n \log n + cn \rceil$$

(from previous lecture)

Next lecture we will relate this to variation dist.

## Lecture 33: Riffle shuffling

Last time: for top-in-at-random shuffling, we showed

$$\text{Prob}[T > k] \leq e^{-c}$$

$$\text{for } k = \lceil n \log n + cn \rceil$$

$n$  = number of cards

$T$  = stopping time for strong uniform stopping rule.

Lemma: Let  $Q: S_n \rightarrow \mathbb{R}$  be a prob. distribution that defines a shuffling process  $Q^{*k}$  with a strong uniform stopping rule whose stopping time is  $T$ . Then

$$\|Q^{*k} - U\| \leq \text{Prob}[T > k] \quad \forall k \geq 0$$

proof:  $X$  random var. with values in  $S_n$ , prob dist.  $Q$ . For  $S \in S_n$ ,

$$U(S) = \text{Prob}[X \in S] = \frac{|S|}{n!}$$

← cardinality

$$\begin{aligned}
 Q^{*h}(S) &= \text{Prob}[X_h \in S] \\
 &= \sum_{j \leq h} \text{Prob}[X_h \in S \wedge T=j] \\
 &\quad + \text{Prob}[X_h \in S \wedge T > h]
 \end{aligned}$$

Recall:  $\text{Prob}[X_j \in S | T=j] = U(S)$

$$\text{Prob}[X_j \in S \wedge T=j] = U(S) \text{Prob}[T=j]$$

*↪ can replace by  $h$ , since  $h \geq j$*

$$\begin{aligned}
 Q^{*h}(S) &= \sum_{j \leq h} U(S) \text{Prob}[T=j] \\
 &\quad + \text{Prob}[X_h \in S | T > h] \text{Prob}[T > h]
 \end{aligned}$$

$$= U(S) (1 - \text{Prob}[T > h]) + \dots$$

$$= U(S) + \underbrace{(\text{Prob}[X_h \in S | T > h] - U(S))}_{| \cdot | \leq 1 \text{ (difference of probs.)}} \text{Prob}[T > h]$$

Hence,

$$|Q^{*h}(S) - U(S)| \leq \text{Prob}[T > h].$$



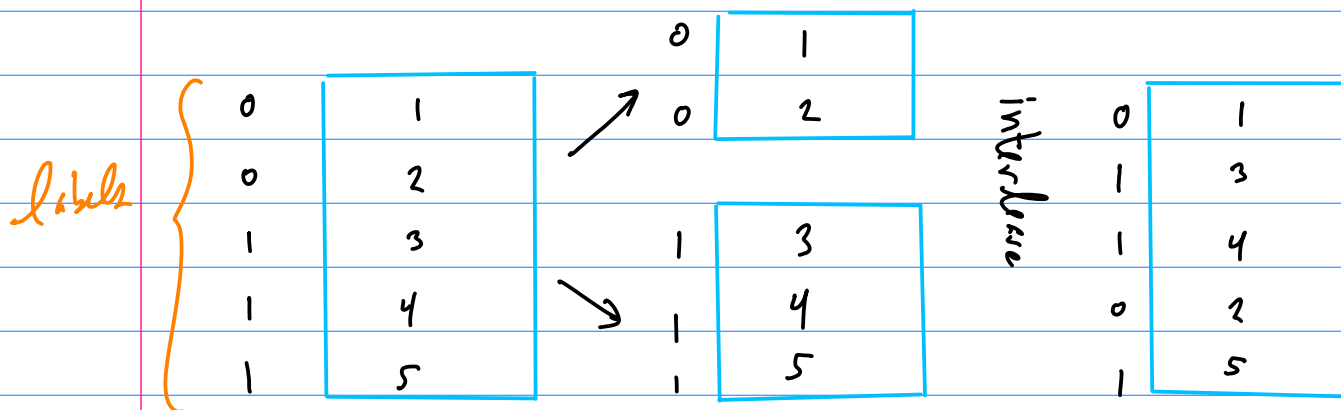
Conclude:  $d(h) = \| \text{Top}^{\times h} - U \| \leq e^{-c}$ .

Riffle shuffle: • split deck into two parts  
• interleave

Model: riffle shuffle consists of permutations  $\pi \in S_n$   
such that

$$(\pi(1), \pi(2), \dots, \pi(n))$$

consists of two interlaced increasing sequences.  
(except if it is the identity)



$t$  cards in right hand ( $0 \leq t \leq n$ )  
 $n-t$  " " left hand

$\binom{n}{t}$  ways to interleave

All distinct, except for identity.



total # of ways.  $\sum_{t=0}^n \binom{n}{t} = 2^n$

But  $n+1$  of these are the identity  $\rightarrow$  keep only one

$\Rightarrow 2^n - n$  distinct ways. (one is the identity)

What probability distribution to assign?

Gilbert & Shannon (1955) [simple]

Rif:  $S_n \rightarrow \mathbb{R}$ :

$$\text{Rif}(\pi) = \begin{cases} \frac{1}{2^n}(n+1) & , \quad \pi = \text{id} \\ \frac{1}{2^n} & , \quad \pi \text{ consists of two increasing sequences} \\ 0 & , \quad \text{otherwise} \end{cases}$$

Check:  $\sum_{\pi} \text{Rif}(\pi) = \frac{1}{2^n}(n+1) + \frac{1}{2^n}(2^n - n - 1) = 1$

Inverse shuffle: assign label 0 or 1 to each card randomly with prob.  $1/2$ .  
Move 0 cards to the top.

Inverse shuffling yields the same probability distribution as Rif.

For instance, inverse shuffling gives identity whenever all the 0 cards are already on top.

$$\frac{n+1}{2^n} \leftarrow \text{ways to have 0's on top (including none)}$$

$$2^n \leftarrow \text{total \# of configs}$$

which is the same as Rif(id).

$$\text{Show: } \|\text{Rif}^{x^h} - U\| \leq 1 - \prod_{i=1}^{n-1} \left(1 - \frac{i}{2^h}\right)$$

proof: Analyze inverse shuffles instead:

$$\overline{\text{Rif}(\pi)} := \text{Rif}(\pi^{-1})$$

Since every perm has unique inverse, and  $U(\pi) = U(\pi^{-1})$ :

$$\|\text{Rif}^{x^h} - U\| = \|\overline{\text{Rif}^{x^h}} - U\|$$

In every inverse riffle shuffle, each card gets a  $d$ -bit; remember these digits.

Stopping rule:

STOP as soon as all cards have distinct strings.

The cards are sorted according to the binary numbers

$$b_h b_{h-1} \dots b_2 b_1 \quad b_i = i^{\text{th}} \text{ shuffle}$$

But since these bits are independent and random, the order of the deck must then be random!  
 → stopping rule is strong uniform.

But how long does this take?

$$K = 2^h \text{ configurations (boxes)}$$

Put 2 cards in the same box if they have the same label  $b_h b_{h-1} \dots b_2 b_1$ .

What is the probability that some box gets more than one card?

Birthday paradox!

$$\text{Prob}[T > h] = 1 - \prod_{i=1}^{n-1} \left(1 - \frac{i}{2^h}\right)$$

and this bounds the variation distance. 1/19

How many shuffles do we need? ( $n$  large)

$$\text{let } h = 2 \log_2(cn), \quad c \geq 1$$

$$1 - \frac{i}{2^h} = 1 - \frac{i}{(cn)^2}$$

$$\sum_{i=1}^{n-1} \log \left(1 - \frac{i}{(cn)^2}\right) \approx \sum_{i=1}^{n-1} \frac{-i}{(cn)^2}$$

$$= \frac{-1}{(cn)^2} \frac{1}{2} n(n-1) \approx -\frac{1}{2c^2}$$

Hence,

$$\text{Prob}[T > h] \approx 1 - e^{-1/2c^2} \approx \frac{1}{2c^2}$$

We get from this

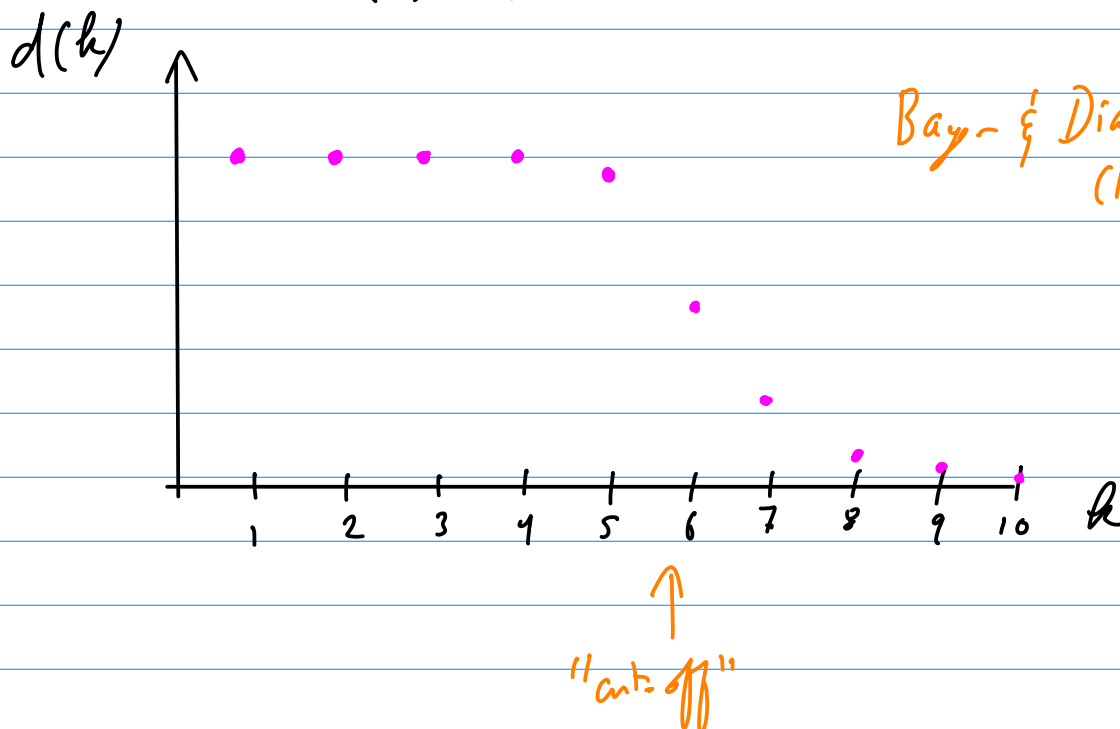
$$d(12) \leq 0.28$$

This is actually a pretty bad bound.  
In practice,

$$d(6) \leq 0.614$$

$$d(7) \leq 0.334 \quad \leftarrow \text{magic \# of shuffles}$$

$$d(8) \leq 0.167$$



## Lecture 34: Permutations generated by Brownian particles

### 1 Brownian particles on the real line

Consider  $n$  Brownian particles on the real line, with diffusion constant  $D$ . The position of each particle is denoted  $x_k(t)$ . The position of a walker at time  $t$  has a probability density function  $p(x, t; x', 0)$  that satisfies the heat equation,

$$\partial_t p - D\Delta p = \delta(x - x') \delta(t), \quad (1)$$

where  $x'$  is the initial position of the particle, with solution

$$p(x, t; x', 0) = \frac{1}{\sqrt{4\pi Dt}} e^{-(x-x')^2/4Dt}, \quad t > 0. \quad (2)$$

Assume the initial ordering

$$x_1(0) < x_2(0) < \cdots < x_n(0), \quad (3)$$

and define the probability  $P(t, s)$  that the particles are ordered according to the permutation  $s \in S_n$ , the symmetric group on  $n$  symbols; thus,

$$P(0, s) = \begin{cases} 1, & s = \text{id}; \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

At later times, we have

$$P(t, s^{-1}) = \int_{-\infty}^{\infty} dx_{s(1)} \int_{x_{s(1)}}^{\infty} dx_{s(2)} \cdots \int_{x_{s(n-1)}}^{\infty} dx_{s(n)} \prod_{k=1}^n p(x_{s(k)}, t; x_{s(k)}(0), 0). \quad (5)$$

That is, the particle  $s(1)$  is to the left of all the others,  $s(2)$  is to the right of  $s(1)$  but to the left of the rest, etc.

Now let

$$y_k := x_{s(k)}/\sqrt{4Dt}, \quad \varepsilon_k(t) := x_{s(k)}(0)/\sqrt{4Dt}, \quad (6)$$

from which (5) becomes

$$P(t, s^{-1}) = \int_{-\infty}^{\infty} dy_1 \int_{y_1}^{\infty} dy_2 \cdots \int_{y_{n-1}}^{\infty} dy_n \prod_{k=1}^n \frac{1}{\sqrt{\pi}} e^{-(y_k - \varepsilon_k(t))^2}. \quad (7)$$

For large time, we have  $\varepsilon_k(t) \ll 1$ , and

$$e^{-(y_k - \varepsilon_k(t))^2} = e^{-y_k^2} (1 + 2\varepsilon_k y_k) + O(\varepsilon^2). \quad (8)$$

We must have

$$\frac{1}{\pi^{n/2}} \int_{-\infty}^{\infty} dy_1 \int_{y_1}^{\infty} dy_2 \cdots \int_{y_{n-1}}^{\infty} dy_n e^{-\sum_{k=1}^n y_k^2} = \frac{1}{n!}. \quad (9)$$

(This must work for any PDF with the right symmetry, so must be the fraction of volume occupied by an  $n$ -dimensional ‘wedge.’)

Define

$$I_{n,\ell} := -\frac{1}{\pi^{n/2}} \int_{-\infty}^{\infty} dy_1 \int_{y_1}^{\infty} dy_2 \cdots \int_{y_{n-1}}^{\infty} dy_n y_\ell e^{-\sum_{k=1}^n y_k^2}. \quad (10)$$

By changing order of integration and replacing  $y_k$  by  $-y_{n-k+1}$ , we can show  $I_{n,\ell} = -I_{n,n-\ell+1}$ . Some specific values are  $I_{2,1} = -I_{2,2} = 1/(2\sqrt{2\pi})$ ,  $I_{3,1} = -I_{3,3} = 1/(4\sqrt{2\pi})$ ,  $I_{3,2} = 0$ . Challenge: compute this in general (must be known...).

In any case, the time-asymptotic solution is

$$P(t, s^{-1}) = \frac{1}{n!} - \sum_{k=1}^n 2\varepsilon_k I_{n,k} + O(\varepsilon^2), \quad (11)$$

which, from (6), shows a rather slow approach to the uniform distribution as  $1/\sqrt{t}$ .

## 2 Brownian particles on the unit interval

Now we turn to Brownian particles on the interval  $[0, 1]$ , with reflecting boundary conditions at the endpoints (Figure 1). The same heat equation (2) is satisfied by the

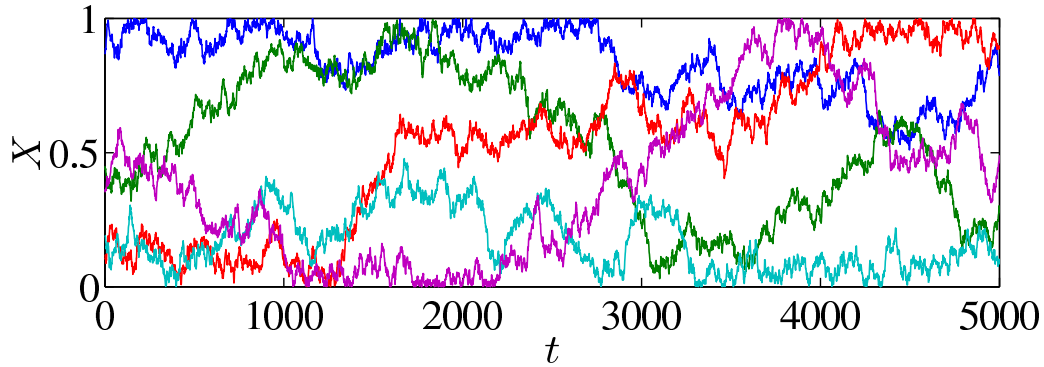


Figure 1: Five Brownian particles on the interval  $[0, 1]$ , with reflecting boundary conditions.

probability density (Green’s function), but now the reflecting (Neumann) boundary conditions lead to

$$p(x, t; x', 0) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{4\pi Dt}} \left( e^{-(x-x'-2k)^2/4Dt} + e^{-(x+x'-2k)^2/4Dt} \right), \quad t > 0, \quad (12)$$

which can also be written

$$p(x, t; x', 0) = \sum_{k=-\infty}^{\infty} \cos(\pi kx) \cos(\pi kx') e^{-(\pi k)^2 Dt}, \quad t > 0. \quad (13)$$

The same formula (5) applies for  $P(t, s^{-1})$ . Let’s take  $n = 2$ ; then after some integrals

$$P(t, s^{-1}) = \frac{1}{2} + \sum'_{\substack{k, \ell \\ k \neq \ell}} \frac{(1 - (-1)^{k+\ell})}{\pi^2(k^2 - \ell^2)} \cos(\pi kx_{s(1)}(0)) \cos(\pi \ell x_{s(2)}(0)) e^{-\pi^2(k^2 + \ell^2)Dt}. \quad (14)$$

( $\sum'$  means  $k \neq 0$  and  $\ell \neq 0$ .) The slowest exponential has  $k^2 + \ell^2 = 1$ ; hence, we have

$$\left| P(t, s^{-1}) - \frac{1}{2} \right| \leq |C| e^{-\pi^2 Dt} \quad (15)$$

where

$$C = \sum'_{\substack{k, \ell \\ k \neq \ell}} \left| \frac{(1 - (-1)^{k+\ell})}{\pi^2(k^2 - \ell^2)} \right|, \quad (16)$$

as long as  $C$  is finite. (Challenge: does this diverge? If so need to refine the analysis.)

In any case it appears to be the right bound: see Fig. 2. Is this a cut-off? How do we show this?



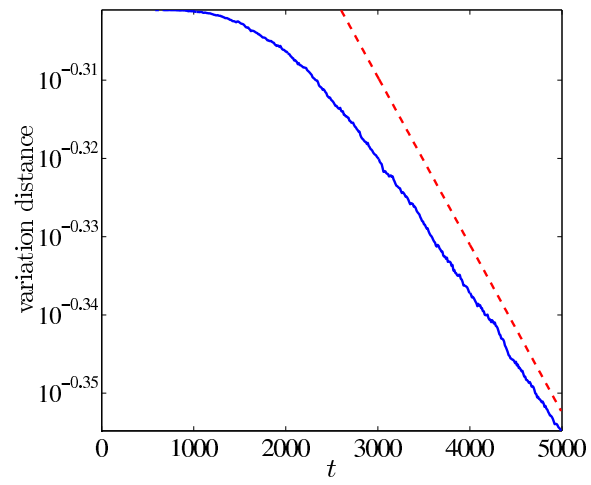


Figure 2: Variation distance as a function of time for 2 particles, with  $D = 5 \times 10^{-6}$  (100,000 realizations). The dashed line is proportional to  $e^{-\pi^2 Dt}$ .

## Lecture 35: Random braids and winding numbers

### 1 Random walks on Cayley graphs

A physical braid is a collection of strands that are anchored at both ends.

$$\sigma_i \sigma_j = \sigma_j \sigma_i \quad \text{for } |i - j| > 1; \quad \sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_j \quad \text{for } |i - j| = 1. \quad (1)$$

A convenient representation of a group is given by its *Cayley graph*. Figure 2(a) shows part of the Cayley graph for  $B_3$ : the identity is at the center, and moving by one step north, south, east, or west in the graph corresponds to multiplication by  $\sigma_1$ ,  $\sigma_1^{-1}$ ,  $\sigma_2$ , or  $\sigma_2^{-1}$ , respectively. At first glance, the Cayley graph would appear to be a tree, but the braid group relations (1) imply that we sometimes get the same element along different branches, or even return to the identity (since, for example,  $\sigma_1 \sigma_2 \sigma_1 \sigma_2^{-1} \sigma_1^{-1} \sigma_2^{-1} = e$ ). The graph thus has a very complicated topology.

A random walk on a Cayley graph is defined by starting from some group element and repeatedly moving in a random direction, typically with equal probability, to generate a random sequence of generators. We can then ask typical questions, such

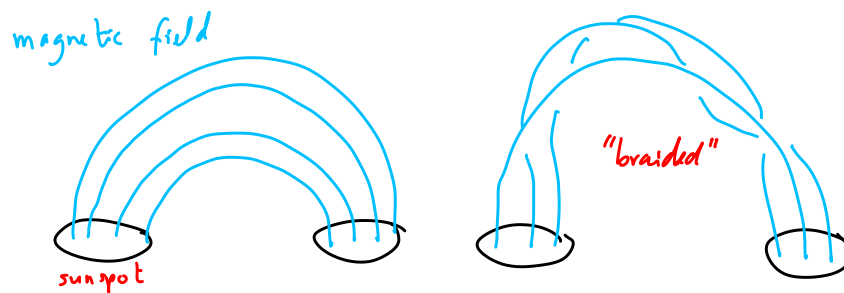
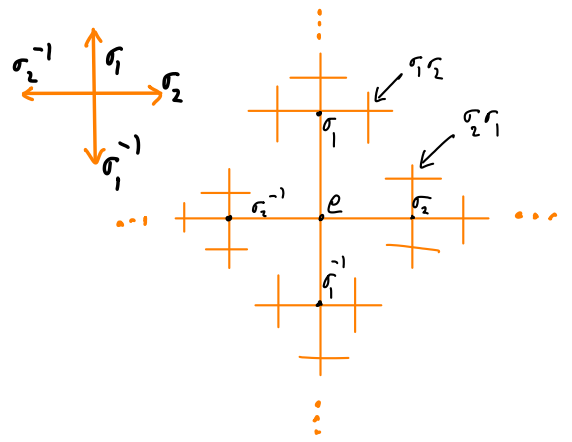
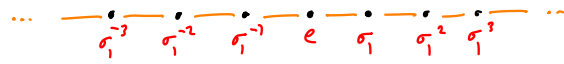


Figure 1: Magnetic flux tubes.



(a)



(b)

Figure 2: (a) Cayley graph for  $B_3$ . (b) Cayley graph for  $B_2$ .

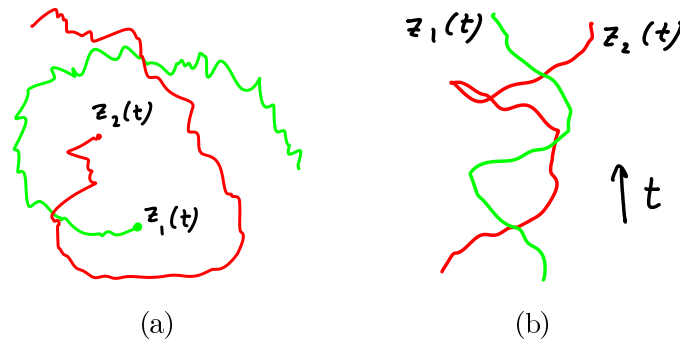


Figure 3: (a) Two Brownian particles winding around each other. (b) Lifted trajectories, with ‘time’ running vertically.

as the probability of recurrence, or if the walk is not recurrent we can ask about the asymptotic distance from our starting point. Random walks on Cayley graphs are also used to generate braids to study polymer entanglement, and their ‘complexity’ is gauged by the asymptotic behavior of a knot polynomial associated with the closure of the braid (Nechaev, 1996).

By far the simplest graph, though, is that of  $B_2$ , depicted in Figure 2(b). Here we can only twist the two strands in one direction or the other, so the braid after  $N$  steps in the graph is  $\sigma_1^m$ , where  $m$  is a random variable with binomial distribution

$$P_N(m) = \binom{N}{(N+m)/2} p^{(N+m)/2} (1-p)^{(N-m)/2}, \quad m + N \text{ even}, \quad (2)$$

and  $P_N(m) = 0$  if  $m + N$  is odd. Here  $p$  is the probability of moving right in the graph, and  $(1-p)$  is the probability of moving left. The mean of  $m$  is  $N(p - \frac{1}{2})$ , and its variance is  $Np(1-p)$ . For large  $N$ ,  $P_N(m)$  will converge to a normal distribution.

## 2 Braid of two Brownian particles

Let us formulate a somewhat more ‘physical’ version of the random walk on  $B_2$  discussed at the end of the previous section. Consider two Brownian particles on the plane,  $z_1(t)$  and  $z_2(t)$ , each with diffusion constant  $D$  (Figure 3(a)). We can regard these as a braid with two strands by plotting the trajectories with time as a vertical axis, as in Figure 3(b). How is the resulting random braid distributed for large time?

For two strands all that matters is the winding angle of one particle around the other. We consider the vector  $z(t) = z_1(t) - z_2(t)$ , which behaves like a Brownian

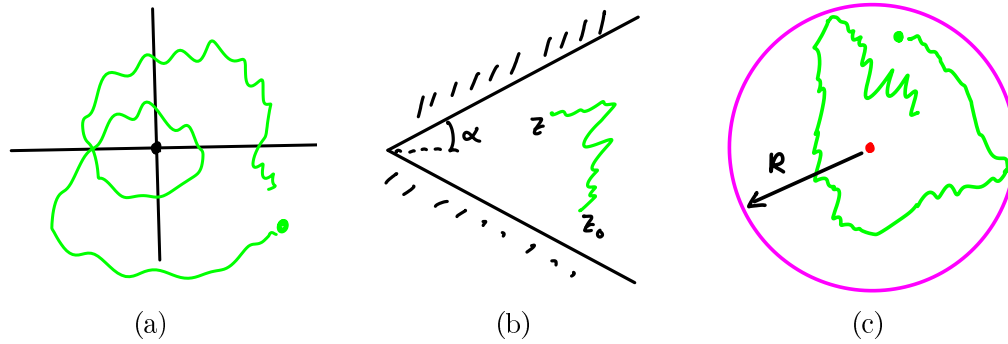


Figure 4: (a) A single Brownian particle winding around the origin. (b) A particle starting at  $z_0$  and ending at  $z$  in a wedge-shaped domain. (c) Particle in a disk of radius  $R$ , with reflecting boundaries.

particle of diffusivity  $2D$ . We define  $\theta \in (-\infty, \infty)$  to be the total winding angle of  $z(t)$  around the origin. The geometrical relationship between  $m$  from Section 1 and  $\theta$  is  $\theta = \pi m$ , though their distributions are very different. The time-asymptotic distribution of  $\theta$  is given by the classical Spitzer formula (Spitzer, 1958),

$$P(x) = \frac{1}{\pi} \frac{1}{1+x^2}, \quad x := \frac{2\theta}{\log(4Dt/r_0^2)}, \quad 4Dt/r_0^2 \gg 1, \quad (3)$$

where  $r_0 = |z(0)|^2$ . This result can be derived from conformal invariance of the Brownian process (Drossel & Kardar, 1996), or by solving the heat equation in polar coordinates in the interior of a wedge-shaped domain (Fig. 4(b)), then taking the wedge angle to infinity (Edwards, 1967; Redner, 2001). Going around the origin then corresponds to moving to a different Riemann sheet. See Appendix A for a derivation.

Equation 3 is a Cauchy–Lorentz distribution, which has mean zero but an infinite variance (all even moments diverge; the odd moments are zero by symmetry if the integrals are interpreted as principal values). This peculiarity can be traced to the ‘scale-free’ nature of a Brownian process: since the particle motion is rough at all scales, if it comes near the origin it can wind an infinite number of times around the origin in finite time.

A straightforward numerical simulation of a Brownian particle yields the winding angle distribution in Fig. 6(b). Also shown is the Spitzer formula (3); the agreement is good for small  $\theta$ , but the tails are off by a wide margin. The discrepancy is due to the numerical difficulty of recreating a true Brownian process: we cannot have an

unbounded number of windings in a small time. Thus, the observed distribution has exponential tails rather than a power-law. B elisle (1989) derived the distribution for a small (but finite) step size:

$$P(x) = \frac{1}{2} \operatorname{sech}(\pi x/2), \quad 4Dt/r_0^2 \gg 1, \quad (4)$$

with  $x$  defined as in (3). Notice that the step size does not enter the formula (for large time). The same distribution applies to a Brownian process winding around a small disk centered on the origin, even as the disk radius is taken to zero. Most regularizations that prevent infinite winding around the origin will turn (3) into (4). The latter is this the proper ‘physical’ distribution, since physical entities such as magnetic field lines are not true Brownian processes.

## A Derivation of Spitzer’s law

Consider the two-dimensional diffusion equation

$$\frac{\partial v}{\partial t} - D\Delta v = \frac{1}{r_0} \delta(r - r_0) \delta(\theta - \theta_0) \delta(t) \quad (5)$$

for the Green’s function  $v(r, \theta, t)$ . This equation has Laplace transform

$$s\bar{v} - D\Delta\bar{v} = \frac{1}{r_0} \delta(r - r_0) \delta(\theta - \theta_0) \quad (6)$$

where  $\bar{v}(r, \theta, s)$  is the Laplace transform of  $v(r, \theta, t)$ . The free space Green’s function is

$$v(x, y, t | x_0, y_0, 0) = \frac{1}{4\pi Dt} e^{-R^2/4Dt} \quad (7)$$

with Laplace transform

$$\bar{v}(x, y, s | x_0, y_0) = \frac{1}{2\pi D} K_0(Rq) \quad (8)$$

where  $K_0$  is a modified Bessel function of the second kind, and

$$q = \sqrt{s/D}, \quad R^2 = (x - x_0)^2 + (y - y_0)^2 = r^2 + r_0^2 - 2rr_0 \cos(\theta - \theta_0). \quad (9)$$

We wish to find the Green’s function in an infinite domain, but over multiple Riemann sheets. We write  $v = u + w$  and solve for  $w$ , which satisfies the homogeneous equation

$$\frac{\partial w}{\partial t} - D\Delta w = 0, \quad \text{or} \quad s\bar{w} - D\Delta\bar{w} = 0, \quad (10)$$

and vanishes at  $t = 0$ . Assume the separable form

$$\bar{w} = \mathcal{R}(r)\mathcal{T}(\theta). \quad (11)$$

Then using

$$\Delta\bar{w} = \frac{\partial^2\bar{w}}{\partial r^2} + \frac{1}{r}\frac{\partial\bar{w}}{\partial r} + \frac{1}{r^2}\frac{\partial^2\bar{w}}{\partial\theta^2} \quad (12)$$

we find

$$\Delta\bar{w} = \mathcal{T}\mathcal{R}'' + \frac{\mathcal{T}\mathcal{R}'}{r} + \frac{\mathcal{R}\mathcal{T}''}{r^2} = q^2\mathcal{R}\mathcal{T} \quad (13)$$

or

$$\frac{r^2\mathcal{R}''}{\mathcal{R}} + \frac{r\mathcal{R}'}{\mathcal{R}} - r^2q^2 = -\frac{\mathcal{T}''}{\mathcal{T}} = \nu^2. \quad (14)$$

The angular part is

$$\mathcal{T}'' + \nu^2\mathcal{T} = 0, \quad (15)$$

with independent solutions

$$\mathcal{T} = \cos\nu\theta \quad \text{and} \quad B \sin\nu\theta, \quad (16)$$

whilst the radial part satisfies Bessel's equation

$$r^2\mathcal{R}'' + r\mathcal{R}' - (q^2r^2 + \nu^2)\mathcal{R} = 0. \quad (17)$$

The solutions are

$$\mathcal{R} = K_\nu(qr) \quad \text{and} \quad I_\nu(qr). \quad (18)$$

The solution that is regular at the origin, vanishes at  $\infty$ , and is continuous at  $r = r_0$  is

$$\mathcal{R} = K_\nu(qr_>) I_\nu(qr_<) \quad (19)$$

where  $r_> = r$  for  $r > r_0$ ,  $r_> = r_0$  for  $r < r_0$  (vice versa for  $r_<$ ). Now use this formula from Carslaw & Jaeger (1959, eq. 14.14(1)):

$$K_0(Rq) = P \int_{-i\infty}^{i\infty} \frac{\cos\nu(\pi - \theta + \theta_0)}{\sin\nu\pi} K_\nu(qr) I_\nu(qr_0) i d\nu, \quad r > r_0, \quad (20)$$

where  $P$  denotes the principal value at the origin, which suggests we write  $\bar{w}$  as

$$\bar{w} = \frac{1}{2\pi D} P \int_{-i\infty}^{i\infty} (A(\nu) \cos\nu\theta + B(\nu) \sin\nu\theta) K_\nu(qr) I_\nu(qr_0) i d\nu \quad (21)$$

and choose  $A(\nu)$  and  $B(\nu)$  to satisfy the boundary conditions. The Green's function  $\bar{v} = \bar{u} + \bar{w}$  is then

$$\bar{v} = \frac{1}{2\pi D} P \int_{-i\infty}^{i\infty} \left( A(\nu) \cos \nu\theta + B(\nu) \sin \nu\theta + \frac{\cos \nu(\pi - \theta + \theta_0)}{\sin \nu\pi} \right) K_\nu(qr) I_\nu(qr_0) i d\nu. \quad (22)$$

Let's try to satisfy the zero condition at  $\theta = 0$  and  $\theta = \gamma$ , as in C&J. At  $\theta = \gamma$ ,

$$\bar{v} = \frac{1}{2\pi D} P \int_{-i\infty}^{i\infty} \left( A(\nu) \cos \nu\gamma + B(\nu) \sin \nu\gamma + \frac{\cos \nu(\pi - \gamma + \theta_0)}{\sin \nu\pi} \right) K_\nu(qr) I_\nu(qr_0) i d\nu = 0 \quad (23)$$

so

$$B(\nu) = -\frac{A(\nu) \sin \nu\pi \cos \nu\gamma + \cos \nu(\pi - \gamma + \theta_0)}{\sin \nu\pi \sin \nu\gamma}. \quad (24)$$

At  $\theta = 0$ ,

$$\bar{v} = \frac{1}{2\pi D} P \int_{-i\infty}^{i\infty} \left( A(\nu) + \frac{\cos \nu(\pi + \theta_0)}{\sin \nu\pi} \right) K_\nu(qr) I_\nu(qr_0) i d\nu = 0. \quad (25)$$

We need to be careful here: solving for  $A(\nu)$  to make the expression in parentheses vanish is possible but leads to  $\bar{w} = -\bar{u}$ , so  $\bar{v} = 0$ . What we have to do is to choose  $A(\nu)$  to remove the pole in the denominator when  $\theta = 0$ :

$$A(\nu) = -\frac{\cos \nu(\pi - \theta_0)}{\sin \pi\nu}. \quad (26)$$

After combining those two expressions, we find

$$\bar{v} = -\frac{1}{\pi D} \int_{-i\infty}^{i\infty} \frac{\sin \nu(\gamma - \theta) \sin \nu\theta_0}{\sin \nu\gamma} K_\nu(qr) I_\nu(qr_0) i d\nu. \quad (27)$$

where we removed the principal value since the integral is now regular at  $\nu = 0$ . This is the 2D version of 14.14(3) in C&J. We evaluate the integral by closing the contour on the right and summing the residues. We obtain the residues from

$$\frac{\sin \nu(\gamma - \theta)}{\sin \nu\gamma} \sim \frac{\sin \nu_k(\gamma - \theta)}{(-1)^k \gamma(\nu - \nu_k)} = -\frac{\sin \nu_k\theta}{\gamma(\nu - \nu_k)}, \quad \nu \text{ near } \nu_k = \pi k/\gamma, \quad (28)$$

which leads to

$$\bar{v} = \frac{2}{\gamma D} \sum_{k=1}^{\infty} \sin(\nu_k\theta) \sin(\nu_k\theta_0) K_{\nu_k}(qr) I_{\nu_k}(qr_0), \quad \nu_k = \pi k/\gamma, \quad (29)$$



valid for  $r > r_0$  and  $0 \leq \theta \leq \gamma$ . We can use the symmetry of the Green's function to write

$$\bar{v} = \frac{2}{\gamma D} \sum_{k=1}^{\infty} \sin(\nu_k \theta) \sin(\nu_k \theta_0) K_{\nu_k}(qr_>) I_{\nu_k}(qr_<), \quad 0 \leq \theta \leq \gamma, \quad (30)$$

valid for  $0 < r < \infty$ . Using formula 22 from Appendix V of C&J, we can invert the Laplace transform to obtain finally

$$v = \frac{1}{\gamma Dt} e^{-(r^2+r_0^2)/4Dt} \sum_{k=1}^{\infty} \sin(\nu_k \theta) \sin(\nu_k \theta_0) I_{\nu_k}\left(\frac{rr_0}{2Dt}\right) \quad 0 \leq \theta \leq \gamma. \quad (31)$$

We can center the wedge so that it extends between  $-\gamma/2 \leq \theta \leq \gamma/2$  by replacing  $\theta$  by  $\theta + \gamma/2$ :

$$v = \frac{1}{\gamma Dt} e^{-(r^2+r_0^2)/4Dt} \left( \sum_{k \text{ even}}^{\infty} \sin(\nu_k \theta) \sin(\nu_k \theta_0) + \sum_{k \text{ odd}}^{\infty} \cos(\nu_k \theta) \cos(\nu_k \theta_0) \right) I_{\nu_k}\left(\frac{rr_0}{2Dt}\right), \quad (32)$$

valid for  $-\gamma/2 \leq \theta, \theta_0 \leq \gamma/2$ . This may appear more cumbersome but now we can take the limit  $\gamma \rightarrow \infty$  while keeping  $\theta$  and  $\theta_0$  finite. The difference between two successive  $\nu_k$  is

$$d\nu_k = \nu_{k+2} - \nu_k = 2\pi/\gamma. \quad (33)$$

Thus, in the limit as  $\gamma \rightarrow \infty$ ,

$$v = \frac{1}{2\pi Dt} e^{-(r^2+r_0^2)/4Dt} \int_0^{\infty} (\sin(\nu\theta) \sin(\nu\theta_0) + \cos(\nu\theta) \cos(\nu\theta_0)) I_{\nu}\left(\frac{rr_0}{2Dt}\right) d\nu. \quad (34)$$

or

$$v = \frac{1}{2\pi Dt} e^{-(r^2+r_0^2)/4Dt} \int_0^{\infty} \cos \nu(\theta - \theta_0) I_{\nu}\left(\frac{rr_0}{2Dt}\right) d\nu. \quad (35)$$

This form is equivalent to (see Duffy (2001, p. 215))

$$v = \frac{1}{\pi} \int_0^{\infty} \cos \nu(\theta - \theta_0) \int_0^{\infty} \alpha J_{\nu}(\alpha r) J_{\nu}(\alpha r_0) e^{-D\alpha^2 t} d\alpha d\nu. \quad (36)$$

At  $t = 0$  we can use (see <http://functions.wolfram.com/Bessel-TypeFunctions/BesselJ/21/02/02/>)

$$\int_0^{\infty} \alpha J_{\nu}(\alpha r) J_{\nu}(\alpha r_0) d\alpha = \frac{1}{r} \delta(r - r_0), \quad \frac{1}{\pi} \int_0^{\infty} \cos \nu(\theta - \theta_0) d\nu = \delta(\theta - \theta_0) \quad (37)$$

to explicitly recover the delta-function initial condition from (36). We can also show that the normalization is preserved. Reversing the role of  $\alpha$  and  $r$ ,  $\nu$  and  $\theta$  in (37), we have

$$\int_0^\infty r J_0(\alpha r) dr = \int_0^\infty r J_0(\alpha r) J_0(0r) dr = \frac{1}{\alpha} \delta(\alpha), \quad (38)$$

and

$$\frac{1}{\pi} \int_{-\infty}^\infty \cos \nu(\theta - \theta_0) d\theta = 2\delta(\nu), \quad (39)$$

but note that  $\int_0^\infty \delta(\nu) d\nu = 1/2$ , so the factor of 2 in (39) cancels out upon integration. Together these give

$$\begin{aligned} \int_0^\infty \int_{-\infty}^\infty r \nu d\theta dr &= \int_0^\infty \int_0^\infty \alpha r J_0(\alpha r) J_0(\alpha r_0) e^{-D\alpha^2 t} d\alpha dr \\ &= \int_0^\infty \delta(\alpha) J_0(\alpha r_0) e^{-D\alpha^2 t} d\alpha = 1. \end{aligned}$$

Let's rewrite the PDF (35) in terms of the scaled variables  $x = r/2\sqrt{Dt}$ ,  $y = r_0/2\sqrt{Dt}$ , and write  $\theta$  for  $\theta - \theta_0$  without loss of generality:

$$p = \frac{2}{\pi} e^{-(x^2+y^2)} \int_0^\infty \cos \nu \theta I_\nu(2xy) d\nu. \quad (40)$$

Since  $2xy$  is small, we use the asymptotic form

$$I_\nu(x) \sim \frac{1}{\Gamma(\nu+1)} (x/2)^\nu \quad (41)$$

and the integral

$$\int_0^\infty \cos \nu \theta \xi^\nu d\nu = -\frac{\log \xi}{\theta^2 + \log^2 \xi}, \quad 0 < \xi < 1 \quad (42)$$

to obtain

$$p \simeq -\frac{2}{\pi} \frac{\log(xy)}{\theta^2 + \log^2(xy)} e^{-(x^2+y^2)}. \quad (43)$$

The most interesting thing to us is the total probability of reaching a certain angle  $\theta$  for any  $x$ , so we integrate over  $x$  using  $\log x \ll \log y$ , since  $y$  is very small whereas  $x$  varies from 0 to  $\infty$ . Thus,

$$\int_0^\infty x p dx \approx \frac{1}{\pi} \frac{-\log y}{\theta^2 + \log^2 y}, \quad y = \frac{r_0}{2\sqrt{Dt}} \ll 1, \quad (44)$$

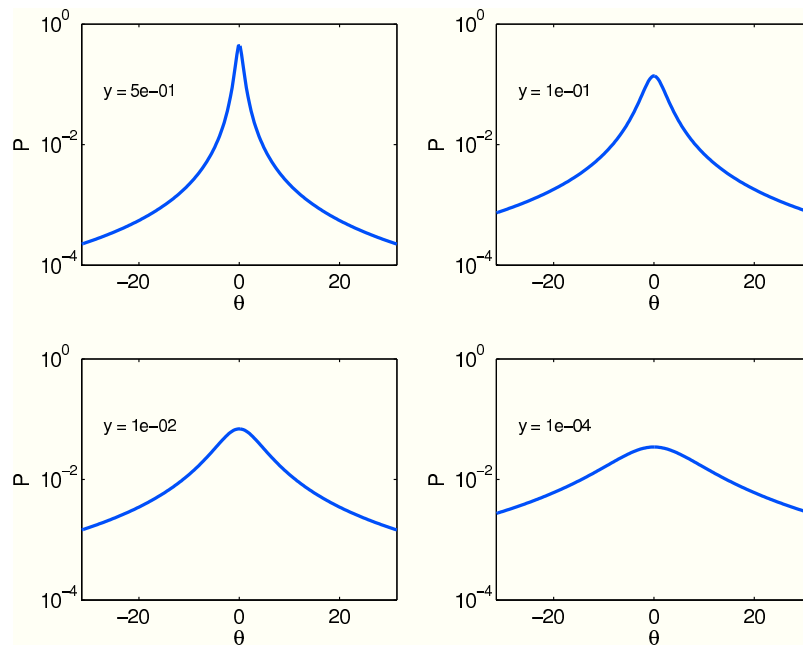


Figure 5: The Cauchy-Lorentz distribution (44) for various  $y$  values.

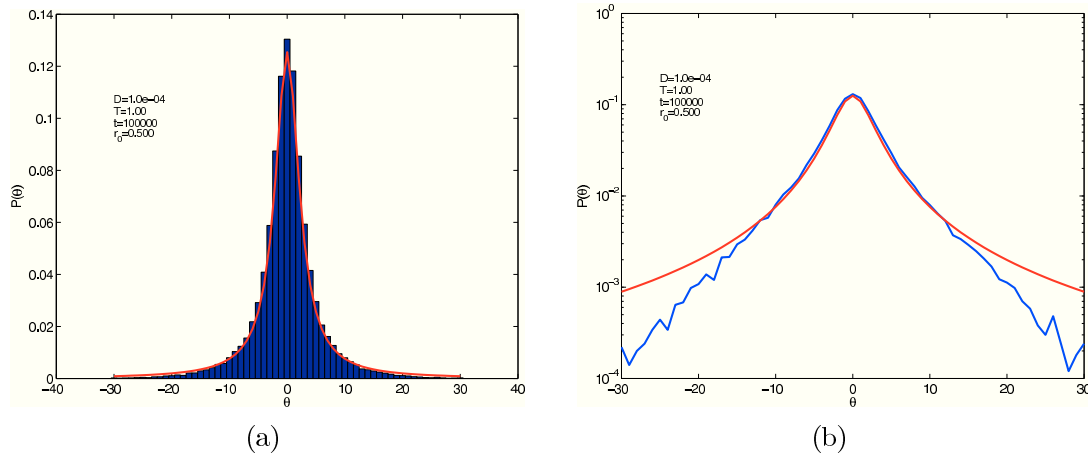


Figure 6: Spitzer distribution compared to numerical simulation.

where we also approximated  $e^{-y^2} \simeq 1$ , to obtain a consistent normalization in  $\theta$ .

The PDF (44) is a Cauchy–Lorentz distribution in  $\theta$  (see Fig. 5). This is the probability of finding the particle at  $\theta$  at time  $t$ , given that it started at  $\theta = 0$  and radius  $r_0$ . The distribution is singular at  $y = 0$ : if the particle starts at the origin, then it’s impossible to define its Riemann sheet. The form (44) compares favorably with numerical simulations (Fig. 6(a)), though there are problems with the tails, as is obvious in Fig. 6(b).

## References

- AREF, H. 1984 Stirring by chaotic advection. *J. Fluid Mech.* **143**, 1–21.
- BÉLISLE, C. 1989 Windings of random walks. *Ann. Prob.* **17** (4), 1377–1402.
- BÉLISLE, C. & FARAWAY, J. 1991 Winding angle and maximum winding angle of the two-dimensional random walk. *J. Appl. Prob.* **28** (4), 717–726.
- BERGER, M. A. 1987 The random walk winding number problem: convergence to a diffusion process with excluded area. *J. Phys. A* **20**, 5949–5960.
- BERGER, M. A. & ROBERTS, P. H. 1988 On the winding number problem with finite steps. *Adv. Appl. Prob.* **20** (2), 261–274.

- CARSLAW, H. S. & JAEGER, J. G. 1959 *Conduction of heat in solids*, 2nd edn. Oxford, U.K.: Oxford University Press.
- DROSSEL, B. & KARDAR, M. 1996 Winding angle distributions for random walks and flux lines. *Phys. Rev. E* **53** (6), 5861–5871.
- DUFFY, D. G. 2001 *Green's functions with applications*. Boca Raton, FL: Chapman & Hall/CRC Press.
- EDWARDS, S. F. 1967 Statistical mechanics with topological constraints: I. *Proc. Phys. Soc.* **91**, 513–519.
- FISHER, M. E., PRIVMAN, V. & REDNER, S. 1984 The winding angle of planar self-avoiding walks. *J. Phys. A* **17**, L569.
- GROBERG, A. & FRISCH, H. 2003 Winding angle distribution for planar random walk, polymer ring entangled with an obstacle, and all that: Spitzer-Edwards-Prager-Frisch model revisited. *J. Phys. A* **36** (34), 8955–8981.
- NECHAEV, S. K. 1996 *Statistics of Knots and Entangled Random Walks*. Singapore; London: World Scientific.
- PITMAN, J. & YOR, M. 1986 Asymptotic laws of planar Brownian motion. *Ann. Prob.* **14** (3), 733–779.
- PITMAN, J. & YOR, M. 1989 Further asymptotic laws of planar Brownian motion. *Ann. Prob.* **17** (3), 965–1011.
- PONTIN, D. I., WILMOT-SMITH, A. L., HORNIG, G. & GALSGAARD, K. 2011 Dynamics of braided coronal loops. II. Cascade to multiple small-scale reconnection events. *Astron. Astrophys.* **525**, A57.
- REDNER, S. 2001 *A guide to first-passage processes*. Cambridge, U.K.: Cambridge University Press.
- RUDNICK, J. & HU, Y. 1987 The winding angle distribution for an ordinary random walk. *J. Phys. A* **20**, 4421–4438.
- SPITZER, F. 1958 Some theorems concerning 2-dimensional Brownian motion. *Trans. Amer. Math. Soc.* **87**, 187–197.

- SUMNERS, D. W. 2009 Random knotting: Theorems, simulations and applications. In *Lectures on Topological Fluid Mechanics* (ed. R. L. Ricca), pp. 187–217. Berlin: Springer.
- THIFFEAULT, J.-L. 2005 Measuring topological chaos. *Phys. Rev. Lett.* **94** (8), 084502.
- THIFFEAULT, J.-L. 2010 Braids of entangled particle trajectories. *Chaos* **20**, 017516.
- TÖRÖK, T., BERGER, M. A. & KLIEM, B. 2010 The writhe of helical structures in the solar corona. *Astron. Astrophys.* **516**, A49.
- TUMASZ, S. E. 2012 Topological stirring. PhD thesis, University of Wisconsin – Madison, Madison, WI.
- WILMOT-SMITH, A. L. & DE MOORTEL, I. 2007 Magnetic reconnection in flux-tubes undergoing spinning footpoint motions. *Astron. Astrophys.* **473**, 615–623.
- WILMOT-SMITH, A. L., PONTIN, D. I. & HORNIG, G. 2010 Dynamics of braided coronal loops. I. Onset of magnetic reconnection. *Astron. Astrophys.* **516**, A5.
- WILMOT-SMITH, A. L., PONTIN, D. I., YEATES, A. R. & HORNIG, G. 2011 Heating of braided coronal loops. *Astron. Astrophys.* **536**, A67.
- YEATES, A. R. & HORNIG, G. 2011 Dynamical constraints from field line topology in magnetic flux tubes. *J. Phys. A* **44**, 265501.