Université catholique de Louvain

École polytechnique de Louvain (EPL)

Complex Network Analysis for Marine Models

Subtitle (optional)

Dissertation presented by
Renaud DUFAYS

for obtaining the Master's degree in
Mathematical Engineering

Supervisors
Eric DELEERSNIJDER, Jean-Charles DELVENNE

Reader
Laurent DELANNAY

Academic year 2016-2017

Chapter 1

The "overturner" model

1.1 Mathematical model

1.1.1 An idealised velocity field

We developp here the idealized representation of the meridian circulation in the Atlantic ocean that will be studied in the next chapters. ici : développement mathématique. Plus loin on donne des valeurs aux paramètres avec un "physical insight". We consider a rectangular domain in the (y, z)-coordinate system. The coordinate y is associated to the latitude with $\hat{\mathbf{e}}_y$ pointing towards the North, and z is associated to the depth with $\hat{\mathbf{e}}_z$ pointing upwards. The domain Ω is delimited by

$$0 \leq y \leq L, \quad 0 \leq z \leq H, \tag{1.1}$$

where L and H are positive constants. The ocean surface is thus located at $z = H$ while $z = 0$ stands for the deep-ocean. The South and North boundaries are respectively given by $y = 0$ and $y = L$. We aim at defining a stationnary velocity field $\mathbf{u}(y, z) = (v(y, z), w(y, z))$ that would roughly reproduce the main qualitative features of the meridian circulation in the Atlantic ocean. The continuity equation reads

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{1.2}$$

where ρ is the density of the seawater mixture. We note $\partial\Omega$ its boundary. We simplify this equation by making the very common *Boussinesq appromation* : in the aquatic environment, water is, by far, the dominant constituent. The density of seawater is thus close to that of pure water, ρ_w. The latter depends on the temperature and pressure, but the variations are often very small.Let $\bar{\rho}$ and $\Delta\rho$ be appropriate reference values of the density and the order of magnitude of its variation. The key assumption in the *Boussinesq approximation* is that

$$\frac{\Delta \rho}{\bar{\rho}} \ll 1. \tag{1.3}$$

To assess the impact of this assumption on the continuity equation, we consider its dimensionless form. Let U, T and X be relevant velocity-, time- and space-scales. This allows to introduce the following dimensionless variables, denoted by primes :

$$\rho' = \frac{\rho - \bar{\rho}}{\Delta \rho}, \quad \mathbf{u}' = \frac{\mathbf{u}}{U}, \quad t' = \frac{t}{T}, \quad \text{and} \quad \mathbf{x}' = \frac{\mathbf{x}}{X}, \tag{1.4}$$

where $\mathbf{x} = (y, z)$. The dimensionless version of the continuity equation (1.2) reads then:

$$\frac{\Delta \rho}{T} \frac{\partial \rho'}{\partial t'} + \frac{U \Delta \rho}{X} \mathbf{u}' \cdot \nabla' \rho' + \frac{U(\bar{\rho} + \rho' \Delta \rho)}{X} \nabla' \cdot \mathbf{u}' = 0. \tag{1.5}$$

Multiplying both sides by $X/(U\bar{\rho})$ yields :

$$\frac{X}{UT}\frac{\Delta\rho}{\bar{\rho}}\frac{\partial\rho'}{\partial t'} + \frac{\Delta\rho}{\bar{\rho}}\mathbf{u}'\cdot\nabla'\rho' + \left(1+\frac{\Delta\rho}{\bar{\rho}}\rho'\right)\nabla'\cdot\mathbf{u}' = 0. \tag{1.6}$$

By taking (1.3) into account, this equation simplifies to $\nabla'\cdot\mathbf{u}' = 0$, or equivalently in dimensional variables $\nabla\cdot\mathbf{u} = 0$. For our particular problem, this amounts to

$$\frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0. \tag{1.7}$$

No-through boundary conditions are imposed at the boundaries of the domain, which implies that $\mathbf{u}(y,z)\cdot\hat{\mathbf{n}} = 0$ everywhere on $\partial\Omega$ (where $\hat{\mathbf{n}}$ is the outwards unit normal at the boundary), or equivalently :

$$v(0,z) = 0, \quad v(L,z) = 0, \quad w(y,0) = 0 \quad \text{and} \quad w(y,H) = 0. \tag{1.8}$$

Blahblah à mettre en relation avec ce qu'on doit dire plus tôt sur les modèles 2D de l'océan Atlantique,... Éventuellement s'inspirer de Timmermans mais attention quand même...

Furthermore, since the relation $\nabla\cdot(\nabla\times\mathbf{a}) = 0$ holds true for any 3-dimensional potential vector $\mathbf{a}(x,y,z)$ whose second partial derivatives are continuous,[1] we can choose a relevant $\mathbf{a}(x,y,z) = (a_x(x,y,z),\ a_y(x,y,z),\ a_z(x,y,z))$ with a_x, a_y and a_z of class $\mathcal{C}^2$ and impose that

$$\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = -\nabla\times\mathbf{a} = -\begin{pmatrix} \frac{\partial a_z}{\partial y} - \frac{\partial a_y}{\partial z} \\ \frac{\partial a_x}{\partial z} - \frac{\partial a_z}{\partial x} \\ \frac{\partial a_y}{\partial x} - \frac{\partial a_x}{\partial y} \end{pmatrix}. \tag{1.9}$$

This ensures that $\mathbf{u}$ satisfies the continuity equation. Here, we consider a 2-dimensional flow in the plane $(\hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z)$. Hence $u = 0$ and $\partial\cdot/\partial x = 0$. The relation (1.9) becomes

$$\mathbf{u} = \begin{pmatrix} 0 \\ v \\ w \end{pmatrix} = -\nabla\times\mathbf{a} = \begin{pmatrix} 0 \\ -\frac{\partial a_x}{\partial z} \\ \frac{\partial a_x}{\partial y} \end{pmatrix}, \tag{1.10}$$

where only the component a_x is needed to describe $\mathbf{u}$. Hence, the velocity field of a flow in the plane is described by a scalar quantity, the so-called *streamfunction*, generally noted ψ. The potential vector $\mathbf{a}$ is thus of the form $\mathbf{a}(y,z) = (\psi(y,z),\ 0,\ 0)$, and the meridional and vertical components of the velocity vector are given by :

$$v = -\frac{\partial\psi}{\partial z}, \quad w = \frac{\partial\psi}{\partial y}. \tag{1.11}$$

Note that adding any constant to ψ leaves the velocity vector unchanged. This adds some freedom to the choice of ψ. The idea is now to propose a reasonable streamfunction. Then, deriving

[1] The proof is quite straightforward :

$$\begin{aligned}
\nabla\cdot(\nabla\times\mathbf{a}(x,y,z)) &= \nabla\cdot\left[\left(\frac{\partial a_z}{\partial y} - \frac{\partial a_y}{\partial z}\right)\hat{\mathbf{e}}_x + \left(\frac{\partial a_x}{\partial z} - \frac{\partial a_z}{\partial x}\right)\hat{\mathbf{e}}_y + \left(\frac{\partial a_y}{\partial x} - \frac{\partial a_x}{\partial y}\right)\hat{\mathbf{e}}_z\right] \\
&= \frac{\partial^2 a_z}{\partial x\partial y} - \frac{\partial^2 a_y}{\partial x\partial z} + \frac{\partial^2 a_x}{\partial y\partial z} - \frac{\partial^2 a_z}{\partial y\partial x} + \frac{\partial^2 a_y}{\partial z\partial x} - \frac{\partial^2 a_x}{\partial z\partial y} \\
&= 0.
\end{aligned}$$

Here, we have assumed that $\mathbf{a}(x,y,z)$ is sufficiently smooth, or more precesily that the second partial derivatives of a_x, a_y and a_z are continuous. This allows to use *Schwarz's theorem* which states that in that case, the second partial derivatives are symmetric.

the velocity components from that streamfunction will ensure that the continuity equation is satisfied. In order to derive a streamfunction that is relevant to our problem, we need to get some physical intuition about the streamfunction. To this end, two fundamental properties of the streamfunction are rederived in the frame below.

Some properties of the streamfunction

First, notice that assuming that $\psi \in \mathcal{C}^2$ implies straightforwardly that $\mathrm{d}\psi = (\partial\psi/\partial y)\mathrm{d}y + (\partial\psi/\partial z)\mathrm{d}z$ is an *exact differential* since by Schwarz's theorem

$$\frac{\partial^2\psi}{\partial y\partial z} = \frac{\partial^2\psi}{\partial z\partial y}. \tag{1.12}$$

Thus,

$$\int_{\mathbf{x}_1}^{\mathbf{x}_2} \mathrm{d}\psi = \psi(y_2, z_2) - \psi(y_1, z_1) \tag{1.13}$$

is path-independant.

An important property of the streamfunction in two dimensions is that the curves along which ψ is constant are exactly the *streamlines* of the flow, namely the family of curves that are instantaneously tangent to the velocity vector. To show that, let such a curve be parametrized by $s \mapsto \mathbf{x}_S(s) = (y_S(s), z_S(s))$. The fact that ψ is constant along that curve implies that $\mathrm{d}\psi_S = (\partial\psi/\partial y)\mathrm{d}y_S + (\partial\psi/\partial z)\mathrm{d}z_S = \nabla\psi \cdot \mathrm{d}\mathbf{x}_S = 0$. This shows that vector $\nabla\psi$ is normal to the curve $\mathbf{x}_S(s)$. Hence, showing that $\mathbf{x}_S(s)$ is everywhere tangent to $\mathbf{u}$ is equivalent to showing that $\mathbf{u} \cdot \nabla\psi = 0$ everywhere. The latter is straightforward using relation (1.11) :

$$\mathbf{u} \cdot \nabla\psi = -\frac{\partial\psi}{\partial z}\frac{\partial\psi}{\partial y} + \frac{\partial\psi}{\partial y}\frac{\partial\psi}{\partial z} = 0, \tag{1.14}$$

which concludes the proof.

Now we show another interesting property of the streamlines, namely that the *volume flow rate* between two streamlines of values ψ_1 and ψ_2 is equal to the difference of those streamlines, $\psi_1 - \psi_2$. To show that, consider two infinitely close points $\mathbf{x}_1 = (y_1,\, z_1)$ and $\mathbf{x}_1 + \mathrm{d}\mathbf{x} = (y_1 + \mathrm{d}y,\, z_1 + \mathrm{d}z)$. At those points, the streamfunction has values $\psi(y_1, z_1) = \psi_1$ and $\psi(y_1 + \mathrm{d}y, z_1 + \mathrm{d}z) = \psi_1 + \mathrm{d}\psi$. Let us now consider the volume flow rate $\mathrm{d}q$ accross the infinitesimal segment $[\mathbf{x}_1,\, \mathbf{x}_1 + \mathrm{d}\mathbf{x}]$, positive in the right-hand side direction of the segment if the latter is directed from $\mathbf{x}_1$ to $\mathbf{x}_1 + \mathrm{d}\mathbf{x}$. It is equal to $\mathbf{u} \cdot \hat{\mathbf{n}}$, where $\hat{\mathbf{n}} = (\mathrm{d}z,\, -\mathrm{d}y)$ is the unit normal to the segment, oriented in the right-hand side direction. Hence, $\mathrm{d}q = v\mathrm{d}z - w\mathrm{d}y$, which, using relation (1.11), amounts to

$$\mathrm{d}q = -\mathrm{d}\psi. \tag{1.15}$$

Now, consider any two points $\mathbf{x}_1 = (y_1, z_1)$ and $\mathbf{x}_2 = (y_2, z_2)$ in the (connected) domain. The volume flow rate $q_{1\to 2}$ accross any curve $\gamma_{1\to 2}$ connecting $\mathbf{x}_1$ to $\mathbf{x}_2$, positive in the right-hand side direction of the directed segment $[\mathbf{x}_1,\, \mathbf{x}_2]$ is

$$q_{1\to 2} = \int_{\gamma_{1\to 2}} \mathrm{d}q = \int_{\mathbf{x}_1}^{\mathbf{x}_2} (-\mathrm{d}\psi) = \psi(y_1, z_1) - \psi(y_2, z_2), \tag{1.16}$$

where $\int_{\gamma_{1\to 2}}$ is the line integral along a curve connecting $\mathbf{x}_1$ to $\mathbf{x}_2$, afterwards noted $\int_{\mathbf{x}_1}^{\mathbf{x}_2}$ to emphasize the fact that it does not depend on the integration path, since $\mathrm{d}\psi$ is an exact differential.

Now we are able to derive a relevant streamfunction. In particular, ψ must be such that the boundary conditions (1.8) are satisfied. Those conditions state that $\mathbf{u}$ must be tangent to the boundary everywhere on $\partial\Omega$, which precisely amounts to require that ψ is constant on $\partial\Omega$. Without loss of generality, we can choose this constant to be zero. Hence, we require that

$$\psi(0,z)=0, \quad \psi(L,z)=0, \quad \psi(y,0)=0 \quad \text{and} \quad \psi(y,H)=0, \quad \text{for all } (y,z)\in\Omega. \tag{1.17}$$

Faire des liens avec chapitre précédent In order to build an acceptable idealisation of the meridian circulation in the Atlantic ocean, *Deleersnijder* proposes in his working paper [1] to suppose that the meridian streamfunction has a unique extremum Ψ, which is a maximum, and that it reaches that maximum at the point of coordinates (y_0, z_0), located near the surface and the North boundary of the domain. It is important to recall that the second partial derivatives of ψ must exist and be continuous for the above relations to hold.

Let $\xi_0 \in \mathbb{R}_0^+$, and let $\phi(\xi,\xi_0)$ be defined as

$$\phi(\xi,\xi_0)=\frac{\xi(2\xi_0-\xi)}{\xi_0^2}, \tag{1.18}$$

The derivative $\phi'(\xi,\xi_0)$ of ϕ with respect to ξ is

$$\phi'(\xi,\xi_0)=\frac{2(\xi_0-\xi)}{\xi_0^2}. \tag{1.19}$$

An expression of the meridian streamfunction that satisfies the above constraints is then

$$\psi(y,z)=\Psi\begin{cases}\phi(y,y_0)\phi(z,z_0) & \text{if} \quad 0\le y<y_0, \quad 0\le z<z_0,\\ \phi(y,y_0)\phi(H-z,H-z_0) & \text{if} \quad 0\le y<y_0, \quad z_0<z\le H,\\ \phi(L-y,L-y_0)\phi(H-z,H-z_0) & \text{if} \quad y_0<y\le L, \quad z_0<z\le H,\\ \phi(L-y,L-y_0)\phi(z,z_0) & \text{if} \quad y_0<y\le L, \quad 0\le z<z_0.\end{cases} \tag{1.20}$$

As such, ψ is undefined along the lines $y=y_0$ and $z=z_0$. We consider thus the continuous prolongation of ψ at those points. Hence,

$$\psi(y_0,z)=\Psi\begin{cases}\phi(z,z_0) & \text{if} \quad 0\le z<z_0,\\ \phi(H-z,H-z_0) & \text{if} \quad z_0<z\le H,\end{cases} \tag{1.21}$$

$$\psi(y,z_0)=\Psi\begin{cases}\phi(y,y_0) & \text{if} \quad 0\le y<y_0,\\ \phi(L-y,L-y_0) & \text{if} \quad y_0<y\le L,\end{cases} \tag{1.22}$$

and

$$\psi(y_0,z_0)=\Psi. \tag{1.23}$$

The meridian and vertical components of the velocity are then expressed as

$$v(y,z)=\Psi\begin{cases}-\phi(y,y_0)\phi'(z,z_0) & \text{if} \quad 0\le y<y_0, \quad 0\le z<z_0,\\ \phi(y,y_0)\phi'(H-z,H-z_0) & \text{if} \quad 0\le y<y_0, \quad z_0<z\le H,\\ \phi(L-y,L-y_0)\phi'(H-z,H-z_0) & \text{if} \quad y_0<y\le L, \quad z_0<z\le H,\\ -\phi(L-y,L-y_0)\phi'(z,z_0) & \text{if} \quad y_0<y\le L, \quad 0\le z<z_0,\\ -\phi'(z,z_0) & \text{if} \quad y=y_0, \quad 0\le z<z_0,\\ \phi'(H-z,H-z_0) & \text{if} \quad y=y_0, \quad z_0<z\le H,\\ 0 & \text{if} \quad 0\le y\le L, \quad z=z_0.\end{cases} \tag{1.24}$$

and

$$w(y,z)=\Psi\begin{cases}\phi'(y,y_0)\phi(z,z_0) & \text{if} \quad 0\le y<y_0, \quad 0\le z<z_0,\\ \phi'(y,y_0)\phi(H-z,H-z_0) & \text{if} \quad 0\le y<y_0, \quad z_0<z\le H,\\ -\phi'(L-y,L-y_0)\phi(H-z,H-z_0) & \text{if} \quad y_0<y\le L, \quad z_0<z\le H,\\ -\phi'(L-y,L-y_0)\phi(z,z_0) & \text{if} \quad y_0<y\le L, \quad 0\le z<z_0,\\ \phi'(y,y_0) & \text{if} \quad 0\le y<y_0, \quad 0\le z=z_0,\\ -\phi'(L-y,L-y_0) & \text{if} \quad y_0<y\le L, \quad z=z_0,\\ 0 & \text{if} \quad y=y_0.\end{cases} \tag{1.25}$$

1.1.2 Estimation of the parameter values

Vérifier que c'est bien les valeurs finales The numerical values used here are based on personal communications with *E. Deleersnijder*. The model for the idealised meridian velocity field in the Atlantic ocean will be complete once we have assigned plausible values to the parameters. For this purpose, some physical insight is needed. First, the Atlantic ocean extends approximately from 50° South to 60° North, hence over $\frac{11}{18}\pi$ radians. With the radius of the Earth estimated to 6 371 km, we get that L must be close to $(\frac{11}{18}\pi)(6\,371) = 12\,231$ km. Moreover, the mean depth of the Atlantic ocean is about 4 km. Hence, we choose $L = 12\,000$ km and $H = 4$ km. In virtue of the properties of the streamfunction, the maximum $\Psi = \psi(y_0, z_0)$ of the meridian streamfunction is equal to the volume flow rate accross any curve connecting (y_0, z_0) to a point on the boundary of the domain: it is thus a measure of the intensity of the meridian circulation. With an estimated rate of deep convection in the Atlantic ocean of about 20 Sv and a mean width of about 5 000 km, this yields $\Psi = 4$ m^2/s. Finally, we use $y_0 = 11\,000$ km and $z_0 = 3.5$ km based on qualitative inspection of the meridian streamfunction graph. Characteristic values V and W of the meridional and vertical speed are, in virtue of relations (1.11) :

$$V = \frac{\Psi}{H} \quad \text{and} \quad W = \frac{\Psi}{L}. \tag{1.26}$$

According to *E. Deleersnijder* [personal communication], the characteristic time scale T should be of the order of a few hundred years in order to be physically significant. It is expressed as

$$T = \frac{L}{V} = \frac{H}{W} = \frac{LH}{\Psi} = 1.5\times10^6 \text{ s} \approx 475.6 \text{ years}, \tag{1.27}$$

an acceptable value.

With those values of the parameters, the isolines of the adimensional streamfunction ψ/Ψ are shown in figure 1.1, and the meridional and vertical components of the velocity field are illustrated in figure 1.2.

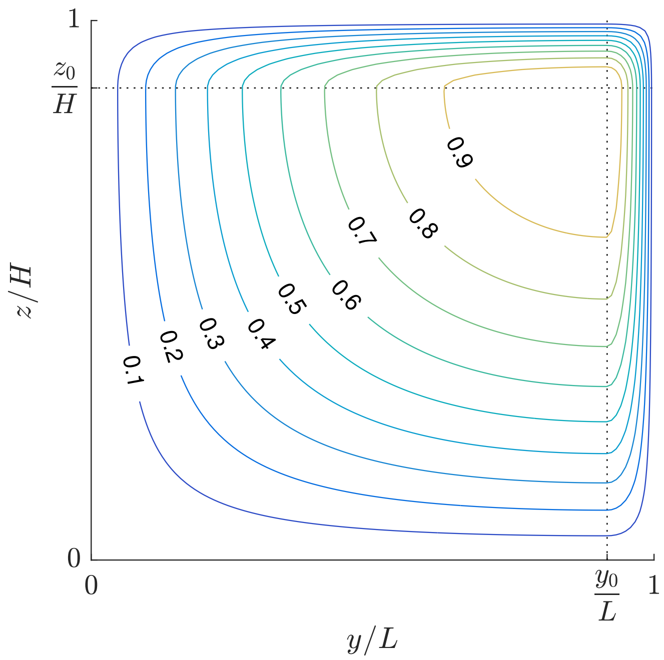
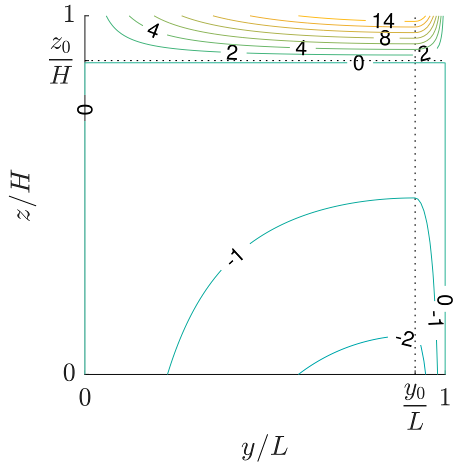
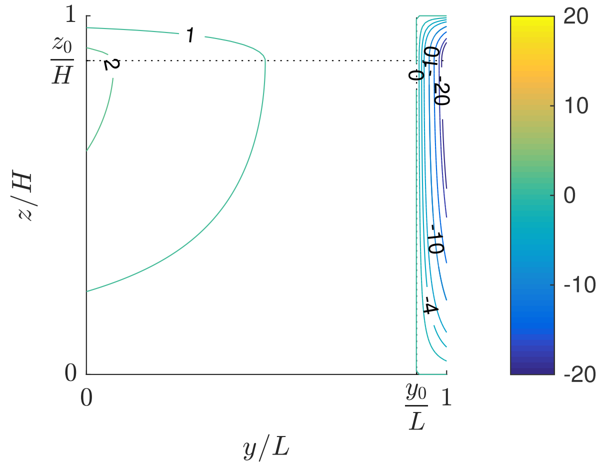


Figure 1.1 – Some isolines of the adimensional meridian streamfunction $\psi(y,z)/\Psi$, which are also streamlines of the idealised meridian circulation in the Atlantic ocean.



(a) $v(y, z)$.



(b) $w(y, z)$.

Figure 1.2 – Meridional and vertical components of the idealised velocity fields in the adimensional domain. Here, $y_0 = \frac{11}{12}L$ and $z_0 = \frac{7}{8}H$.

1.1.3 Injection of a passive tracer into the ocean

The fate of a passive tracer injected at location (y_*, z_*) into the idealised Atlantic ocean depicted previously can be described by a differential problem on that tracer's concentration. The tracer could be any passive tracer whose concentration in the athmosphere is negligible, for example a dye or a set of seawater particles initially located at (y_*, z_*). The concentration of the tracer $C(t, y, z)$ in the ocean obeys the following partial differential equation :

$$\frac{\partial C}{\partial t} = -\nabla \cdot (\mathbf{u}C - \mathbf{K}\nabla C), \tag{1.28}$$

where $\mathbf{K}$ is the *diffusivity tensor*. Without loss of generality, we can assume $\mathbf{K}$ to be symmetric. This is essentially because the impact of the anti-symmetric part of $\mathbf{K}$, if any, may be viewed as additional advection. More details may be found in appendix A of [2]. Of course, the symmetric tensor $\mathbf{K}$ must then be positive-definite in order to represent truly diffusive processes, namely phenoma which tend, at any time and location, to homogenise the concentration of any constituent. For our problem, we consider that $\mathbf{K}$ has the form

$$\mathbf{K}(y, z) = \begin{pmatrix} K_h & 0 \\ 0 & K_v(y, z) \end{pmatrix}, \tag{1.29}$$

where K_h is a positive constant and

$$K_v(y, z) = \begin{cases} K_{v_1} & \text{if} \quad y_0 \leq y \leq L, \quad 0 \leq z \leq H, \\ K_{v_2} & \text{if} \quad 0 \leq y < y_0, \quad 0 \leq z < z_0, \\ K_{v_3} & \text{if} \quad 0 \leq y < y_0, \quad z_0 \leq z \leq H, \end{cases} \tag{1.30}$$

with K_{v_1}, K_{v_2} and K_{v_3} positive constants. In the framework of the idealised model of the meridian circulation in the Atlantic ocean, *E. Deleersnijder* [personal communication] proposes the values $K_h = 10^3$ m^2/s, $K_{v_1} = 10^{-1}$ m^2/s, $K_{v_2} = 10^{-4}$ m^2/s and $K_{v_3} = 10^{-3}$ m^2/s. The relatively large value of K_{v_1} allows to represent deep convection in the corresponding zone without having to implement a convective adjustment algorithm. The developped form of (1.28) is then

$$\frac{\partial C}{\partial t} = -\frac{\partial}{\partial y}\left(vC - K_h \frac{\partial C}{\partial y}\right) - \frac{\partial}{\partial z}\left(wC - K_v(y, z)\frac{\partial C}{\partial z}\right). \tag{1.31}$$

No-flux conditions are imposed at the boundaries vérifier/discuter le flux nul à la surface

$$K_h \frac{\partial C}{\partial y}\bigg|_{y=0} = 0, \quad K_h \frac{\partial C}{\partial y}\bigg|_{y=L} = 0, \quad K_v \frac{\partial C}{\partial z}\bigg|_{z=0} = 0, \quad \text{and} \quad K_v \frac{\partial C}{\partial z}\bigg|_{z=H} = 0. \tag{1.32}$$

The initial condition is

$$C(0, y, z) = \delta(y - y_*)\delta(z - z_*), \tag{1.33}$$

where δ is the Dirac delta function, such that

$$\int_0^L \int_0^H C(0, y, z)\mathrm{d}y\mathrm{d}z = 1. \tag{1.34}$$

In order to get a formulation of the problem using as few independant parameters as possible, it is interesting to consider the adimensional formulation. Such a scaling is particularly interesting for sensitivity analysis. The adimensional independant variables are

$$t' = \frac{t}{T} = \frac{t\Psi}{LH}, \quad y' = \frac{y}{L} \quad \text{and} \quad z' = \frac{z}{H}, \tag{1.35}$$

where T is the time scale introduced in (1.27). The adimensional hydrodynamic variables are:

$$\psi' = \frac{\psi}{\Psi}, \quad v' = \frac{v}{V} = \frac{vH}{\Psi}, \quad \text{and} \quad w' = \frac{w}{W} = \frac{wL}{\Psi}, \tag{1.36}$$

where V and W are the velocity scales introduced in (1.26). The adimensional concentration is

$$C' = \frac{C}{C_r}, \tag{1.37}$$

where C_r is a characteristic value of the concentration. We will see shortly that there is no needed to assign a particular value to C_r. The adimensional form of equation (1.31) is then

$$\frac{\partial C'}{\partial t'} = -\frac{\partial}{\partial y'}\left(v'C' - \frac{1}{Pe_h}\frac{\partial C'}{\partial y'}\right) - \frac{\partial}{\partial z'}\left(w'C' - \frac{1}{Pe_v(y,z)}\frac{\partial C'}{\partial z'}\right), \tag{1.38}$$

where

$$Pe_h = \frac{\Psi L}{K_h H} \quad \text{and} \quad Pe_v(y, z) = \frac{\Psi H}{K_v(y, z)L} \tag{1.39}$$

are the horizontal and vertical Péclet numbers. They correspond to the ratio between the characteristic advective and diffusive velocity scales. Indeed, the horizontal and vertical diffusive velocity scales V_d and W_d are

$$V_d = \frac{K_h}{L} \quad \text{and} \quad W_d(y, z) = \frac{K_v(y, z)}{H}. \tag{1.40}$$

There are three different vertical diffusive velocity scale depending on which zone of the ocean we consider Donner des noms aux zones dans le chap 1 : 1 = ?, 2 = "Deep convection" et 3 = "surface flow". The advective velocity scales V and W have already been introduced in (1.26). The Péclet numbers may then be rewritten as

$$Pe_h = \frac{V}{V_d} = 12, \tag{1.41}$$

and

$$Pe_v(y, z) = \frac{W}{W_d(y, z)} = \begin{cases} Pe_{v_1} = 1.33 \times 10^{-2} & \text{if} \quad y_0 \le y \le L, \quad 0 \le z \le H, \\ Pe_{v_2} = 13.3 & \text{if} \quad 0 \le y < y_0, \quad 0 \le z < z_0, \\ Pe_{v_3} = 1.33 & \text{if} \quad 0 \le y < y_0, \quad z_0 \le z \le H. \end{cases} \tag{1.42}$$

This shows that the advective and diffusive processes are of equal importance in the dynamics of our model, excepted in the zone of deep convection where the vertical diffusion dominates the vertical convection. This is because we have chosen to represent deep convection via a heavy vertical mixing in that zone.

Chapter 2

Numerical Considerations

À retravailler une fois le travail fini. The final goal of this work is to develop a simpler box-model of the overturner model, where the boxes are determined via a community detection method. In order to apply such a community detection method, the solution to the overturner transport model (1.28) must be computed at different times and for different initial conditions. However, equation (1.28) cannot be solved analytically so that numerical methods must be resorted to. A lot of different numerical methods have been developed through the years, each even their pros and cons. Here, we briefly summarize the main types of methods available, with their main advantages and disadvantages. This allows then to make an informed decision about which method to use in this work. Most of the discussion in the following paragraphs is inspired from [3] and [4].

A very popular class of numerical methods is formed by the Eulerian methods, in which the advection-diffusion equation is solved on a fixed spatial grid. This class encompass the finite difference method, finite element method and finite volume method. A second class is formed by the Lagrangian methods, where particles are followed through space at every time step. As we shall see later, the movement of an individual particle is modeled with a stochastic differential equation (SDE) which is consistent with the advection-diffusion equation. The idea is to estimate the concentration by simulating the trajectories of a large number of particles and taking averages. Several averaging methods have been developed to estimate the concentration from the set of particles' positions: we shall come back to this further. Finally, a third class of mixed Eulerian-Lagrangian methods has been developed, which basically attempt to combine the advantages of both approaches. Such conceptually attractive methods have been widely applied in applications. They are however more complex to implement; in view of the relative simplicity of the overturner problem, we will not consider such mixed methods here.

Both Eulerian and Lagrangian methods have their own advantages and disadvantages. The Eulerian methods provide the convenience of a fixed grid and are easy to implement. The main drawbacks are the inherent dispersion errors and artificial oscillations, leading to solutions that may be neither mass conservative nor positive [5], [6]. Basically, the effect of dispersion errors is similar to physical dispersion but it is due to truncation errors. Artificial oscillations are typical from higher order methods designed to reduce dispersion errors. Those problems could become excessively severe in case of problems involving a sharp concentration front, for instance advection-dominated problems or problems with large gradients on the initial concentration field (typically delta-like initial concentration) [7]. In those cases, numerical dispersion (i.e. dispersion due to dispersion errors) tends to inappropriately smooth out the sharp concentration front, whereas artificial oscillations tends to become more important, leading to serious problem with the positiveness of the solution.

On the other hand, Lagrangian methods ensures that the solution is always mass conservative and nonnegative. They are thus more suited than Eulerian methods to advection-dominated problems and to problems with large concentration gradients, since they do not suffer from dispersion errors and artificial oscillations. Moreover, if the tracer does not occupy the whole domain, the Lagrangian methods may be computationally more efficient than their Eulerian counterpart [8]. Depending on the number of particles used, Lagrangian methods may also require less storage than finite differences or finite element methods. Another advantage of Lagrangian methods is that they make it possible to advect the particles exactly when the velocity field can locally be described by an analytical function [9]. Finally, because each realization of the particle movement is independent from the others, Lagrangian methods are perfect candidates for parallelization. For instance, the MPI library makes it pretty easy and efficient to parallelize a code based on random walk models. Among the drawbacks of particle methods, the lack of a fixed grid may lead to numerical instability and computational difficulties [10]. If flow variables are not known analytically, their interpolation to the particle location could lead to local mass balance errors and solution anomalies [11]. Finally, the number of particles needed to get a smooth solution might be large leading to a large computational time, but this can be compensated by a parallelization of the code.

Considering the above discussion, it seems that a Lagrangian method is more appropriate for our concern. Indeed, in order to build an approximation of the transition probability matrix, the domain Ω is partitioned into boxes. For a simulation time T, an entry $[\mathbf{M}(T)]_{ij}$ is the probability that a particle goes from box i to box j in a time T. To estimate the entries of a line i of the matrix, the idea is to run a simulation for a time T and an initial concentration which is uniform in box i and zero in every other box. The initial concentration is thus sharp, an first argument in favor of a Lagrangian method. Furthermore, the concentration is interpreted as a probability, hence positiveness and mass conservation are crucial topics, another point for Lagrangian methods. Finally, the flow variables are in our case known analytically, which considerably reduce the drawbacks of Lagrangian methods pointed out above. For these reasons, we choose to go for a Lagrangian method.

2.1 Preliminaries

We introduce here the notions of stochastic differential equations (SDE's) and stochastic integrals. Those are the fundamental tools at the basis of the Lagrangian numerical methods. The discussion that comes next is based on several references, including [12], [13], [14] and [3]. Other references have been used for local parts of the work; they are then cited when they come in handy. Some results are stated without proofs. In such cases, unless otherwise stated, the reader may refer to [12] for formal proofs.

The idea behind Lagrangian methods is to estimate the concentration obeying an advection-diffusion-reaction equation by simulating the trajectories of a large number of particles in the flow. In this work, we restrict ourselves to advection diffusion equations of the form (1.28), which we recall here for the sake of readability:

$$\frac{\partial C}{\partial t} = \nabla \cdot (-\mathbf{u}C + \mathbf{K}\nabla C) \tag{2.1}$$

In the next, equation (2.1) will be referred to as the *transport model*. In order to implement a numerical method tracking the fates of individual particles, an equation describing the fate of such a particle must be derived, and that equation must be consistent with the transport model. Formally, the transport model can be interpreted as a Fokker-Planck equation, namely the partial differential equation governing the time evolution of the probability density function $p(\mathbf{x}, t)$ of the position of a particle. The correspondence is made by interpreting the concentration as the probability density function: $p = C$.

At the microscopic scale, Brownian diffusion is modeled by a stochastic force acting on the particles. This force is interpreted as the resultant of atomic bombardment on the particle. Intuitively, the direction of the force due to atomic bombardment is constantly changing, and at different times the particle is hit more on one side than another, leading to the seemingly random nature of the force, and hence of the motion. Therefore, the differential equation governing the position $\mathbf{x}(t)$ of a particle is stochastic. For example, Langevin proposed in 1908 an equation governing the position of a Brownian particle, which in 1D can be written in the form :

$$\frac{dx}{dt} = a(x,t) + b(x,t)\xi(t), \tag{2.2}$$

where x is the position of the particle, $a(x,t)$ and $b(x,t)$ are known functions and $\xi(t)$ is the rapidly fluctuating random term. The simplest model is obtained by considering that $\xi(t)$ is a white noise, i.e.

$$\begin{cases} \langle \xi(t) \rangle = 0, & \text{(2.3a)} \\ \langle \xi(t)\xi(t') \rangle = \delta(t-t'), & \text{(2.3b)} \end{cases}$$

where $\langle \cdot \rangle$ denotes expectation. The fact that ξ has zero mean is because any nonzero mean can be absorbed in the term $a(x,t)$. The second condition states that $\xi(t)$ is uncorrelated, namely that the random force acting on a particle at a time is independent of the random forces acting on that particle at any other time. This simple form of the noise is of course an unrealistic idealization of the atomic bombardment force. It is possible to show that

$$\int_0^t \xi(t')\mathrm{d}t' = W(t), \tag{2.4}$$

where $W(t)$ is the *Wiener process*, a *continuous* stochastic process defined by the following characteristics:

$$\begin{cases} W(0) = 0, & \text{(2.5a)} \\ W(t_2) - W(t_1) \sim \mathcal{N}(0, t_2 - t_1), & \text{(2.5b)} \\ \langle [W(t_4) - W(t_3)][W(t_2) - W(t_1)] \rangle = 0, & \text{(2.5c)} \end{cases}$$

where $t_1 < t_2 < t_3 < t_4$. In other words, $W(t)$ is a zero mean gaussian process of variance t which has the property of independent increments. Suppose now that $a(x,t) = a$ and $b(x,t) = b$ are constant. The above relations imply that the solution to (2.2) is

$$x(t) = at + bW(t), \tag{2.6}$$

where we implicitly assumed that $x(0) = 0$. However, one can show that the Wiener process is not differentiable with probability 1. We are thus faced with a paradox here since this implies that $x(t)$ is itself non-differentiable, and hence that the Langevin equation as stated in (2.2) *does not exist mathematically.* In fact, $\xi(t)$ is the derivative of $W(t)$ in the *distributive sense.* From (2.4), it follows directly that

$$\mathrm{d}W(t) \equiv W(t + \mathrm{d}t) - W(t) = \xi(t)\mathrm{d}t, \tag{2.7}$$

but it is incorrect (or at least very misleading) to write $\frac{dW(t)}{dt} = \xi(t)$, since the Wiener process is nowhere differentiable with probability 1, as already stated above.

Hopefully this introductory example shows the need for some preliminary steps in order to rigorously define a Stochastic Differential Equation (SDE), and to formalize the link between SDE's and Fokker-Planck equations. This is precisely the goal of the next pages.

2.1.1 Formal definition of a SDE

In this section, we restrict ourselves to a one-dimensional problem. This allows to make the notations less cumbersome while still introducing all the tools and concepts that are needed in order to understand the two-dimensional Lagrangian model which is at the basis of our numerical resolution of the overturner model for the meridian concentration in the Atlantic ocean. Indeed, all the results presented here can almost straightforwardly be extended to several dimensions. Consider again the Langevin equation (2.2). We have shown in the introduction that the equation does not really make sense under that form. What we are now going to show is that the corresponding *integral equation*

$$x(t) = x(t_0) + \int_{t_0}^{t} a(x(s), s)\mathrm{d}s + \int_{t_0}^{t} b(x(s), s)\xi(s)\mathrm{d}s \tag{2.8}$$

can be interpreted consistently. The first integral is a standard Lebesgue integral of a function a of the stochastic process $x(t)$. The second integral has to be defined carefully, because of the presence of the white noise. By (2.7), we rewrite it as

$$\int_{0}^{t} b(x(s), s)\mathrm{d}W(s), \tag{2.9}$$

which is a kind of stochastic Stieltjes integral (see appendix A.1). Consider the partition of interval $[t_0,\, t]$:

$$t_0 \leq t_1 \leq t_2 \leq \cdots \leq t_{n-1} \leq t_n = t, \tag{2.10}$$

and define intermediate points $\tau_i \in [t_{i-1},\, t_i]$. Such a partition will often be used thereafter without introducing it explicitly every time it appears. The stochastic integral of b with respect to the Wiener process $\int_{t_0}^{t} b(x(s), s)\mathrm{d}W(s)$ is defined as a mean-square limit (see appendix A.2) of the partial sum

$$S_n = \sum_{i=1}^{n} b(x(\tau_i), \tau_i)[W(t_i) - W(t_{i-1})]. \tag{2.11}$$

Such a limit is not unique: it depends on the particular choice of τ_i. This sensibility to the choice of location $x(\tau_i)$ at which the function is evaluated is a consequence of the unbounded variation of the Wiener process. Popular choices are $\tau_i = t_i$, $\tau_i = \frac{1}{2}(t_{i-1} + t_i)$ or $\tau_i = t_i$, corresponding to the *Itô*, *Stratonovich* and *backward Itô* stochastic integrals, respectively. Hence, for a stochastic integral such as (2.9) to be well-defined, its *interpretation* must be stated explicitly. Namely, one must specify if the integral is taken in the *Itô*, *Stratonovich* or *backward Itô* sense, or any other interpretation corresponding to a given choice of τ_i. Only the Itô and backward Itô interpretations will be considered in this work. To avoid any confusion, we will denote the Itô integral by $(\mathbf{I})\int$ and the backward Itô integral by $(\mathbf{bI})\int$. Hence

$$(\mathbf{I})\int_{t_0}^{t} b(x(s), s)\mathrm{d}W(s) = \operatorname*{ms-lim}_{n\to\infty} \sum_{i=1}^{n} b(x(t_{i-1}), t_{i-1})[W(t_i) - W(t_{i-1})], \tag{2.12a}$$

$$(\mathbf{bI})\int_{t_0}^{t} b(x(s), s)\mathrm{d}W(s) = \operatorname*{ms-lim}_{n\to\infty} \sum_{i=1}^{n} b(x(t_i), t_i)[W(t_i) - W(t_{i-1})]. \tag{2.12b}$$

Remark In the backward Itô definition of the stochastic integral, only x needs to be evaluated at t_i: if $b(x, t)$ is differentiable in t (which is assumed here), the integral is independent of the particular choice of value for t in the range $[t_{i-1},\, t_i]$. Hence, we could replace $b(x(t_i), t_i)$ by $b(x(t_i), \tau_i)$ with $\tau_i \in [t_{i-1},\, t_i]$ in the definition (2.12b). Note that this remark is not restricted to backward Itô integration: if b is differentiable in t, only the location $x(\tau_i)$ at which b is evaluated in the sum affects the limit.

Conventionally, a stochastic differential equation such as (2.2) is written in the form

$$\begin{cases} \mathrm{d}x(t) = a(x(t), t)\mathrm{d}t + b(x(t), t)\mathrm{d}W(t), \\ x(t_0) = x_0. \end{cases} \tag{2.13}$$

Equation (2.13) has to be interpreted as the implicit integral equation

$$x(t) = x_0 + \int_{t_0}^{t} a(x(s), s)\mathrm{d}s + \int_{t_0}^{t} b(x(s), s)\mathrm{d}W(s), \tag{2.14}$$

and the interpretation of the stochastic integral must thus be stated. For example, one will talk about an Itô SDE or a backward Itô SDE, and the stochastic process $x(t)$ which is the solution to the SDE (2.13) is different whether the stochastic integral is computed in the Itô or backward Itô sense.

It can be shown that an Itô stochastic integral $(\mathbf{I})\int_{t_0}^{t} G(s)\mathrm{d}W(s)$ exists whenever the function G is *continuous* and *nonanticipating* on the closed interval $[t_0,\, t]$. G is a *nonanticipating* function of t is for all t and s such that $t < s$, $G(t)$ is statistically independent of $W(s) - W(t)$. This condition seems obviously satisfied for any deterministic function $b(x(t), t)$ of the stochastic process $x(t)$ obeying the SDE (2.13), since $x(t)$ only depends on anterior values of the Wiener process. For example, in the context of the position of a brownian particle, it is intuitively obvious that the unknown future collisions cannot affect the present position of the particle. From now on, we assume thus that we are dealing with *nonanticipating* functions.

2.1.2 Properties of the Itô integral

When working with nonanticipating functions, it is possible to take advantage of the fact that $G(t_{i-1})$ is independent of $W(t_i) - W(t_{i-1})$ to derive particularly useful properties of the Itô integral. Such properties are generally not true for Stratonovich or backward Itô integrals. However, there is a formula called Ito's formula that allows to build rules to transform any SDE into an Itô SDE. Hence, when working with a Stratonovich or backward Itô SDE, it is often useful to transform that SDE into an equivalent Itô SDE so that the previously mentioned properties can be applied. In this section, we first derive those properties, and then we show Itô's formula for the change of variables. Those results are fundamental for connecting a SDE to the Fokker-Plank equation and for building consistent numerical schemes, as we shall see later.

Rules for the differentials

One preliminary result of first-order importance in the context of stochastic integration is that, for any nonanticipating function G

$$(\mathbf{I})\int_{t_0}^{t} G(s)[dW(s)]^2 = \int_{t_0}^{t} G(s)\mathrm{d}s, \tag{2.15}$$

i.e. that

$$\lim_{n\to\infty} \left\langle \left(\sum_{i=1}^{n} G(t_{i-1})[W(t_i) - W(t_{i-1})]^2 - \int_{t_0}^{t} G(s)\mathrm{d}s \right)^2 \right\rangle = 0. \tag{2.16}$$

Let $\Delta W_i := W(t_i) - W(t_{i-1})$ and $\Delta t_i = t_i - t_{i-1}$. By the properties of the Wiener process, $\Delta W_i \sim \mathcal{N}(0, \Delta t_i)$. Hence, $\Delta W_i^2/\Delta t_i$ follows a χ-squared distribution with one degree of freedom. It has thus mean 1 and variance 2 and thus

$$\left\langle \Delta W_i^2 \right\rangle = \Delta t_i, \quad \text{and} \tag{2.17a}$$

$$\left\langle (\Delta W_i^2 - \Delta t_i)^2 \right\rangle = 2\Delta t_i^2. \tag{2.17b}$$

Using Riemann sums, the left hand side of equation (2.16) can be rewritten as

$$\lim_{n\to\infty}\left\langle\left(\sum_{i=1}^{n}G(t_{i-1})(\Delta W_i^2-\Delta t_i)\right)^2\right\rangle, \tag{2.18}$$

which, by developing the square and using the linearity of the expectation operator is equal to

$$\lim_{n\to\infty}\sum_{i=1}^{n}\left[\left\langle G^2(t_{i-1})(\Delta W_i^2-\Delta t_i)^2\right\rangle+\sum_{j=1}^{i-1}\left\langle 2G(t_{i-1})G(t_{j-1})(\Delta W_j^2-\Delta t_j)(\Delta W_i^2-\Delta t_i)\right\rangle\right]. \tag{2.19}$$

Since G is nonanticipating, we can use independence between terms to rewrite (2.19) as

$$\lim_{n\to\infty}\sum_{i=1}^{n}\left[\left\langle G^2(t_{i-1})\right\rangle\underbrace{\left\langle(\Delta W_i^2-\Delta t_i)^2\right\rangle}_{=2\Delta t_i^2}+\sum_{j=1}^{i-1}\left\langle 2G(t_{i-1})G(t_{j-1})(\Delta W_j^2-\Delta t_j)\right\rangle\underbrace{\left\langle(\Delta W_i^2-\Delta t_i)\right\rangle}_{=0}\right], \tag{2.20}$$

which simplifies to

$$2\lim_{n\to\infty}\sum_{i=1}^{n}\left\langle G^2(t_{i-1})\right\rangle\Delta t_i^2. \tag{2.21}$$

If G is bounded on $[t_0,\, t]$, the latter goes to zero, which concludes the proof. Since $[\mathrm{d}W(t)]^2$ only appears in the context of stochastic integration, property (2.15) is often written

$$[\mathrm{d}W(t)]^2=\mathrm{d}t, \tag{2.22}$$

but one must not forget the underlying meaning. It is important to remember that this property only holds in the context of Itô integration.

By a similar method, one can show that for any $N\geq 3$

$$[\mathrm{d}W(t)]^N=0, \tag{2.23}$$

and that for any $N_1\geq 1$, $N_2\geq 1$

$$[\mathrm{d}t]^{N_1}\,[\mathrm{d}W(t)]^{N_2}=0. \tag{2.24}$$

Those results are often summarized by saying that $\mathrm{d}W(t)$ is an infinitesimal of order $\frac{1}{2}$ in $\mathrm{d}t$ and that infinitesimals of order higher than 1 are discarded when it comes to compute differentials. Intuitively, $\mathrm{d}W(t)$ is a gaussian of variance $\mathrm{d}t$; a characteristic magnitude for $\mathrm{d}W(t)$ is its standard deviation, $\sqrt{\mathrm{d}t}$ which is indeed of order $\frac{1}{2}$.

The Itô formula

Consider an arbitrary function $\phi(x(t),t)$ with $x(t)$ obeying the Itô SDE (2.13). We are interested in the SDE governing ϕ. By definition,

$$\mathrm{d}\phi(x(t),t)=\phi(x(t)+\mathrm{d}x(t),t+\mathrm{d}t)-\phi(x(t),t). \tag{2.25}$$

Expanding $\phi(x(t)+\mathrm{d}x(t),t+\mathrm{d}t)$ in Taylor series and keeping the terms up to order 1 in $\mathrm{d}t$ yields

$$\begin{aligned}\mathrm{d}\phi(x(t),t)&=\frac{\partial\phi}{\partial t}\mathrm{d}t+\frac{\partial\phi}{\partial x}\mathrm{d}x(t)+\frac{1}{2}\frac{\partial^2\phi}{\partial x^2}[\mathrm{d}x(t)]^2+\dots\\&=\frac{\partial\phi}{\partial t}\mathrm{d}t+\frac{\partial\phi}{\partial x}\left(a(x(t),t)\mathrm{d}t+b(x(t),t)\mathrm{d}W(t)\right)+\frac{1}{2}\frac{\partial^2\phi}{\partial x^2}\left(b^2(x(t),t)[\mathrm{d}W(t)]^2+\dots\right)+\dots,\end{aligned}$$

where the derivatives are evaluated at $(x(t),t)$. Now, using that $[\mathrm{d}W(t)]^2=\mathrm{d}t$ we get the Itô formula:

$$\mathrm{d}\phi(x(t),t)=\left[\frac{\partial\phi}{\partial t}+\frac{\partial\phi}{\partial x}a(x(t),t)+\frac{1}{2}\frac{\partial^2\phi}{\partial x^2}b^2(x(t),t)\right]\mathrm{d}t+\frac{\partial\phi}{\partial x}b(x(t),t)\mathrm{d}W(t). \tag{2.26}$$

2.1.3 Link between Itô and backward Itô SDE's

A same stochastic process $x(t)$ can be described both by a Itô and by a backward Itô SDE. Suppose that $x(t)$ obeys the Itô SDE

$$dx(t) = a_I(x(t),t)\mathrm{d}t + b_I(x(t),t)\mathrm{d}W(t), \tag{2.27}$$

and the equivalent backward Itô SDE

$$dx(t) = a_{bI}(x(t),t)\mathrm{d}t + b_{bI}(x(t),t)\mathrm{d}W(t). \tag{2.28}$$

The goal is to compute the relations between the functions a_I, b_I and a_{bI}, b_{bI}. By (2.28),

$$x(t) = x(t_0) + \int_{t_0}^{t} a_{bI}(x(s),s)\mathrm{d}s + (\mathbf{bI})\int_{t_0}^{t} b_{bI}(x(t),t)\mathrm{d}W(t). \tag{2.29}$$

Let us rewrite the backward Itô stochastic integral term

$$(\mathbf{bI})\int_{t_0}^{t} b_{bI}(x(t),t)\mathrm{d}W(t) \triangleq \underset{n\to\infty}{\text{ms-lim}} \sum_{i=1}^{n} b_{bI}(x(t_i),t_i)[W(t_i) - W(t_{i-1})] \tag{2.30}$$

as an Itô integral. Since $x(t)$ satisfies the Itô SDE (2.27), we can apply Itô formula. This yields

$$\begin{aligned} b_{bI}(x(t_i),t_i) = {} & b_{bI}(x(t_{i-1}),t_{i-1}) \\ & + \left[\frac{\partial b_{bI}}{\partial t} + \frac{\partial b_{bI}}{\partial x} a_I(x(t_{i-1}),t_{i-1}) + \frac{1}{2}\frac{\partial^2 b_{bI}}{\partial x^2} b_I^2(x(t_{i-1}),t_{i-1})\right](t_i - t_{i-1}) \\ & + \frac{\partial b_{bI}}{\partial x} b_I(x(t_{i-1}),t_{i-1})(W(t_i) - W(t_{i-1})), \end{aligned} \tag{2.31}$$

where the derivatives are evaluated at $(x(t_{i-1}),t_{i-1})$. Introducing (2.31) in (2.30) and setting $[\mathrm{d}W(t)]^2 = \mathrm{d}t$ yields, after dropping the terms in $\mathrm{d}t\mathrm{d}W(t)$ and $\mathrm{d}t^2$:

$$\begin{aligned} (\mathbf{bI})\int_{t_0}^{t} b_{bI}(x(t),t)\mathrm{d}W(t) = {} & \underset{n\to\infty}{\text{ms-lim}} \sum_{i=1}^{n} b_{bI}(x(t_{i-1}),t_{i-1})[W(t_i) - W(t_{i-1})] \\ & + b_I(x(t_{i-1}),t_{i-1})\frac{\partial b_{bI}}{\partial x}[t_i - t_{i-1}], \end{aligned} \tag{2.32}$$

and finally

$$(\mathbf{bI})\int_{t_0}^{t} b_{bI}(x(t),t)\mathrm{d}W(t) = (\mathbf{I})\int_{t_0}^{t} b_{bI}(x(t),t)\mathrm{d}W(t) + \int_{t_0}^{t} b_I(x(s),s)\frac{\partial b_{bI}}{\partial x}(x(s),s)\mathrm{d}s. \tag{2.33}$$

Therefore, we have the equivalences

$$\begin{array}{ccc} \underline{\text{Itô SDE}} & & \underline{\text{backward Itô SDE}} \\ \mathrm{d}x(t) = a_I\mathrm{d}t + b_I\mathrm{d}W(t) & \Leftrightarrow & \mathrm{d}x(t) = \left[a_I - b_I\dfrac{\partial b_I}{\partial x}\right]\mathrm{d}t + b_I\mathrm{d}W(t) \end{array} \tag{2.34}$$

and conversely

$$\begin{array}{ccc} \underline{\text{backward Itô SDE}} & & \underline{\text{Itô SDE}} \\ \mathrm{d}x(t) = a_{bI}\mathrm{d}t + b_{bI}\mathrm{d}W(t) & \Leftrightarrow & \mathrm{d}x(t) = \left[a_{bI} + b_{bI}\dfrac{\partial b_{bI}}{\partial x}\right]\mathrm{d}t + b_{bI}\mathrm{d}W(t). \end{array} \tag{2.35}$$

Here, the dependence of a_I, b_I, a_{bI} and b_{bI} on $x(t)$ and t have been made implicit to simplify the notations.

2.1.4 Connection between Itô and backward Itô SDE's and the Fokker Planck equation

Consider a particle in one dimension whose position $x(t)$ obeys the Itô SDE

$$\begin{cases} \mathrm{d}x(t) = a(x(t),t)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t), \\ x(t_0) = x_0. \end{cases} \tag{2.36}$$

Let $p(x,t;y,s)$ be the probability density function of the position x of the particle at time t given that the particle was in position y at time s, with $s < t$. For an infinitesimal $\mathrm{d}x$ and $\bar{x} \in \Omega$, the probability that the random variable $x(t)$ describing the position of the particle at time t has value between $\bar{x}$ and $\bar{x} + \mathrm{d}x$ is given by:

$$\Pr\left(\bar{x} < x(t) < \bar{x} + \mathrm{d}x \,|\, x(s) = y\right) = p(\bar{x},t;y,s)\mathrm{d}x. \tag{2.37}$$

From (2.36) we are going the derive the partial differential equation governing the evolution of $p(x,t;x_0,t_0)$. Let $K(x)$ be an arbitrary function of $\mathcal{C}^2$ with compact support. In the next, all the derivative terms of functions a and b are evaluated at $(x(t),t)$ and the derivatives of K are evaluated at $x(t)$, unless otherwise stated. By Itô's formula:

$$\mathrm{d}K(x(t)) = \left[\frac{\partial K}{\partial x}a(x(t),t) + \frac{1}{2}b^2(x(t),t)\frac{\partial^2 K}{\partial x^2}\right]\mathrm{d}t + \frac{\partial K}{\partial x}b(x(t),t)\mathrm{d}W(t), \tag{2.38}$$

where the derivatives are evaluated at $(x(t),t)$. Taking the expectation yields

$$\langle \mathrm{d}K(x(t))\rangle = \left\langle \frac{\partial K}{\partial x}a(x(t),t) + \frac{1}{2}b^2(x(t),t)\frac{\partial^2 K}{\partial x^2}\right\rangle \mathrm{d}t, \tag{2.39}$$

and thus

$$\frac{d\,\langle K(x(t))\rangle}{dt} = \left\langle \frac{\partial K}{\partial x}a(x(t),t) + \frac{1}{2}b^2(x(t),t)\frac{\partial^2 K}{\partial x^2}\right\rangle. \tag{2.40}$$

Using the probability density $p(x,t)$, we can rewrite (2.40) as

$$\frac{d}{dt}\int_{-\infty}^{\infty} K(x)p(x,t;x_0,t_0)\mathrm{d}x = \underbrace{\int_{-\infty}^{\infty}\frac{\partial K}{\partial x}a(x,t)p(x,t;x_0,t_0)\mathrm{d}x}_{:=I_1} + \underbrace{\frac{1}{2}\int_{-\infty}^{\infty}\frac{\partial^2 K}{\partial x^2}b^2(x,t)p(x,t;x_0,t_0)\mathrm{d}x}_{:=I_2}. \tag{2.41}$$

Integrating I_1 by parts yields:

$$\begin{aligned} I_1 &= [a(x,t)p(x,t;x_0,t_0)K(x)]_{-\infty}^{\infty} - \int_{-\infty}^{\infty} K(x)\frac{\partial(ap)}{\partial x}\mathrm{d}x \\ &= -\int_{-\infty}^{\infty} K(x)\frac{\partial(ap)}{\partial x}\mathrm{d}x, \end{aligned} \tag{2.42}$$

where the second equality is obtained because K has a compact support. Integrating I_2 by parts yields successively:

$$\begin{aligned} I_2 &= \left[b^2(x,t)p(x,t;x_0,t_0)\frac{\partial K}{\partial x}\right]_{-\infty}^{\infty} - \int_{-\infty}^{\infty}\frac{\partial K}{\partial x}\frac{\partial(b^2p)}{\partial x} \\ &= b^2(x,t)p(x,t;x_0,t_0)\frac{\partial K}{\partial x}\bigg|_{-\infty}^{\infty} - \frac{\partial(b^2p)}{\partial x}K(x)\bigg|_{-\infty}^{\infty} + \int_{-\infty}^{\infty} K(x)\frac{\partial^2(b^2p)}{\partial x^2}\mathrm{d}x \\ &= \int_{-\infty}^{\infty} K(x)\frac{\partial^2(b^2p)}{\partial x^2}\mathrm{d}x. \end{aligned} \tag{2.43}$$

Again, the surface terms vanish because of the compact support of K. Inserting (2.42) and (2.43) in (2.41) and rearranging the terms yields

$$\int_{-\infty}^{\infty} K(x)\left(\frac{\partial p}{\partial t} + \frac{\partial (ap)}{\partial x} - \frac{1}{2}\frac{\partial^2 (b^2 p)}{\partial x^2}\right)\mathrm{d}x = 0. \tag{2.44}$$

Since K is arbitrary we must have that

$$\frac{\partial p}{\partial t} = -\frac{\partial (ap)}{\partial x} + \frac{1}{2}\frac{\partial^2 (b^2 p)}{\partial x^2}, \tag{2.45}$$

the Fokker-Planck equation (or Kolmogorov forward equation) corresponding to the Itô SDE (2.36).

Now suppose that the particle's position $x(t)$ obeys the backward Itô SDE

$$\begin{cases} \mathrm{d}x(t) = a(x(t),t)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t), \\ x(t_0) = x_0. \end{cases} \tag{2.46}$$

By (2.35), $x(t)$ is equivalently governed by the Itô SDE

$$\begin{cases} \mathrm{d}x(t) = \left(a(x(t),t) + b(x(t),t)\dfrac{\partial b}{\partial x}\right)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t) \\ x(t_0) = x_0. \end{cases} \tag{2.47}$$

By the above results, the probability density of $x(t)$ is governed by the partial differential equation

$$\frac{\partial p}{\partial t} = -\frac{\partial}{\partial x}\left[\left(a + b\frac{\partial b}{\partial x}\right)p\right] + \frac{1}{2}\frac{\partial^2 (b^2 p)}{\partial x^2}. \tag{2.48}$$

The latter can be simplified to

$$\frac{\partial p}{\partial t} = -\frac{\partial (ap)}{\partial x} + \frac{1}{2}\frac{\partial}{\partial x}\left(b^2\frac{\partial p}{\partial x}\right), \tag{2.49}$$

the Fokker-Planck equation corresponding to the backward Itô SDE (2.46).

2.1.5 Generalization to multiple dimensions

Here we generalize the results of previous section to the cases with n variables and m independent noise components. In that case, $\mathbf{x}(t)$ and $\mathbf{a}(\mathbf{x},t)$ are n-dimensional vectors, $\mathbf{B}(\mathbf{x},t)$ is a $n \times m$ matrix and $\mathbf{W}(t)$ is a m-dimensional multivariate Wiener process of mean $\mathbf{0}$. Hence, $\mathbf{W}(t) = [W_1(t), W_2(t), \ldots, W_m(t)]^\intercal$ is a vector of m *independent* Wiener processes such as defined in (2.5). The general form of a multidimensional SDE is then

$$\begin{cases} \mathrm{d}\mathbf{x}(t) = \mathbf{a}(\mathbf{x}(t),t)\mathrm{d}t + \mathbf{B}(\mathbf{x}(t),t)\mathrm{d}\mathbf{W}(t), \\ \mathbf{x}(t_0) = \mathbf{x}_0. \end{cases} \tag{2.50}$$

The differential rules are similar to the one-dimensional case. For $N \geq 3$, N_1, $N_2 \geq 1$ and $i, j \in \{1, 2, \ldots, m\}$:

$$\begin{cases} \mathrm{d}W_i(t)\mathrm{d}W_j(t) = \delta_{ij}\mathrm{d}t, & \text{(2.51a)} \\ [\mathrm{d}W_i(t)]^N = 0, & \text{(2.51b)} \\ [\mathrm{d}t]^{N_1}[\mathrm{d}W_i(t)]^{N_2} = 0. & \text{(2.51c)} \end{cases}$$

Itô's formula for a function ϕ of the n-dimensional vector $\mathbf{x}(t)$ satisfying the Itô SDE (2.50) is

$$\mathrm{d}f(\mathbf{x}) = \left[\sum_{i=1}^{n} a_i(\mathbf{x}(t),t)\frac{\partial f}{\partial x_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}[\mathbf{B}(\mathbf{x}(t),t)\mathbf{B}^{\intercal}(\mathbf{x}(t),t)]_{ij}\frac{\partial^2 f}{\partial x_i \partial x_j}\right]\mathrm{d}t + \sum_{i=1}^{n}\sum_{j=1}^{m}[\mathbf{B}(\mathbf{x}(t),t)]_{ij}\frac{\partial f}{\partial x_i}\mathrm{d}W_j(t), \tag{2.52}$$

where the derivatives of f are evaluated at $\mathbf{x}(t)$.

Let $\mathbf{D} = \mathbf{B}\mathbf{B}^{\intercal}$. If (2.50) is interpreted as an Itô SDE, the corresponding Fokker-Planck equation is

$$\frac{\partial p}{\partial t} = -\sum_{i=1}^{n}\frac{\partial (a_i p)}{\partial x_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2 \left([\mathbf{D}]_{ij} p\right)}{\partial x_i \partial x_j}. \tag{2.53}$$

If (2.50) is interpreted as a backward Itô SDE, the corresponding Fokker-Planck equation is

$$\frac{\partial p}{\partial t} = -\sum_{i=1}^{n}\frac{\partial (a_i p)}{\partial x_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial}{\partial x_i}\left([\mathbf{D}]_{ij}\frac{\partial p}{\partial x_j}\right). \tag{2.54}$$

In equations (2.53) and (2.54), all the derivatives are evaluated at $(\mathbf{x}(t), t)$.

Generally, if we have a Fokker-Planck equation and want to compute a corresponding SDE, we have to compute $\mathbf{B}$ from $\mathbf{D} = \mathbf{B}\mathbf{B}^{\intercal}$. The solution to that equation is not unique, so that SDE's with different $\mathbf{B}$ could be consistent with the same Fokker-Planck equation. If $\mathbf{D}$ is positive-semidefinite, the Cholesky decomposition *exists*, namely there is a lower triangular matrix $\mathbf{B}$ which is the solution to $\mathbf{D} = \mathbf{B}\mathbf{B}^{\intercal}$. Besides, if $\mathbf{D}$ is positive-definite, the Cholesky decomposition is *unique* if we require that the diagonal elements of $\mathbf{B}$ are strictly positive. This provides a way to compute $\mathbf{B}$ in the case of a positive-semidefinite matrix $\mathbf{D}$.

Remark The Itô and backward Itô SDE's considered in this section have exactly the same form, given by equation (2.50). However, if $\mathbf{B}$ is not constant, they correspond to different Fokker-Planck equations and thus to different *transport processes*. An important implication that does not arise in the context of deterministic integration is that the numerical method has to be chosen consistently with the type (Itô, backward Itô, Stratonovich, etc.) of the SDE. The derivation of simple numerical schemes consistent with either an Itô or a backward Itô SDE is the topic of the next section.

2.1.6 Numerical methods

We present here the *Euler* and *backward-Euler* methods, which are the simplest numerical schemes for the simulation of an Itô and a backward Itô stochastic process, respectively. The goal of this section is really to provide some intuition about why the *Euler method* is relevant to simulate an Itô process whereas the *backward-Euler method* is relevant in the context of a backward Itô process. We do not aim at providing fully rigorous proofs of convergence and consistency. Such a formalism can be found in the well-known book by Kloeden and Platen about stochastic numerical methods [15]. The backward-Euler method has been introduced more recently by LaBolle [16]. A more recent handbook about stochastic numerical methods is [17].

Consider the one-dimensional SDE

$$\begin{cases} \mathrm{d}x(t) = a(x(t),t)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t), \\ x(t_0) = x_0. \end{cases} \tag{2.55}$$

Through a numerical approximation of (2.55), we can only compute x at discrete times $t_0 < t_1 < \cdots < t_{n-1} < t_n = T$, where T is the final integration time. We consider a constant time step Δt such that $t_{i+1} = t_i + \Delta t$ for any $i \in \{0, 1, \ldots, n-1\}$. Let X_i denote the numerical approximation of $x(t_i)$, and let $\Delta W_i := W(t_{i+1}) - W(t_i) \sim \mathcal{N}(0, \Delta t)$. ΔW_i is thus a gaussian noise of mean 0 and variance Δt, and ΔW_i is independent of ΔW_j for any $i \neq j$. In the next, we shall only verify that the schemes are *consistent*, namely that they tend to the proper SDE when $t_i \to t$, $\Delta t \to \mathrm{d}t$ and $\Delta W_i \to \mathrm{d}W(t)$.

Let us first consider the case where (2.55) is a Itô SDE. Suppose that $x(t_i)$ is known, and we want to compute $x(t_{i+1})$. Now, consider a partition $t_i = t_{i_0} < t_{i_1} < \cdots < t_{i_{m-1}} < t_{i_m} = t_{i+1}$ of $[t_i,\, t_{i+1}]$. By (2.55) and the definition of the Itô stochastic integral:

$$x(t_{i+1}) = x(t_i) + \lim_{m\to\infty} \sum_{k=0}^{m-1} a\left(x(t_{i_k}), t_{i_k}\right)\left(t_{i_{k+1}} - t_{i_k}\right) + \operatorname*{ms-lim}_{m\to\infty} \sum_{k=0}^{m-1} b\left(x(t_{i_k}), t_{i_k}\right)\left[W(t_{i_{k+1}}) - W(t_{i_k})\right]. \tag{2.56}$$

the simplest approximation to that expression is to take $m = 1$. This gives precisely the *Euler method*, also called the *Euler-Maruyama method*:

$$X_{i+1} = X_i + a(X_i, t_i)\Delta t + b(X_i, t_i)\Delta W_i \tag{2.57}$$

Note that if $R_0, R_1, \ldots, R_{n-1}$ are independent standard gaussian random variables, then we can replace ΔW_i by $\sqrt{\Delta t} R_i$ which could be more practical to implement.

Now consider the case where (2.55) is a backward Itô SDE. A similar reasoning as set out above for the Itô case yields :

$$x(t_{i+1}) = x(t_i) + \lim_{m\to\infty} \sum_{k=0}^{m-1} a\left(x(t_{i_k}), t_{i_k}\right)\left(t_{i_{k+1}} - t_{i_k}\right) + \operatorname*{ms-lim}_{m\to\infty} \sum_{k=0}^{m-1} b\left(x(t_{i_{k+1}}), t_{i_k}\right)\left[W(t_{i_{k+1}}) - W(t_{i_k})\right]. \tag{2.58}$$

Notice that since the first limit corresponds to a deterministic integral, we can choose to evaluate a at any time in $[t_{i_k},\, t_{i_{k+1}}]$.[1] The fact that b is evaluated at time t_{i_k} follows from the implicit assumption that b is differentiable in t, cfr. the remark at page 11. Taking $m = 1$ yields the approximation

$$X_{i+1} = X_i + a(X_i, t_i)\Delta t + b(X_{i+1}, t_i)\Delta W_i, \tag{2.59}$$

which is an *implicit* scheme since b has to be evaluated at X_{i+1}. In the general case b is nonlinear in x and solving (2.59) is nontrivial. In particular, it is not always possible to invert b and hence to find an explicit formula for X_{i+1}. The idea is thus to rely on a predictor-corrector method: we first compute an estimate X^*_{i+1} of X_{i+1} using an explicit formula, and then we compute X_{i+1} as

$$X_{i+1} = X_i + a(X_i, t_i)\Delta t + b(X^*_{i+1}, t_i)\Delta W_i. \tag{2.60}$$

Now the question is: how to compute X^*_{i+1} ? One might be tempted to use the Euler method, which is explicit. However, we will see shortly that we do not need to include the advective transport term in the estimation X^*_{i+1} of X_{i+1}. The predictor-corrector scheme is thus

$$\begin{cases} \Delta Y_i = b(X_i, t_i)\Delta W_i, & (2.61a) \\ X_{i+1} = X_i + a(X_i, t_i)\Delta t + b(X_i + \Delta Y_i, t_i)\Delta W_i. & (2.61b) \end{cases}$$

In order to see that the scheme (2.61) consistently approximate the backward Itô SDE (2.55), we can consider a stochastic process $y(t)$ and interpret ΔY_i as the difference between two successive

[1] The choice a_{i_k} corresponds to the Darboux integration, which can be seen as a particular case of the Riemann integration. Remember that a function is Darboux-integrable if and only if it is Riemann-integrable, and the values of the two integrals, if they exist, are equal.

iterates: $\Delta Y_i = Y_{i+1} - Y_i$. But then, the scheme (2.61) is precisely the *Euler method* applied on the Itô SDE with two variables

$$\begin{cases} \mathrm{d}y(t) = b(x(t),t)\mathrm{d}W(t), & (2.62a) \\ \mathrm{d}x(t) = a(x(t),t)\mathrm{d}t + b(x(t)+\mathrm{d}y(t),t)\mathrm{d}W(t). & (2.62b) \end{cases}$$

Therefore, by the equivalence formula between Itô and backward Itô SDE's (2.35), showing that the scheme (2.61) is consistent with the backward Itô SDE (2.55) amounts to show that (2.62) is equivalent to the Itô SDE

$$\mathrm{d}x(t) = \left(a(x(t),t) + b(x(t),t)\frac{\partial b}{\partial x}\right)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t), \tag{2.63}$$

where $\partial_x b$ is evaluated at $(x(t),t)$. Using the same convention for all the derivatives of b, a stochastic Itô-Taylor expansion on $b(x(t)+\mathrm{d}y(t),t)$ yields

$$\begin{aligned} b(x(t)+\mathrm{d}y(t),t) &= b(x(t),t) + \frac{\partial b}{\partial x}\mathrm{d}y(t) + \frac{1}{2}\frac{\partial^2 b}{\partial x^2}[\mathrm{d}y(t)]^2 + \dots & (2.64a) \\ &= b(x(t),t) + \frac{\partial b}{\partial x}\mathrm{b}(x(t),t)\mathrm{d}W(t) + \frac{1}{2}\frac{\partial^2 b}{\partial x^2}b^2(x(t),t)[\mathrm{d}W(t)]^2 + \dots & (2.64b) \\ &= b(x(t),t) + \frac{\partial b}{\partial x}\mathrm{b}(x(t),t)\mathrm{d}W(t) + \frac{1}{2}\frac{\partial^2 b}{\partial x^2}b^2(x(t),t)\mathrm{d}t + \dots & (2.64c) \end{aligned}$$

Finally, introducing (2.64c) in (2.62b) yields

$$\begin{aligned} \mathrm{d}x(t) &= a(x(t),t)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t) + \frac{\partial b}{\partial x}\mathrm{b}(x(t),t)[\mathrm{d}W(t)]^2 + \frac{1}{2}\frac{\partial^2 b}{\partial x^2}b^2(x(t),t)\mathrm{d}t\mathrm{d}W(t), & (2.65a) \\ &= \left(a(x(t),t) + b(x(t),t)\frac{\partial b}{\partial x}\right)\mathrm{d}t + b(x(t),t)\mathrm{d}W(t), & (2.65b) \end{aligned}$$

where we have used the properties (2.22) and (2.24) of Itô integration. Since (2.65b) is exactly (2.63), we have shown the consistency of the backward-Euler method (2.61) for the numerical integration of the backward Itô SDE (2.55). Notice that including the advective transport in the computation of ΔY_i in equation (2.61a) leads to a term of order $\mathrm{d}t\mathrm{d}W$ in equation (2.65a). By (2.24), $\mathrm{d}t\mathrm{d}W = 0$ and this term does not appears in (2.65b).

Remark We have implicitly assumed through all the above developments that b is *smooth enough*, and the equivalence between the Itô and backward Itô formulation has only be proven in that case. In the case of discontinuous b, the consistency of the backward Itô formulation with the Fokker-Planck equation is shown in [16] for the one-dimensional case and demonstrated in the two-dimensional case. There is however no proof for the multi-dimensional case. The efficacy of the backward-Euler method in the case of discontinuous diffusivities is assessed in [4] on two one-dimensional test cases. The second test case is an advection-diffusion equation with constant velocity and a piecewise constant diffusivity (like the diffusivities of the overturner model). It shows a significantly better performance of the backward-Euler method with respect to Itô and Stratonovich methods in estimating the residence time of a tracer.

2.2 Lagrangian equations for the overturner model

Let us recall the transport equation derived in section (1.1.3) for the concentration of a passive tracer in the *overturner* model:

$$\frac{\partial C}{\partial t} = \nabla \cdot (-\mathbf{u}C + \mathbf{K}\nabla C). \tag{2.66}$$

This equation can be interpreted as a Fokker-Planck equation where C is the probability density function of the position $\mathbf{x}(t) = (y(t),\, z(t))$ of the particle. Equation (2.66) can be rewritten as

$$\frac{\partial C}{\partial t} = -\frac{\partial}{\partial y}\left[\left(v + \frac{\partial K_h}{\partial y}\right) C\right] - \frac{\partial}{\partial z}\left[\left(w + \frac{\partial K_{zz}}{\partial z}\right) C\right] + \frac{1}{2}\left[\frac{\partial^2}{\partial y^2}\left(2K_{yy}C\right) + \frac{\partial^2}{\partial z^2}\left(2K_v C\right)\right]. \quad (2.67)$$

This is precisely equation (2.53) with $\mathbf{x} = (y, z)$, $p = C$, $\mathbf{a} = (v + \partial_y K_h,\, w + \partial_z K_v)$ and $\mathbf{D} = 2\mathbf{K}$. Therefore, $\mathbf{x}(t) = (y(t),\, z(t))$ obeys the Itô SDE

$$\mathrm{d}\mathbf{x}(t) = \mathbf{a}(x(t), t)\mathrm{d}t + \mathbf{B}(x(t), t)\mathrm{d}\mathbf{W}(t), \quad (2.68)$$

where $\mathbf{B}$ has to be solved from $2\mathbf{K} = \mathbf{B}\mathbf{B}^{\intercal}$. Since $2\mathbf{K}$ is positive definite, the unique Cholesky decomposition yields

$$\mathbf{B} = \begin{pmatrix} \sqrt{2K_h} & 0 \\ 0 & \sqrt{2K_v} \end{pmatrix}, \quad (2.69)$$

and the Itô SDE (2.68) can be rewritten as

$$\begin{cases} \mathrm{d}y(t) = \left(v + \frac{\partial K_h}{\partial y}\right)\mathrm{d}t + \sqrt{2K_h}\mathrm{d}W_1(t) \\ \mathrm{d}z(t) = \left(w + \frac{\partial K_v}{\partial z}\right)\mathrm{d}t + \sqrt{2K_v}\mathrm{d}W_2(t) \\ (y(0), z(0)) = (y_0, z_0), \end{cases} \quad (2.70)$$

where $W_1(t)$ and $W_2(t)$ are independent Wiener processes. In our model, K_h is constant so that $\partial_y K_h = 0$ uniformly on Ω. The term gradient drift term $\partial_z K_v$ is more problematic: K_v is indeed discontinuous and $\partial_z K_v$ is infinite on the segment defined by $(\lambda y_0, z_0)$ with $\lambda \in [0, 1]$. Such a problem is addressed in [18] by neglecting the gradient drift terms all together, and in [19] by evaluating gradient drift terms via finite differences. Such methods are probably good enough for our simple overturner model.[2] However, we prefer the *backward-Euler* approach as this method applies to a wider range of problems with discontinuous diffusivities.

To derive the backward Itô SDE corresponding to the overturner transport model, notice that equation (2.66) can also be rewritten as

$$\frac{\partial C}{\partial t} = -\frac{\partial}{\partial y}(vC) - \frac{\partial}{\partial z}(wC) + \frac{1}{2}\left[\frac{\partial}{\partial y}\left(2K_h\frac{\partial C}{\partial y}\right) + \frac{\partial}{\partial z}\left(2K_v(y,z)\frac{\partial C}{\partial z}\right)\right]. \quad (2.71)$$

This is precisely equation (2.54) with $\mathbf{x} = (y, z)$, $p = C$, $\mathbf{a} = (v, w)$ and $\mathbf{D} = 2\mathbf{K}$. Hence, we can use the matrix $\mathbf{B}$ computed in (2.69). Then, $\mathbf{x}(t) = (y(t), z(t))$ also obeys the backward Itô SDE

$$\begin{cases} \mathrm{d}y(t) = v\mathrm{d}t + \sqrt{2K_h}\mathrm{d}W_1(t) \\ \mathrm{d}z(t) = w\mathrm{d}t + \sqrt{2K_v}\mathrm{d}W_2(t) \\ (y(0), z(0)) = (y_0, z_0). \end{cases} \quad (2.72)$$

Interestingly, there is no drift term, i.e. no derivative of the diffusivities. In the context of a problem with discontinuous diffusivities, the backward Itô interpretation is thus particularly interesting: see [16] and [4] for more complete discussions about the use of backward Itô method on problem with discontinuous diffusivities.

2.3 Test case to assess the implementation of the overturner model

This note is an adaptation of *Eric Deleersnijder*'s working paper [20].

[2] This is especially through since the discontinuity of the vertical diffusivity is an idealization of the reality. An estimation of $\partial_z K_v$ via finite differences near the discontinuities would thus be more realistic than the infinite value.

2.3.1 Governing equations

Let us consider a water domain, whose width is denoted $B(t, \mathbf{x})$, where t is the time and $\mathbf{x} = (y, z)$ is the position vector. The continuity equation is

$$\frac{\partial B}{\partial t} + \nabla \cdot (B\mathbf{u}) = 0, \tag{2.73}$$

where $\mathbf{u}(t, \mathbf{x})$ is the latitudinally-averaged meridional velocity. Assuming that mixing along the parallels is sufficiently efficient, we may study the concentration of a passive tracer by means of a two-dimensional model. The latitudinally-averaged concentration of the tracer $C(t, \mathbf{x})$ obeys the following partial differential equation :

$$\frac{\partial (BC)}{\partial t} + \nabla \cdot (B\mathbf{u}C) = Q\delta(\mathbf{x} - \mathbf{x}_1) + \nabla \cdot (B\mathbf{K} \cdot \nabla C), \tag{2.74}$$

where $\mathbf{K}$ is the diffusivity tensor (symmetric and positive definite); δ is the Dirac delta function with $\delta(\mathbf{x} - \mathbf{x}_n) = \delta(x - x_n)\delta(y - y_n)$; $Q(t)$ is the rate of release of a lineic source of length B along the latitude direction located at $\mathbf{x} = \mathbf{x}_1$. If $C(t, \mathbf{x})$ represents the density of the tracer in water, then $Q(t)$ is the mass of tracer released per second by the source.

Equation (2.74) is the so-called conservative form of the model. The convective form is obtained by combining equations (2.73) and (2.74):

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C = \frac{Q}{B}\delta(\mathbf{x} - \mathbf{x}_1) + \frac{1}{B}\nabla \cdot (B\mathbf{K} \cdot \nabla C). \tag{2.75}$$

2.3.2 An idealised model

For our test case to be interesting, we must be able to compute its analytical solution. Accordingly, we make some simplifying assumptions which will allow us to compute the solution analytically. First, we assume a constant width B and a constant velocity field

$$\mathbf{u}(t, \mathbf{x}) = v\mathbf{e}_y + w\mathbf{e}_z, \tag{2.76}$$

where $\mathbf{e}_y$ and $\mathbf{e}_z$ are the unit vectors associated respectively with the y- and z-coordinate axis. Furthermore, the diffusivity tensor is supposed constant and diagonal :

$$\mathbf{K} = \begin{pmatrix} K_{yy} & 0 \\ 0 & K_{zz} \end{pmatrix}, \tag{2.77}$$

where $K_{yy},\ K_{zz} > 0$. Finally, we consider a sudden pointwise release of tracer at $t = 0$. Hence, $Q(t)$ is of the form :

$$Q(t) = M\delta(t), \tag{2.78}$$

where M is the mass of tracer released at $t = 0$.

Under these assumptions, equation (2.75) simplifies to :

$$\frac{\partial C}{\partial t} + v\frac{\partial C}{\partial y} + w\frac{\partial C}{\partial z} = J\delta(t)\delta(y - y_1)\delta(z - z_1) + K_{yy}\frac{\partial^2 C}{\partial y^2} + K_{zz}\frac{\partial^2 C}{\partial z^2}, \tag{2.79}$$

where $J := M/B$. For the sake of simplicity, we can forget about the fact that our model is width-integrated and consider that it is a purely two-dimensional model with a point-source

$$Q := J\delta(t). \tag{2.80}$$

A part of the physical meaning of the model is lost but this makes representations of the problem easier. C now represents the two-dimensional density (i.e., in $[kg/m^2]$) of the tracer in water. J can then be regarded as the mass of tracer released by the sudden point source at $\mathbf{x} = \mathbf{x}_1$. The three- and two-dimensional interpretations of the problem are represented on figure 2.1.

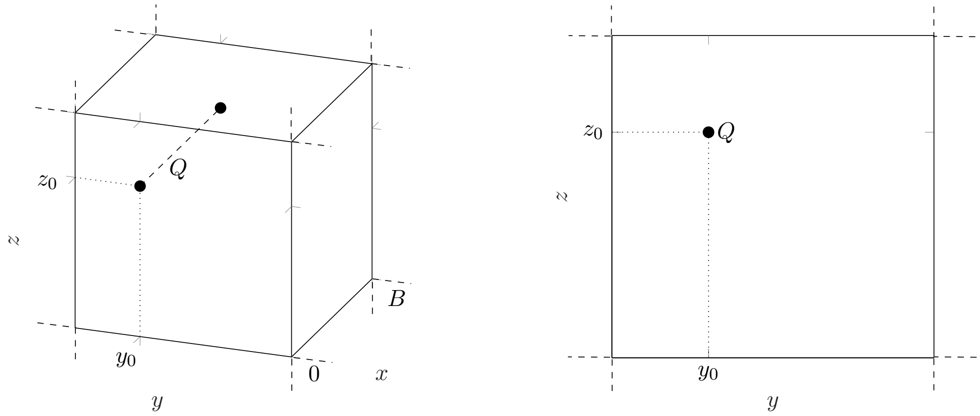


Figure 2.1 – Illustration of the 3D and 2D interpretations of the model.

Test case 1 : infinite domain

The first (an most simple) test case is to consider an infinite domain, i.e.

$$-\infty < y,\, z < \infty, \tag{2.81}$$

with nonzero velocities v and w. This test case provides a check that our numerical implementation handles the diffusion and advection processes properly both in the y- and z-directions. The parameters are chosen from the values of the overturner model :

$$v = \frac{\Psi}{H} = 4\times10^{-4}\ [m/s], \quad w = \frac{\Psi}{L} = 1.33\times10^{-7}\ [m/s], \tag{2.82}$$

and

$$K_{yy} = K_h = 10^3\ [m^2/s], \quad K_{zz} = K_{v_2} = 10^{-4}\ [m^2/s]. \tag{2.83}$$

For the length scales of the overturner model, those diffusivities corresponds to Péclet numbers

$$Pe_y = \frac{v}{K_{yy}/L} = 6, \quad Pe_z = \frac{w}{K_{zz}/H} = 6.67. \tag{2.84}$$

Hence, in both the y- and z-directions, the transport is neither dominated by advection nor by diffusion. This is interesting as a test case since it allows to assess how the numerical solver handles both physical processes in both directions. Finally, $J = 50\,000$ particles are released at $t = 0$ at the location $(y_1, z_1) = (0, 0)$.

Test case 2 : semi-infinite domain

Another interesting case is to consider a semi-infinite domain with a wall at $z = 0$:

$$-\infty < y < \infty, \quad 0 < z < \infty. \tag{2.85}$$

This is useful to assess how our numerical model handles no-through boundary conditions. Again, the parameters values are related to the ones from the overturner model :

$$v = \frac{\Psi}{H} = 4\times10^{-4}\ [m/s], \quad w = 0\ [m/s], \tag{2.86}$$

and

$$K_{yy} = K_h = 10^3\ [m^2/s], \quad K_{zz} = K_{v_1} = 10^{-1}\ [m^2/s]. \tag{2.87}$$

Notice the choice of K_{zz} : it is chosen equal to K_{v_1}, which is 10^3 times larger than K_{v_2}, the value chosen for test case 1. The goal here is to assess that the boundary condition is well handled by the solver. Since $w = 0$, only the vertical diffusivity could possibly drive the particles towards the wall. By increasing K_{zz}, we ensure that more particles will bounce against the wall, which is relevant in this context. $J = 50\,000$ particles are released at $t = 0$ at the location $(y_1, z_1) = (0, H)$.

2.3.3 Analytical solution and properties

Green's function

In order to build the analytical solution of the problem, we need to compute the Green's function associated to this particular problem. We derive the Green's function G associated to test case 1. We will show later how this function can be used to compute the concentration for both test case 1 and test case 2. $G(t, t', \mathbf{x})$ is zero for $t < t'$ and is the solution of

$$\begin{cases} \frac{\partial G}{\partial t} + v\frac{\partial G}{\partial y} + w\frac{\partial G}{\partial z} = K_{yy}\frac{\partial^2 G}{\partial y^2} + K_{zz}\frac{\partial^2 G}{\partial z^2} \\ G(t, t', y, z)|_{t=t'} = \delta(y)\delta(z) \end{cases} \tag{2.88}$$

for $t \geq 0$, and on an infinite domain $\infty < y,\ z < \infty$. It can be shown that source ? ou le papier de deleersnijder suffit ?

$$G(t, t', y, z) = \frac{\exp\left[-\frac{(y-s_v)^2}{4K_{yy}\tau} - \frac{(z-s_w)^2}{4K_{zz}\tau}\right]}{4\pi\sqrt{K_{yy}K_{zz}}\tau}, \tag{2.89}$$

where $\tau = t - t'$ and

$$\mathbf{s}(t, t') = (s_v(t, t'), s_w(t, t')) = \left(\int_{t'}^{t} v\mathrm{d}\xi, \int_{t'}^{t} w\mathrm{d}\xi\right) = (v\tau, w\tau)\,. \tag{2.90}$$

G has some interesting properties. The "mass" of the solution is

$$m(t, t') \equiv \int_{\mathbb{R}^2} G(t, t', \mathbf{x})\mathrm{d}\mathbf{x} = 1. \tag{2.91}$$

The "center of mass" is located at

$$\mathbf{r}(t, t') \equiv \frac{1}{m(t, t')}\int_{\mathbb{R}^2} \mathbf{x}G(t, t', \mathbf{x})\mathrm{d}\mathbf{x} = \mathbf{s}(t, t'). \tag{2.92}$$

The variance of the solution is

$$\sigma^2(t, t') \equiv \frac{1}{m(t, t')}\int_{\mathbb{R}^2} |\mathbf{x} - \mathbf{r}(t, t')|^2 G(t, t', \mathbf{x})\mathrm{d}\mathbf{x} = 2(K_{yy} + K_{zz})\tau. \tag{2.93}$$

Test case 1

The analytical solution of test case 1 is now obtained with the help of the Green's function derived above by computing the convolution between G and the source terms :

$$\begin{aligned} C(t, \mathbf{x}) &= \int_0^t \int_{\mathbb{R}^2} G(t, t', \mathbf{x} - \mathbf{x}')J\delta(t)\delta(\mathbf{x} - \mathbf{x}_1)\mathrm{d}\mathbf{x}'\mathrm{d}t' \\ &= JG(t, 0, \mathbf{x} - \mathbf{x}_1). \end{aligned} \tag{2.94}$$

The concentration profile for test case 1 is thus

$$\frac{J}{4\pi\sqrt{K_{yy}K_{zz}}t}\exp\left[-\frac{(y-s_v)^2}{4K_{yy}t}-\frac{(z-s_w)^2}{4K_{zz}t}\right]. \tag{2.95}$$

The total mass of tracer present in the domain is

$$m(t)\equiv\int_{\mathbb{R}^2}C(t,\mathbf{x})\mathrm{d}\mathbf{x}=J. \tag{2.96}$$

Note that this number is independant of the transport processes.

The mass center is located at

$$\begin{aligned}\mathbf{r}(t)&\equiv\frac{1}{m(t)}\int_{\mathbb{R}^2}\mathbf{x}C(t,\mathbf{x})\mathrm{d}\mathbf{x}\\&=\int_{\mathbb{R}^2}\mathbf{x}G(t,0,\mathbf{x}-\mathbf{x}_1)\mathrm{d}\mathbf{x}\\&=\int_{\mathbb{R}^2}(\mathbf{x}-\mathbf{x}_1)G(t,0,\mathbf{x}-\mathbf{x}_1)+\mathbf{x}_1G(t,0,\mathbf{x}-\mathbf{x}_1)\mathrm{d}\mathbf{x}\\&=\mathbf{x}_1+\mathbf{s}(t,0),\end{aligned} \tag{2.97}$$

where properties (2.91) and (2.92) are used to perform the last step.

Finally, the variance of the solution is

$$\begin{aligned}\sigma^2(t)&=\frac{1}{m(t)}\int_{\mathbb{R}^2}|\mathbf{x}-\mathbf{r}(t)|^2C(t,\mathbf{x})\mathrm{d}\mathbf{x}\\&=\int_{\mathbb{R}^2}|(\mathbf{x}-\mathbf{x}_1)-\mathbf{s}(t,0)|^2G(t,0,\mathbf{x}-\mathbf{x}_1)\mathrm{d}\mathbf{x}\\&=2(K_{yy}+K_{zz})t,\end{aligned} \tag{2.98}$$

where property (2.93) is used.

Test case 2

To compute the solution to test case 2, a little trick must be applied. Consider the problem on an infinite domain with two sudden point sources of equal intensity located at $z=H$ and $z=-H$. By symmetry, one can see that the concentration of that problem in the region $[-\infty,\infty]\times[0,\infty]$ is precisely the concentration of test case 2. Hence, we can use the Green's function G derived or test case 1 to compute the concentration. In this case, the convolution has to be performed with two point sources :

$$\begin{aligned}C(t,\mathbf{x})&=\int_0^t\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}J\delta(t')\delta(y'-y_1)\left[\delta(z'-z_1)+\delta(z'+z_1)\right]G(t,t',y-y',z-z')\mathrm{d}y'\mathrm{d}z'\mathrm{d}t'\\&=J\int_0^t\delta(t')[G(t,t',y-y_1,z-z_1)+G(t,t',y-y_1,z+z_1)]\\&=J[G(t,0,y-y_1,z-z_1)+G(t,0,y-y_1,z+z_1)].\end{aligned} \tag{2.99}$$

The concentration of the tracer for test case 2 is thus

$$C(t,y,z)=\frac{J}{4\pi\sqrt{K_{yy}K_{zz}}t}\exp\left[-\frac{(y-s_v)^2}{4K_{yy}t}\right]\left\{\exp\left[-\frac{(z-z_1)^2}{4K_{zz}t}\right]+\exp\left[-\frac{(z+z_1)^2}{4K_{zz}t}\right]\right\} \tag{2.100}$$

The mass is obtained as

$$m(t) \equiv \int_0^\infty \int_{-\infty}^\infty C(t,y,z)\mathrm{d}y\mathrm{d}z = J, \tag{2.101}$$

i.e. the number of particles released at $t = 0$. This result is obvious since there is no other source or sink, and we impose a no-through condition at the boundary.

NOTE : Le calcul du centre de masse me pose problème dans la direction z. On trouve assez facilement que $r_y = y_1 + s_v = y_1 + vt$. Par contre dans la direction z, il faut calculer

$$\int_0^\infty z\left\{\exp\left[-\frac{(z-z_1)^2}{4K_{zz}t}\right] + \exp\left[-\frac{(z+z_1)^2}{4K_{zz}t}\right]\right\}\mathrm{d}z.$$

Pas sûr qu'une solution simple existe... Je m'attends à une expression du type "H + (...)", où (...) est un terme qui permet de tenir compte du décalage du centre de masse selon z à cause du rebond sur le mur, et qui serait donc une fonction de K_{zz}, t et z_1.
Cependant, le calcul du centre de masse ne me parait pas indispensable et même si cela m'intrigue, j'ai du m'efforcer de lacher l'affaire pour ne pas passer la journée dessus. (D'autant plus que je peux les approximer numériquement à partir de l'expression analytique de la concentration.)

2.4 Validation of the numerical solver

This section aims to show that the numerical results obtained with the solver are in good agreement with the analytical ones. As the combination of the two test cases cover the main features of the overturner model, this constitutes a validation of the solver.

Both test cases are simulated for 1 year with a time step of 1 hour, and $J = 50\,000$ il faudrait tester avec 10 000 particles are released at $t = 0$. The concentration is computed at the final time $T = 1$ year on the domain $[y_{min}, y_{max}] \times [z_{min}, z_{max}]$, where the subscripts *min* and *max* stands for the minimal and maximal position at time T amongst all the particles. The notation $C(y,z)$ will be used to denote $C(T,y,z)$ in the next. That domain is divided into 20×20 boxes, and the (normalized) concentration in a box is computed as the number of particles in that box divided by the total number of particles J.

2.4.1 Test case 1

Figure 2.2 shows a comparison between the numerical result and the analytical solution for the (normalized) concentrations. Dark (resp. light) shaded areas correspond to zones where the numerically computed concentration is "above" (resp. "below") the exact concentration. Figures 2.3a and 2.3b represent respectively a cut of the concentrations at fixed $y = r_{y,exact}$ and at fixed $z = r_{z,exact}$. The numerically computed concentration C_{num} seems to be an appreciable approximation of the exact concentration C_{exact}. To be more specific, the maximal local error is

$$\|C_{exact} - C_{num}\|_\infty = 1.037 \times 10^{-3}. \tag{2.102}$$

The centers of mass are located at

$$\mathbf{r}_{exact} = (12\,614.4,\ 4.2)\ [m], \quad \mathbf{r}_{num} = (10\,985.3,\ 4.4)\ [m]. \tag{2.103}$$

The relative error is

$$\mathbf{e}_r = \left|\frac{\mathbf{r}_{exact} - \mathbf{r}_{num}}{\mathbf{r}_{exact}}\right| = \left(1.29 \times 10^{-1}, 4.75 \times 10^{-2}\right), \tag{2.104}$$

where the division is taken element-wise on the vectors. The 2-norm of the relative error is

$$\|\mathbf{e}_r\|_2 = 1.38\times 10^{-1}. \tag{2.105}$$

This can be seen as a quantification of the error on advection. To quantify the error on diffusion, we compute the variance of the concentration :

$$\sigma^2_{exact} = 6.31\times 10^{10}\ [m^2], \quad \sigma^2_{num} = 6.35\times 10^{10}\ [m^2]. \tag{2.106}$$

The relative error is

$$e_{\sigma^2} = \left|\frac{\sigma^2_{exact} - \sigma^2_{num}}{\sigma^2_{exact}}\right| = 7.26\times 10^{-3}. \tag{2.107}$$

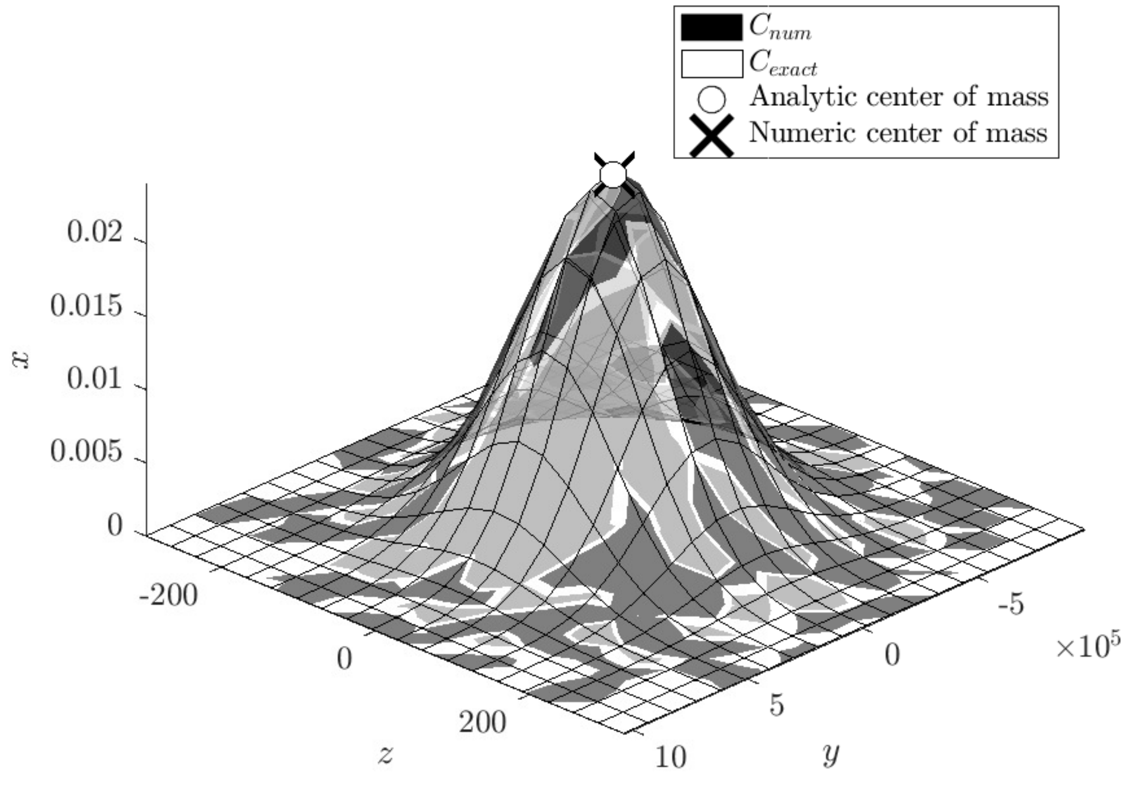
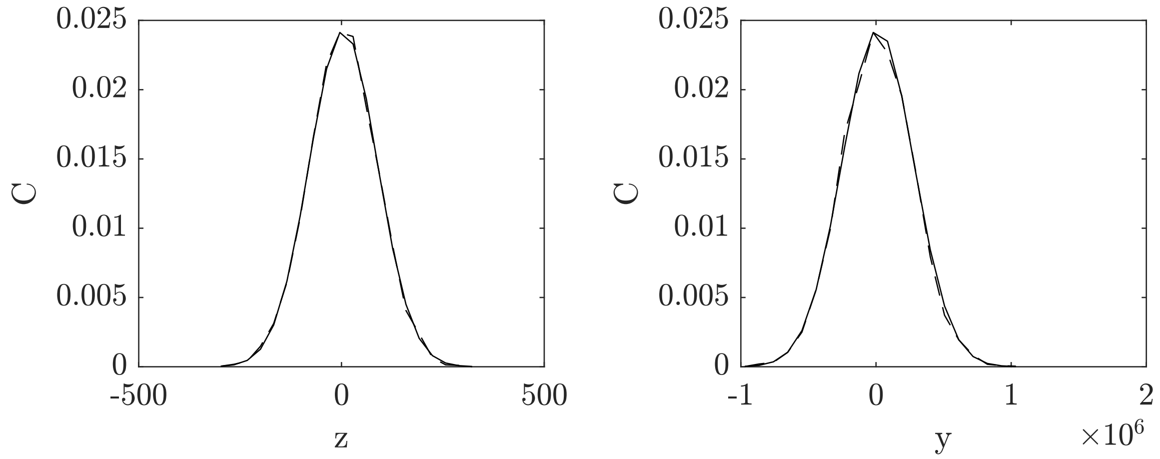


Figure 2.2 – Comparison of the concentrations obtained analytically and numerically. The "centers of mass" of the concentration obtained numerically (black cross) and numerically (white bullet) are also shown on the figure.



(a) $C_{num}(y_1 + vT, z)$ and $C_{exact}(y_1 + vT, z)$. **(b)** $C_{num}(y, z_1 + wT)$ and $C_{exact}(y, z_1 + wT)$.

Figure 2.3 – Cut of the concentrations at fixed $y = r_{y,exact}$ and at fixed $z = r_{z,exact}$. The dashed line represent C_{num} and the continuous line is for C_{exact}.

2.4.2 Test case 2

Figure 2.4 shows a comparison between the numerical result and the analytical solution for the (normalized) concentrations for test case 2. Since we do not have analytical expressions of the center of mass and of the variance for this test case, the "exact" values are approximated numerically thanks to the analytical expression of the concentration. Figures 2.5a and 2.5b represent respectively a cut of the concentrations at fixed $y = r_{y,exact}$ and along the boundary $z = 0$. The big picture about test case 2 is the presence of a boundary with no-through condition. A first verification is to check if all the particles are still in the domain, which is indeed the case. Although this might seem trivial here, it is sometimes a real challenge to ensure that no particle crosses the boundary, especially for geometrically complex domains. Besides, one can see on figure 2.5b that the concentration profile is well approximated along the boundary. Indeed, the maximal local error at the boundary is

$$\|C_{exact}(y, 0) - C_{num}(y, 0)\|_\infty = 3.53 \times 10^{-4}, \tag{2.108}$$

The maximal local error on the whole domain is

$$\|C_{exact} - C_{num}\|_\infty = 1.097 \times 10^{-3}. \tag{2.109}$$

The centers of mass are located at

$$\mathbf{r}_{exact} = (1.26 \times 10^4, 5.06 \times 10^3)\ [m], \quad \mathbf{r}_{num} = (1.10 \times 10^4, 5.05 \times 10^3)\ [m]. \tag{2.110}$$

The relative error is

$$\mathbf{e}_r = \left| \frac{\mathbf{r}_{exact} - \mathbf{r}_{num}}{\mathbf{r}_{exact}} \right| = \left(1.29 \times 10^{-1}, 1.825 \times 10^{-3}\right). \tag{2.111}$$

The 2-norm of the relative error is

$$\|\mathbf{e}_r\|_2 = 1.29 \times 10^{-1}. \tag{2.112}$$

This can be seen as a quantification of the error on advection. To quantify the error on diffusion, we compute the variance of the concentration :

$$\sigma^2_{exact} = 6.39 \times 10^{10}\ [m^2], \quad \sigma^2_{num} = 6.35 \times 10^{10}\ [m^2]. \tag{2.113}$$

The relative error is

$$e_{\sigma^2} = \left| \frac{\sigma^2_{exact} - \sigma^2_{num}}{\sigma^2_{exact}} \right| = 4.85 \times 10^{-3}. \tag{2.114}$$

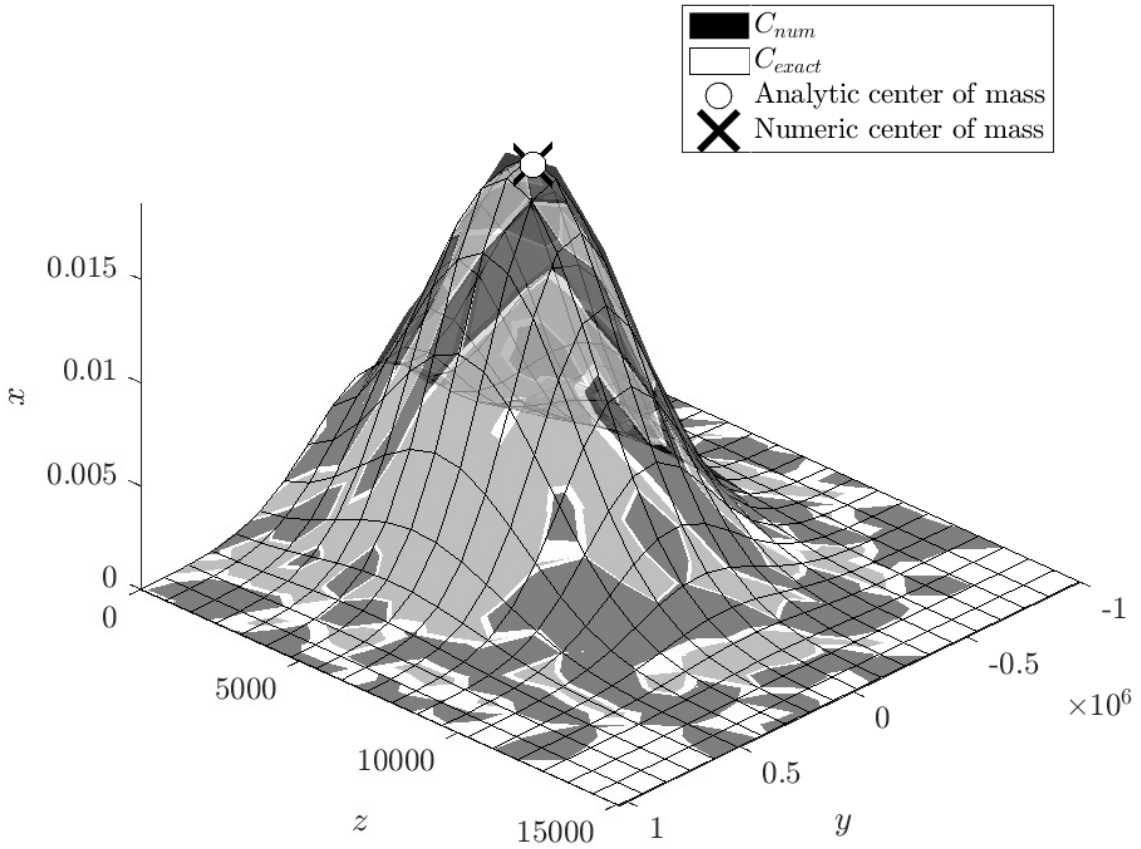
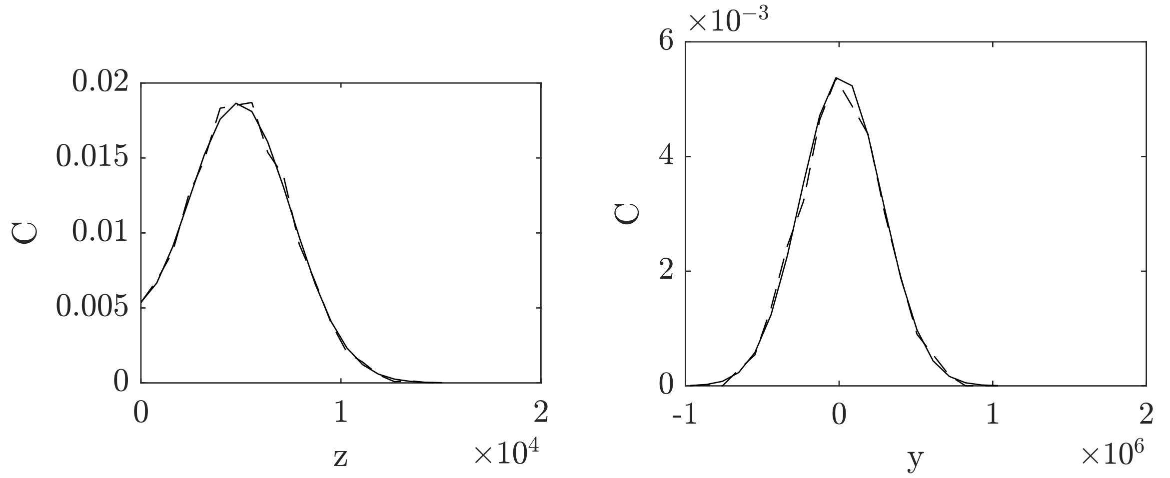


Figure 2.4 – Comparison of the concentrations obtained analytically and numerically. The "centers of mass" of the concentration obtained numerically (black cross) and numerically (white bullet) are also shown on the figure.



(a) $C_{num}(y_1 + vT, z)$ and $C_{exact}(y_1 + vT, z)$.

(b) $C_{num}(y, z_1 + wT)$ and $C_{exact}(y, z_1 + wT)$.

Figure 2.5 – Cut of the concentrations at fixed $y = r_{y,exact}$ and at fixed $z = r_{z,exact}$. The dashed line represent C_{num} and the continuous line is for C_{exact}.

Chapter 3

Clustering

3.1 The stability criterion for graph communities

The partition of a graph into communities (or clusters) as been widely studies those last two decades. Clustering comes indeed pretty handy to gain insight into the underlying structure of a system represented by a network. In some cases one can even build a simplified functional description based on the clusters. Many partitioning methods have been proposed, each relying on a particular measure to quantify the quality of a community structure. Such methods include normalized cut, (α,ϵ) clustering or modularity and its variants and extensions. See for instance [21] for a 2010 survey. In this work, we choose the stability approach, which is based on the statistical properties of a dynamical process taking place on the network. This approach was initially presented in [22] and further expended in [23] and [24].

The stability method presents a number of advantages. First, it does not require the number of communities to be specified beforehand, ensuring a natural partitioning of the graph. Second, it is flexible in the sense that it does not seek a *unique* optimal partition. Instead, it reveals several community structures, each appearing to be the most relevant at particular values of the Markov time: at a given time scale, natural clusters corresponds to sets of states from which escape is unlikely within that time scale. The stability method provides thus a dynamical interpretation of the partitioning problem. The Markov time acts as an intrinsic resolution parameter, as will be developed shortly. Finally, it is probably the most unifying approach since many of the standard partitioning measures find an interpretation through the stability framework.

In order to compute stability partitions in the next of this work, we make use of Michael Schaub's free software *PartitionStability*. This C++ implementation of the stability method with a MATLAB© interface is available at `https://github.com/michaelschaub/PartitionStability`. It relies on the Louvain algorithm [25] to optimize the stability quality function. This heuristic algorithm has been initially developed for modularity optimization. However one can show that stability can be written as the *modularity* of a time-dependent network evolving under the Markov process [23]. Hence, the Louvain method can almost straightforwardly be applied to stability optimization.

This section is devoted to the explanation of the stability measure, and how to find good clusterings using stability analysis. This theoretical part is intended to cover everything that is needed to make a proper, informed use of the stability toolbox. The stability measure has initially been presented for discrete times in [22]. We follow the same approach here: discrete-time stability is developed in the first part of this section; it is then extended to continuous time in a second part; finally, a few tools to analyze the robustness of a partition are presented in the third part of the section.

3.1.1 Discrete-time stability as an autocovariance

The stability criterion is based on the two-way relationship between graphs and Markov chains: On one hand, any graph has an associated Markov chain where the states are the nodes of the graph and the transitions probabilities between states are given by the weights of the edges. On the other hand, any Markov chain can be represented by a graph whose edges are weighted according to the transition probabilities. Concretely, consider a graph of N nodes whose $N \times N$ weighted adjacency matrix is denoted $\mathbf{A}$. Let $\mathbf{k} = \mathbf{A1}$; k_i is thus the total weight of the outgoing edges from node i. Let $\mathbf{K} = \mathrm{diag}(\mathbf{k})$. Then, by normalizing the rows of $\mathbf{A}$ we get the matrix $\mathbf{M} = \mathbf{K}^{-1}\mathbf{A}$, the transition probability matrix. $\mathbf{M}$ is row-stochastic (or right-stochastic) and $[\mathbf{M}]_{ij}$ is the probability to go from node i to node j. Consider a particle moving in the network according to the transition probabilities in $\mathbf{M}$. Now let $\mathbf{p}_t$ be the $1 \times N$ probability vector at Markov time t, namely that $p_{t,i}$ is the probability that the particle is located in node i at time t. The dynamics of the discrete-time Markov process are given by :

$$\mathbf{p}_{t+1} = \mathbf{p}_t \mathbf{K}^{-1}\mathbf{A} = \mathbf{p}_t \mathbf{M}. \tag{3.1}$$

Now, suppose that the Markov chain is ergodic, i.e. that it is possible to go from every state to every state and that the Markov process is aperiodic. The ergodicity assumption implies that any initial state will asymptotically reach the same stationary solution. Let $\boldsymbol{\pi}$ be that stationary distribution, given by $\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{M}$, and $\boldsymbol{\Pi} = \mathrm{diag}(\boldsymbol{\pi})$. Now, let $\mathbf{x}_t$ be the N-dimensional random indicator vector describing the position of a particle undergoing the above dynamics : $x_{t,i} = 1$ if the particle is located in node i at time t, and 0 otherwise. At stationarity, the *autocovariance matrix* of $\mathbf{x}$ is

$$\begin{aligned}
\mathbf{C}(\mathbf{x}_\tau, \mathbf{x}_{\tau+t}) &\triangleq \mathbb{E}\left[(\mathbf{x}_\tau - \mathbb{E}[\mathbf{x}_\tau])^\intercal(\mathbf{x}_{\tau+t} - \mathbb{E}[\mathbf{x}_{\tau+t}])\right] && (3.2)\\
&= \mathbb{E}\left[(\mathbf{x}_\tau - \boldsymbol{\pi})^\intercal(\mathbf{x}_{\tau+t} - \boldsymbol{\pi})\right] && (3.3)\\
&= \mathbb{E}\left[\mathbf{x}_\tau^\intercal \mathbf{x}_{\tau+t}\right] - \mathbb{E}[\mathbf{x}_\tau^\intercal]\boldsymbol{\pi} - \boldsymbol{\pi}^\intercal\mathbb{E}\left[\mathbf{x}_{\tau+t}\right] + \boldsymbol{\pi}^\intercal\boldsymbol{\pi} && (3.4)\\
&= \boldsymbol{\Pi}\mathbf{M}^t - \boldsymbol{\pi}^\intercal\boldsymbol{\pi}, && (3.5)
\end{aligned}$$

where the fact that $\mathbf{C}(\mathbf{x}_\tau, \mathbf{x}_{\tau+t})$ only depends on the time difference t at stationarity is readily verified. Here, $^\intercal$ is the transposed sign and $\mathbf{M}^t$ is $\mathbf{M}$ at the power t. $[\mathbf{C}(\mathbf{x}_\tau, \mathbf{x}_{\tau+t})]_{ij}$ is interpreted as the correlation between $\mathbf{x}_{\tau,i}$ and $\mathbf{x}_{\tau+t,j}$. The independence on the initial time τ implies that it can indifferently be chosen equal to 0.

Suppose now a partition $\mathcal{P}$; we note $\mathbf{H}_\mathcal{P}$ the indicator matrix of $\mathcal{P}$. If c is the number of communities in $\mathcal{P}$, $\mathbf{H}_\mathcal{P}$ is a binary $N \times c$ matrix such that

$$[\mathbf{H}_\mathcal{P}]_{ik} = \begin{cases} 1 & \text{if node } i \text{ is in community } k, \\ 0 & \text{otherwise.} \end{cases} \tag{3.6}$$

Let us define $\mathcal{H}_\mathcal{P} : \mathbb{R}^{N\times N} \to \mathbb{R}^{c\times c} : \mathbf{B} \mapsto \mathcal{H}(\mathbf{B}) = \mathbf{H}_\mathcal{P}^\intercal\mathbf{B}\mathbf{H}_\mathcal{P}$. Let $\mathbf{X}$ be any $N \times N$ matrix, then $\mathbf{Y} = \mathcal{H}_\mathcal{P}(\mathbf{X})$ is a $c \times c$ matrix such that $[\mathbf{Y}]_{kl} = \sum_{i\in\mathcal{C}_k, j\in\mathcal{C}_l}[\mathbf{X}]_{ij}$, where $\mathcal{C}_k$ and $\mathcal{C}_l$ denote communities k and l of partition $\mathcal{P}$. One could thus say that operator $\mathcal{H}_\mathcal{P}$ returns the *clustered version* of any $N \times N$ matrix, namely the matrix where the contributions of every nodes belonging to the same community are gathered by summing them. Finally, let $\mathbf{y}_t = \mathbf{H}_\mathcal{P}^\intercal\mathbf{x}_t$ denote the c-dimensional community indicator vector: $\mathbf{y}_{t,k}$ is equal to 1 if the particle is in community k at time t and zero otherwise. Using those notations and the interpretation of $\mathcal{H}_\mathcal{P}$, the *clustered* *autocovariance matrix* for partition $\mathcal{P}$ at time t is defined as

$$\mathbf{R}_t(\mathcal{P}) = \mathcal{H}_{\mathcal{P}}\left(\mathbf{C}(\mathbf{x}_\tau, \mathbf{x}_{\tau+t})\right) \tag{3.7}$$

$$= \mathbf{C}(\mathbf{y}_\tau, \mathbf{y}_{\tau+t}) \tag{3.8}$$

$$= \mathbf{H}_{\mathcal{P}}^\intercal(\mathbf{\Pi}\mathbf{M}^t - \boldsymbol{\pi}^\intercal\boldsymbol{\pi})\mathbf{H}_{\mathcal{P}}. \tag{3.9}$$

Notice that $\mathbf{R}_t$ depends only on the topology of the graph and on the partition. If the graph has well defined communities given by $\mathcal{P}$ *over a given time scale*, we expect that the particle is more likely to remain within the starting community overt that time scale. This implies that the values of $\mathbf{y}_{0,i}$ and $\mathbf{y}_{t,i}$ are positively correlated for t in that time scale, which in turn implies large diagonal elements in $\mathbf{R}_t(\mathcal{P})$ and hence a large trace of $\mathbf{R}_t(\mathcal{P})$. The elements of $\mathbf{R}_t(\mathcal{P})$ are interpreted as follows in terms of the random walk of a particle : $[\mathbf{R}_t(\mathcal{P})]_{kl}$ is the probability that a particle is in community $\mathcal{C}_l$ after t discrete time-steps if it has started in $\mathcal{C}_k$ minus the probability that two independent random walkers are in $\mathcal{C}_k$ and $\mathcal{C}_l$, evaluated at stationarity. A good partition is such that there is a high likelihood of remaining in the starting community over a given time scale. The definition of the stability of a *clustering* $\mathcal{P}$ follows naturally:

$$r_t(\mathcal{P}) = \min_{0\le s\le t} \sum_{i=1}^{c} [\mathbf{R}_s]_{ii} = \min_{0\le s\le t} \operatorname{trace}(\mathbf{R}_s). \tag{3.10}$$

Note that taking the minimum for all times up to t implies that the stability of the clustering at time t is large only if it is large for all times preceding t. This allows to assign a low stability to partitions where there is a high probability of leaving the community and coming back to it later. According to [24], this minimization is unnecessary in most cases and we have $r_t(\mathcal{P}) \approx \operatorname{trace}(\mathbf{R}_t)$. Nevertheless, taking the minimization ensures maximum generality and allows for example to deal with almost bipartite graphs where trace($\mathbf{R}_s$) can be oscillatory. The definitions above stands all for a given partition $\mathcal{P}$. As we are interested in the optimal clustering in the sense of stability, we search the partition that maximize the stability. Clearly, the optimal partition might be different for each Markov time t. Computing the optimal partition for each Markov time gives the *stability curve of the graph* :

$$r_t = \max_{\mathcal{P}} r_t(\mathcal{P}). \tag{3.11}$$

We understand thus how Markov time acts as an intrinsic resolution parameter: as Markov time grows, the number of communities is expected to decrease, since there are more possibilities for a random walker to escape a community when the time window increases. Hence, communities get bigger (or coarser) with Markov time increasing. Interestingly, one can prove that in the case of *undirected* networks, stability at time 1 is equivalent to the well-known *configuration modularity* measure. But this equivalence does not hold for *directed* networks and therefore does not concern the present work.

At this stage, an important remark has to be made about the assumption of ergodicity. The verification of this assumption is often far from being obvious, especially in the case of big undirected networks. The trick in that case is to introduce "à la Google" random teleportations.[1] Let τ be the *teleportation probability.* Then, if a random walker is located on a node with at least one outlink (which is always the case for the networks that we will consider), it follows one of the outlinks with probability $1 - \tau$. Otherwise, the node is called a *dangling node* and the random

[1] In the original PageRank proposed by S. Brin and L. Page in 1998 (ref. [26]), this consist essentially in applying a perturbation to the transition probability matrix between web pages in order to ensure that at least one row of the matrix is positive, which implies the convergence of the Power Method. If we note the teleportation probability τ, the perturbation can be interpreted as follows: a web surfer follows a link in his current page with probability $1 - \tau$ and jumps to an arbitrary web page with probability τ.

walker is teleported with a uniform probability to another random node. The corresponding perturbation of the transition probability matrix is, in the most general case:

$$\widetilde{\mathbf{M}} = (1-\tau)\mathbf{M} + \frac{1}{N}[(1-\tau)\mathbf{d} + \tau\mathbf{1}]\mathbf{1}^{\intercal}, \tag{3.12}$$

where N is the number of nodes, $\mathbf{d}$ is a binary $N \times 1$ vector whose entries are equal to 1 if the corresponding node is a dangling node and 0 otherwise, and $\mathbf{1}$ is the $N \times 1$ unity vector. In the case that we will consider in the next section, $\mathbf{d}$ is the zero vector. This perturbation is known to make the dynamics ergodic, ensuring the existence and uniqueness of the stationary solution $\boldsymbol{\pi}$.

3.1.2 Extension to continuous time

From a general viewpoint, the discrete process can be interpreted as an approximation of its continuous counterpart : whereas the state of the discrete-time random walker can only change at unit-time intervals, the continuous-time random walkers undergoes a waiting time between each change of state which is itself a random variable. More precisely, it is a continuous memoryless random variable distributed exponentially. Obviously, the transition probabilities from one node to the other are the same for both discrete- and continuous-time processes, only the time at which the jump occurs may vary. The continuous-time process corresponding to (3.1) is governed by the following dynamics :

$$\dot{\mathbf{p}} = \mathbf{p}\,\mathrm{diag}\,\{\boldsymbol{\lambda}(\mathbf{k})\}\,\mathbf{K}^{-1}\mathbf{A} - \mathbf{p}\,\mathrm{diag}\,\{\boldsymbol{\lambda}(\mathbf{k})\} = -\mathbf{p}\mathbf{L}, \tag{3.13}$$

where $\lambda_i(\mathbf{k})$ is the rate at which random walkers leave node i, and $\mathbf{L} = \mathrm{diag}\{\boldsymbol{\lambda}(\mathbf{k})\}[-\mathbf{K}^{-1}\mathbf{A} + \mathbf{I}]$. Two particular cases of this process are implemented by the stability software and are thus examined here, depending on the choice of $\boldsymbol{\lambda}(\mathbf{k})$: the so-called *normalized Laplacian dynamics* and *standard (combinatorial) Laplacian dynamics*. Their names comes from the similarity that arise between $\mathbf{L}$ and the normalized/standard Laplacian matrix. Each of those two dynamics represent best different physical processes. The former correspond to the choice $\boldsymbol{\lambda}_{norm}(\mathbf{k}) = \mathbf{1}$. Hence, the expected waiting time is 1 at every node, and $\mathbf{L} = -\mathbf{K}^{-1}\mathbf{A} + \mathbf{I} = -\mathbf{M} + \mathbf{I}$. The latter corresponds to $\boldsymbol{\lambda}_{combi}(\mathbf{k}) = \mathbf{k}/\langle\mathbf{k}\rangle$. In that case, $\mathbf{L} = (-\mathbf{A} + \mathbf{K})/\langle\mathbf{k}\rangle$ and the average waiting time at node i is $\langle\mathbf{k}\rangle/k_i$. Hence, the expected waiting time at a given node is smaller (resp. larger) than 1 if the total weight of the outgoing edges from that node is larger (resp. smaller) than the average total weight of the outgoing edges on the network. However, the expected waiting time over the whole network is $\langle\langle\mathbf{k}\rangle/\mathbf{k}\rangle = 1$. The corresponding governing equations are respectively

$$\dot{\mathbf{p}} = \mathbf{p}\mathbf{K}^{-1}\mathbf{A} - \mathbf{p} = \mathbf{p}\mathbf{M} - \mathbf{p} \tag{3.14}$$

for the normalized Laplacian and

$$\dot{\mathbf{p}} = \mathbf{p}\frac{\mathbf{A}}{\langle\mathbf{k}\rangle} - \mathbf{p}\frac{\mathbf{K}}{\langle\mathbf{k}\rangle} \tag{3.15}$$

for the combinatorial Laplacian.

The clustered autocovariance matrix for partition $\mathcal{P}$ at time t is easily generalized to

$$\mathbf{R}(t;\mathcal{P}) = \mathbf{H}_{\mathcal{P}}^{\intercal}(\boldsymbol{\Pi}\mathbf{P}(t) - \boldsymbol{\pi}^{\intercal}\boldsymbol{\pi})\mathbf{H}_{\mathcal{P}}, \tag{3.16}$$

where $\mathbf{P}(t)$ is the the transition matrix of the process at time t: $\mathbf{P}(t) = \mathrm{e}^{-t\mathbf{L}}$. The continuous-time definition of the stability of a partition $\mathcal{P}$ follows almost straightforwardly :

$$r(t;\mathcal{P}) = \mathrm{trace}\,[\mathbf{R}(t;\mathcal{P})]\,. \tag{3.17}$$

Notice that it is not necessary to minimize over the time interval $[0,t]$: indeed, it can be shown that $\mathrm{trace}\,[\mathbf{R}(t;\mathcal{P})]$ is monotonically decreasing with time. The interpretation in terms of a random walk is similar to the discrete case : let $P(\mathcal{C}, t)$ be the probability that a random walker is in community $\mathcal{C}$ at time t if it was initially in $\mathcal{C}$, when the system is at stationarity. Discounting the probability of such an event to take place by chance at stationarity and summing over all communities of $\mathcal{P}$ leads to the definition of the stability of the partition $\mathcal{P}$:

$$r(t; \mathcal{P}) = \sum_{\mathcal{C} \in \mathcal{P}} P(\mathcal{C}, t) - P(\mathcal{C}, \infty). \tag{3.18}$$

By ergodicity, the memory of the initial condition is lost at infinity and $P(\mathcal{C}, \infty)$ is thus equal to the probability that two independent walkers are in $\mathcal{C}$ at stationarity. Equation (3.18) tells us that only the communities in which a random walker is likely to stay brings a positive contribution to stability, where *likely to stay* means that the probability for a walker to be in its initial community at time t is larger than the probability of that event occuring by chance at stationarity. The stability curve of the graph can now be expressed as a continuous function of t :

$$r(t) = \max_{\mathcal{P}} r(t; \mathcal{P}). \tag{3.19}$$

3.1.3 Assessing the robustness of a partition

We present here two mechanisms commonly used to assess the relevance of a particular partition. One simple way is to consider that a robust partition should not be alterred by a small modification of the quality function. Such a modification could be for example a perturbation of the Markov time t at which the partition has been found. From this point of view, robust partitions correspond to *plateaux* in the community curve of the graph. In other words, robust partitions should be persistent over a wide interval of Markov time.

The second indicator of the robustness of a partition that we will take into account in this work follows from considering that a robust partition is one that is persistent to small modifications of the optimization algorithm. The central tool to quantify this approach of the robustness of a partition is the *normalized variation of information* [27], which is a popular way to compare two partitions. Let $p(\mathcal{C})$ be the probability for a node to be in community $\mathcal{C}$, i.e. $p(\mathcal{C}) = n_{\mathcal{C}}/N$ where $n_{\mathcal{C}}$ is the number of nodes in community $\mathcal{C}$. The variation of information between partitions $\mathcal{P}_1$ and $\mathcal{P}_2$ is defined by :

$$\mathrm{VI}(\mathcal{P}_1, \mathcal{P}_2) := \frac{H(\mathcal{P}_1, \mathcal{P}_2) - H(\mathcal{P}_1) - H(\mathcal{P}_2)}{\log(N)} = \frac{H(\mathcal{P}_1|\mathcal{P}_2) + H(\mathcal{P}_2|\mathcal{P}_1)}{\log(N)}, \tag{3.20}$$

where $\log(N)$ is a normalization factor; $H(\mathcal{P}) = -\sum_{\mathcal{C}} p(\mathcal{C}) \log[p(\mathcal{C})]$ is the Shannon entropy; $H(\mathcal{P}_1, \mathcal{P}_2)$ is the Shannon entropy of the joint probability $p(\mathcal{C}_1, \mathcal{C}_2)$ that a node belongs to both a community $\mathcal{C}_1$ of $\mathcal{P}_1$ and a community $\mathcal{C}_2$ of $\mathcal{P}_2$. This yields $p(\mathcal{C}_1, \mathcal{C}_2) = n_{\mathcal{C}_1 \cap \mathcal{C}_2}/N$, and $H(\mathcal{P}_1, \mathcal{P}_2) = -\sum_{\mathcal{C}_1 \in \mathcal{P}_1} \sum_{\mathcal{C}_2 \in \mathcal{P}_2} p(\mathcal{C}_1, \mathcal{C}_2) \log[p(\mathcal{C}_1, \mathcal{C}_2)]$; and $H(\mathcal{P}_1|\mathcal{P}_2)$ is the conditional Shannon entropy of partition $\mathcal{P}_1$ given $\mathcal{P}_2$, which is defined in a standard way from the joint distribution: $p(\mathcal{C}_1|\mathcal{C}_2) = p(\mathcal{C}_1, \mathcal{C}_2)/p(\mathcal{C}_2) = n_{\mathcal{C}_1 \cap \mathcal{C}_2}/n_{\mathcal{C}_2}$, and the expression of $H(\mathcal{P}_1|\mathcal{P}_2)$ follows straightforwardly. The latter can be interpreted as the additional information needed to describe $\mathcal{P}_1$ once $\mathcal{P}_2$ is known. This measure of the difference between two partition is then used as follows: for each Markov time, an ensemble of Louvain optimizations of stability are performed, starting from different random initial node ordering. Remember that the problem being $\mathcal{NP}$-hard, we rely on a heuristic algorithm — the Louvain method — that finds a good partition for a given Markov time, but not necessarily the optimal partition. Hence the partition found may differ if a different initial condition is provided. The normalized variation of information allows then to quantify how different the optimized partitions are. Therefore, a low variation of information indicates optimized partitions that are very similar to each others, hence that a small modification of the algorithm barely alter the partition. From the point of view of the field of dynamical system, robust partitions have thus an attractor with a large basin of attraction for the optimization method.

3.2 Clustering of the overturner problem

This section presents the results of applying a stability-based community detection algorithm on the overturner problem. First, we explain how the method is applied and then we present the results for a given set of data's.

3.2.1 Description of the method

In order to apply a clustering algorithm on the overturner problem, we have to define how the model can be considered as a graph. To this end, the domain is decomposed into $n_{box,\,y} \times n_{box,\,z}$ boxes. We note $N_{box} = n_{box,\,y} n_{box,\,z}$ the total number of boxes. Figure 3.1 represents an example of such a domain decomposition with $n_{box,\,y} = 15$ and $n_{box,\,z} = 10$. For any time T, the corresponding directed graph is build as follows : each node represents a box, and the weight of the edge between nodes i and j is the probability $m_{ij}(T)$ that a particle ends up in box j after a time T if it was initially in box i. If $m_{ij}(T) = 0$, one can equivalently consider that there is no edge between nodes i and j. Since the problem is stationary, $m_{ij}(T)$ depends only on the elapsed time T, not on the initial time. Hence, the initial time can indifferently be considered as being zero. The adjacency matrix $\mathbf{M}(T)$ of the graph is build from the weights $m_{ij}(T)$: $[\mathbf{M}(T)]_{ij} = m_{ij}(T)$. For any time T, $\mathbf{M}(T)$ is row-stochastic, i.e. $\mathbf{M}(T)\mathbf{1} = \mathbf{1}$, where $\mathbf{1}$ is the N_{box}-dimensional unit column vector. The latter has a straightforward physical interpretation: every particle remains in the domain.

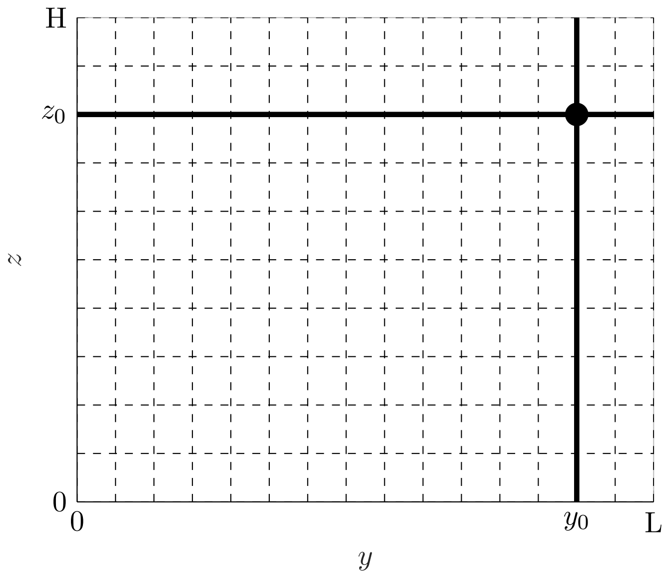


Figure 3.1 – Illustration of the decomposition of the domain into boxes with $n_{box,\,y} = 15$ and $n_{box,\,z} = 10$.

To estimate the probabilities $m_{ij}(T)$, the program is run for a time T with each box containing initially J uniformly distributed particles. $m_{ij}(T)$ is then numerically estimated as the number of particles having started in box i and ending up in box j, divided by J. This *box counting* method has been extensively used to estimate the concentration in studies using random walk modeling, see e.g. [28]. Nevertheless, this method suffers some drawbacks; the most important of them are pointed in [3], but we recall them here for the sake of completeness. The estimated transition probability depends on the choice of the boxes, in particular of their size and their center. Moreover, the number of boxes cannot be chosen to be too large; otherwise the estimated concentration tends to become very irregular or noisy. Finally, the resolution of the estimated concentration is limited to the size of the boxes, as it cannot be described in a box more precisely than a constant. But it is the perfect method for our problem since the volume average over such boxes (the nodes) is precisely what we want. Note however that other methods exist for estimating the concentration, that might be better suited for other studies. For example, the *kernel estimation* method allows to reduce drastically the number of particles, and does not suffer from the resolution limit inherent to the box counting method. This method is briefly presented in [3]. Classical references are [29] and [30].

3.2.2 Use of the stability software

We present here briefly how the *PartitionStability* software is used to compute the partitions. Every concept appearing here has been presented in section 3.1. The `stability` function is simply called as follows :

```
[S,N,VI,C] = stability(M,Markov_T,'directed','plot','teleport',0.01);
```

Here, `M` is the matrix $\mathbf{M}(T)$ at the desired time T; `Markov_T` is the vector containing every Markov times at which the optimal stability partition has to be computed (ideally, the sampling should be exponential); the `'directed'` option specifies that we consider a directed graph; `'plot'` asks the program to plot the stability, number of communities and variation of information as a function of the Markov time; and `'teleport',0.01` allows to specify the value of the teleportation probability τ to 0.01, the default value being 0.15. The choice of 0.01 is motivated by the fact that we believe the graph to be ergodic, even if we cannot prove it. Note that the program allows to choose which type of laplacian should be used to calculate the stability. However, the question does not arise here since both laplacians are equivalent in our case. Indeed, the total outgoing weight is the same at every node and is precisely equal to the number of particles J released in each box. Hence, $k_i = J$ for every node i and $\langle \mathbf{k} \rangle = J$, so that $\boldsymbol{\lambda}_{combi}(\mathbf{k}) = \mathbf{k}/\langle \mathbf{k} \rangle = \mathbf{1} = \boldsymbol{\lambda}_{norm}(\mathbf{k})$. We let thus the program run with the default normalized Laplacian, since it does not make any difference in our case. The output arguments `S`, `N`, `VI` and `C` contain respectively the stability, the number of communities, the variation of information, and the optimal partition for each Markov time contained in `Markov_T`. If the latter is of size n, then `S`, `N` and `VI` are n-dimensional vectors and `C` is a $N_{box} \times n$ matrix. At the jth Markov time, communities are labeled by consecutive integers between 0 and `N(j)`-1 such that `C(i,j)` $= k$ means that node i belongs to community k at Markov time `Markov_T(j)`.

3.2.3 Results

Now we present some results on a particular discretization of the overturner problem. The box decomposition of the domain is the one shown in figure 3.1, and $J = 10\,000$ particles are released in each box. More precisely, a box is decomposed into a 100×100 sub-grid, and one particle is initially located at every point of the sub-grid, so that the particles are initially uniformly distributed within the box. The transition probability matrices $\mathbf{M}(T)$ are generated for different values of T. We show here the results for $T = 1$, 10, 50 and 100 years. The vector of the Markov times `Markov_T` is sampled exponentially from 0.1 to 100 : $\log_{10}$(`Markov_T`) $= [-1, -0.98, \ldots, 1.98, 2]$. Notice that the physical meaning of the Markov time changes with T : a Markov time step of 1 is equal to a physical time step of T. Hence, for $a > 0$, if for $\mathbf{M}(T)$

we find some communities in the range of Markov times $[t_{M_1}, t_{M_2}]$, then for $\mathbf{M}(aT)$ we expect to find similar communities in the range of Markov times $\frac{1}{a}[t_{M_1}, t_{M_2}]$.

Figures 3.2, 3.4, 3.6 and 3.8 show the stability curves, the number of communities and the variation of information as functions of the Markov time for $T = 1$, 10, 50 and 100 years respectively. As discussed in section 3.1.3, robust partitions correspond to plateaux in the community curve of the graphs. By using this criterion, partitions of 6, 5, 4, 3 and 2 communities are found at different time scales. Those partitions are summarized in table 3.1 along with the physical time range at which they reveal themselves. Figure 3.3, 3.5, 3.7 and 3.9 shows the most robust clusterings for $T = 1$, 10, 50 and 100 respectively. From table 3.1, we observe that some similar partitions happen to be the most relevant at different time scales when we modify T. For example, for $T = 1$, 6 communities are found in the time range 9 - 12 years (figure 3.3a). A similar clustering is found for $T = 10$ and $T = 50$ but in the time ranges 24 - 48 years and 36-48 years respectively (figures 3.5a and 3.7a).

It is important to notice that the community detection algorithm may fail to detect the right number of communities. Take figure 3.5d for example : the stability software detects 2 communities. However, the white community consists of two noncontiguous blocks. Intuitively, particles leaving the lower white block should enter the khaki block first before entering the upper white block. Hence, there should be 3 communities rather than 2 for this partitioning. A way to quantify this intuition is by looking at

$$\mathbf{M}_{\mathcal{P}}(T) = \mathrm{diag}^{-1}(\mathbf{n})\mathbf{H}_{\mathcal{P}}^{\intercal}\mathbf{M}(T)\mathbf{H}_{\mathcal{P}}, \tag{3.21}$$

where $\mathbf{n}$ is the c-dimensional vector containing the number of blocks in each community. $[\mathbf{M}_{\mathcal{P}}]_{kl}$ is the transition probability from community k to community l. By considering the clustering where the lower and the upper white blocks are separated communities, we get

$$\mathbf{M}_{\mathcal{P}}(10) = \begin{pmatrix} 0.886 & 0.114 & 0.000 \\ 0.052 & 0.895 & 0.053 \\ 0.017 & 0.256 & 0.727 \end{pmatrix}. \tag{3.22}$$

Here, community 1 is the lower white block, community 2 is the khaki block and community 3 is the upper white block. We observe that $[\mathbf{M}_{\mathcal{P}}]_{13} = 0$ and $[\mathbf{M}_{\mathcal{P}}]_{31} = 0.017$, indicating very weak links between the lower and the upper white blocks. Hence, they should indeed be considered as separated communities. However, this does not mean that 3.5d provides then the optimal clustering with 3 communities ! Such a clustering is rather given by figure 3.5c, and the clustering proposed in figure 3.5d should simply be disregarded as being non-relevant.

Now, let us analyze a seemingly relevant community structure. By looking at table 3.1 together with figures 3.5b and 3.7b, we observe that two similar 5-communities clusterings arise in the time range 50 - 63 years when $T = 10$ and $T = 50$. This indicates that those community structures might be more resilient than others. There is only a 2 boxes difference between the two clusterings; we will therefore focus on the clustering found for $T = 10$, namely the one from figure 3.5b. The communities are numbered from 1 to 5 on the figure. The matrix $\mathbf{M}_{\mathcal{P}}$ for this community structure is

$$\mathbf{M}_{\mathcal{P}}(10) = \begin{pmatrix} 0.907 & 0.024 & 0 & 0.024 & 0.045 \\ 0.073 & 0.827 & 0.043 & 0.057 & 0 \\ 0 & 0.029 & 0.925 & 0.036 & 0.010 \\ 0.039 & 0.041 & 0.089 & 0.776 & 0.055 \\ 0.020 & 0 & 0.048 & 0.022 & 0.910 \end{pmatrix}. \tag{3.23}$$

Obviously, particles in a community tends to stay in that community. But what are the main interconnections between communities ? By looking at matrix $\mathbf{M}_{\mathcal{P}}$, we observe that particles leaving community 1 goes preferentially to community 5; from community 2, the main tendency is to go to community 1; from 3 to 4 and 2; from 4 to 3 (mainly because of the size of 3) and from 5 to 3. Hence, the dominant tendency is that the particles tend to describes a clockwise cycle in the domain, which is exactly the expected behavior.

J'aimerais aller un peu plus loin dans mes commentaires mais les idées ne se bousculent pas... Et les commentaires que je fais ci-dessus ne nous apprennent rien. Cela fait sans doute beaucoup d'images d'images pour au final pas grand chose.

Table 3.1 – Summary of the dominant clusterings found by inspection of the transition probability matrix $\mathbf{M}(T)$ for $T = 1$, 10, 50 and 100 years.

T	Time range [year]				
	6 communities	5 communities	4 communities	3 communities	2 communities
1	9 - 12	15 - 26	28 - 36	38 - ...	
10	24 - 48	50 - 63		91 - 316	331 - ...
50	36 - 48	50 - 66		69 - 138	144 - 190
100	58 - 76	79 - 105		120 - 229	240 - 316

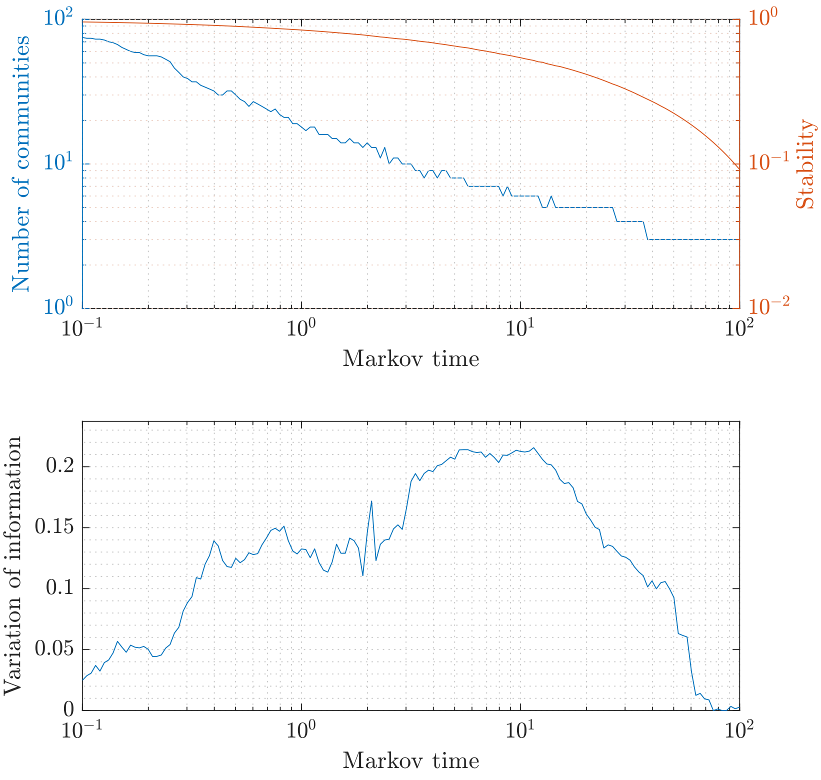
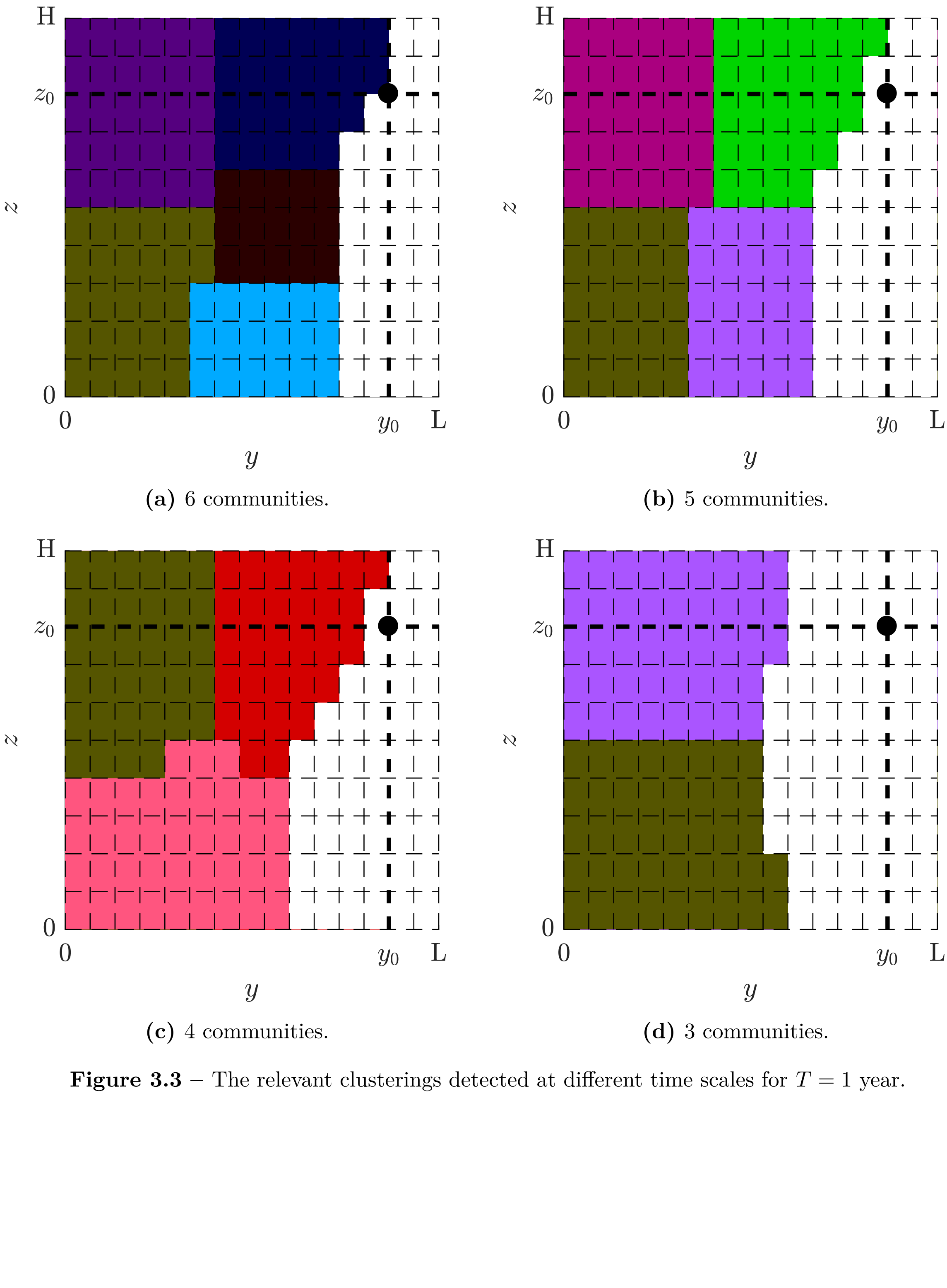


Figure 3.2 – Stability, number of communities and variation of information as a function of the Markov time for $T = 1$ year.



(a) 6 communities.

(b) 5 communities.

(c) 4 communities.

(d) 3 communities.

Figure 3.3 – The relevant clusterings detected at different time scales for $T = 1$ year.

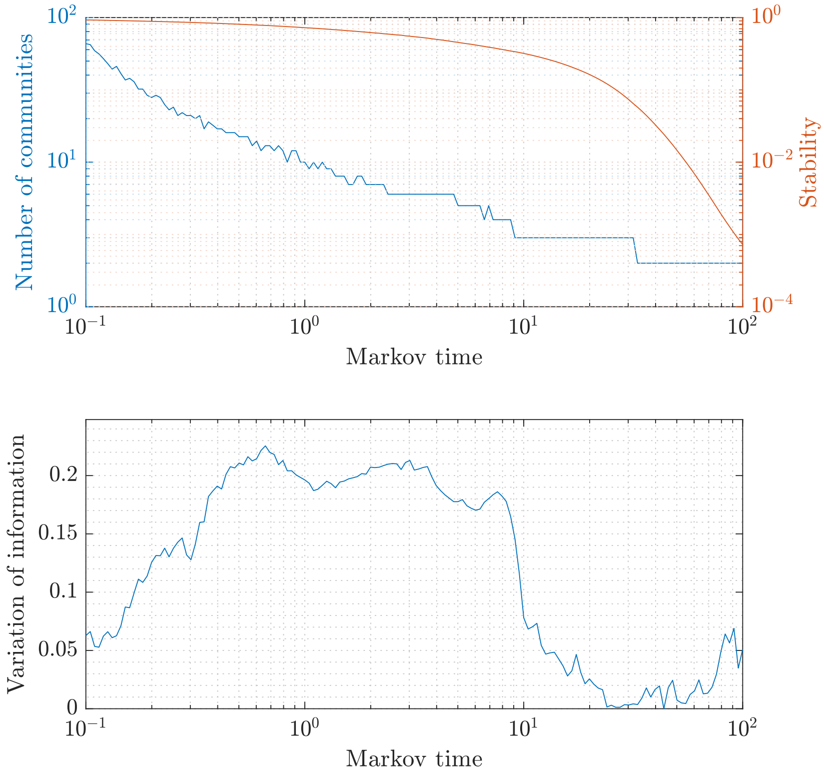
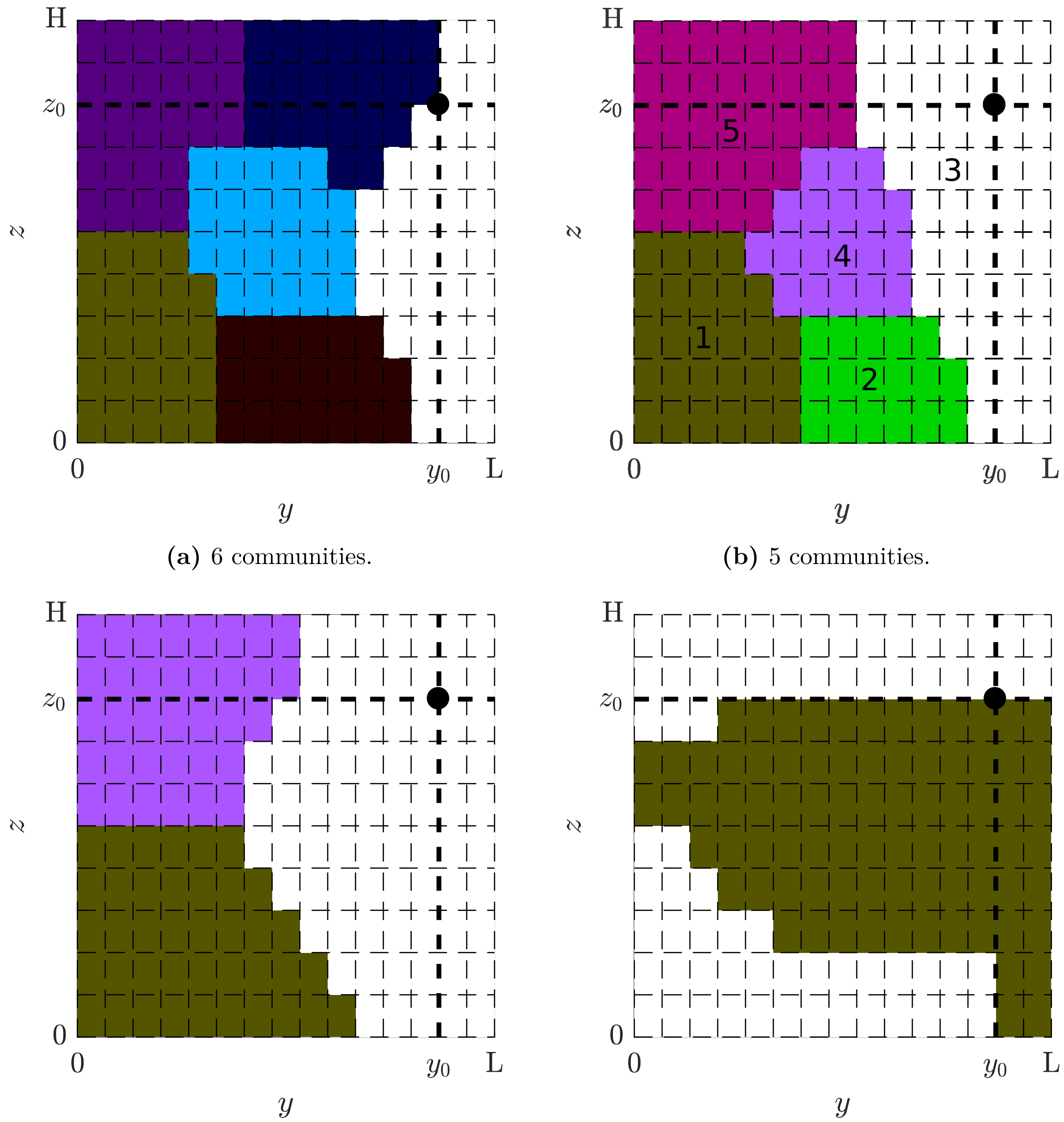


Figure 3.4 – Stability, number of communities and variation of information as a function of the Markov time for $T = 10$ years.



(a) 6 communities.

(b) 5 communities.

(c) 3 communities.

(d) 2 communities detected by the algorithm which should rather be considered as being 3 communities.

Figure 3.5 – The relevant clusterings detected at different time scales for $T = 10$ years.

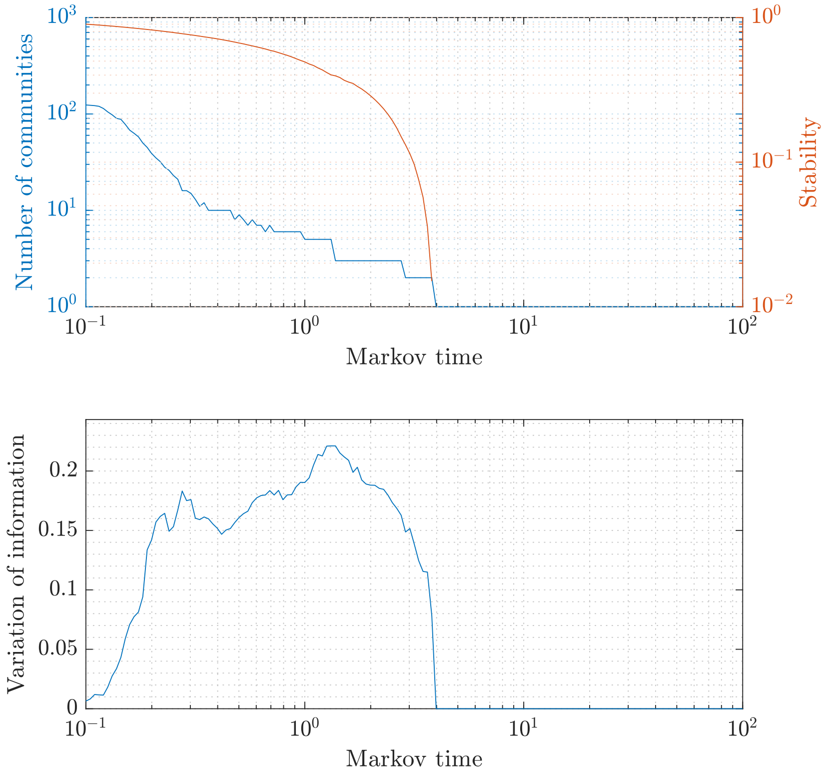
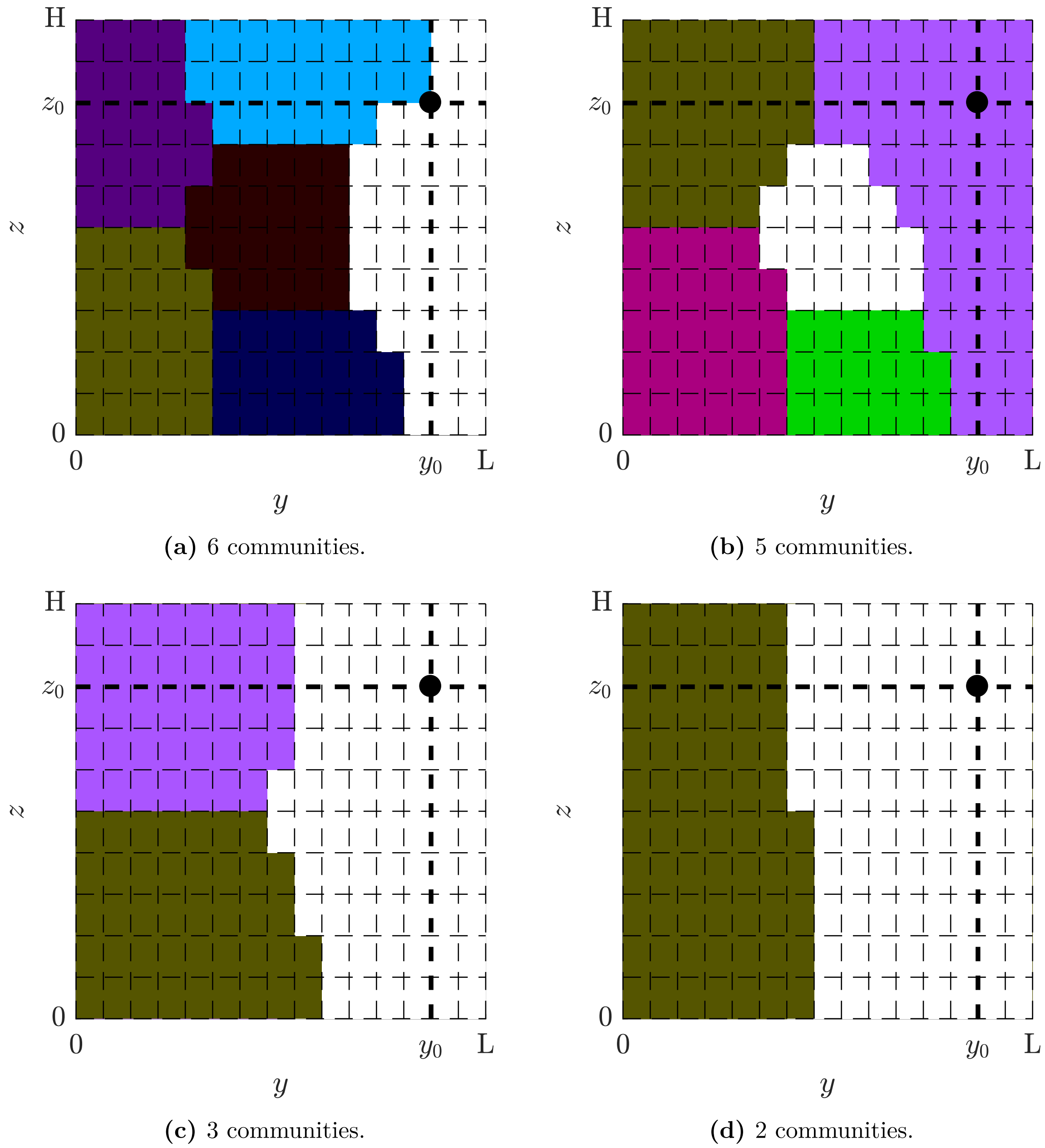


Figure 3.6 – Stability, number of communities and variation of information as a function of the Markov time for $T = 50$ years.



(a) 6 communities.

(b) 5 communities.

(c) 3 communities.

(d) 2 communities.

Figure 3.7 – The relevant clusterings detected at different time scales for $T = 50$ years.

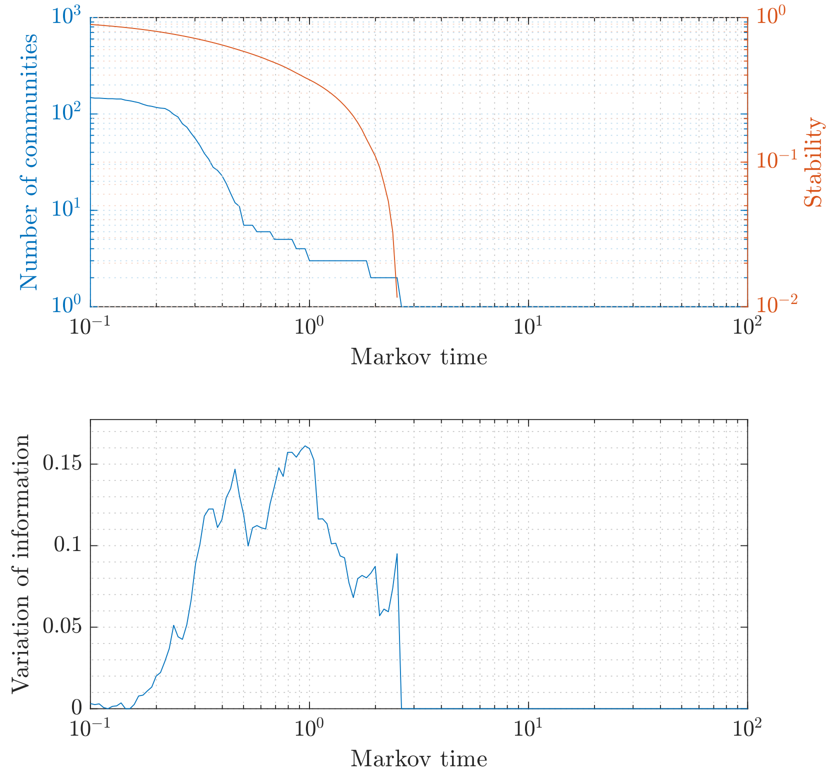
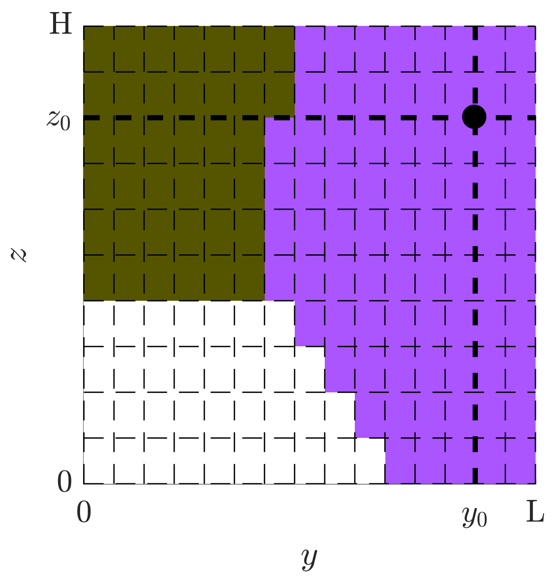
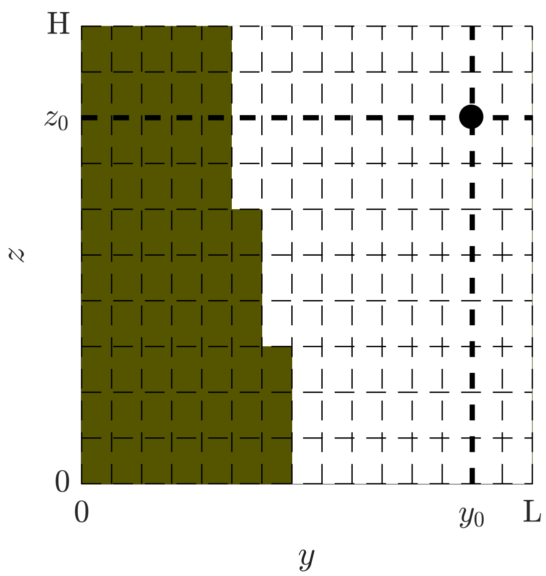


Figure 3.8 – Stability, number of communities and variation of information as a function of the Markov time for $T = 100$ years.



(a) 3 communities.



(b) 2 communities.

Figure 3.9 – The relevant clusterings detected at different time scales for $T = 100$ years.

Bibliography

[1] Eric Deleersnijder. Éléments d'un modèle latitude-profondeur très simple — application à l'injection de "co2 idéalisée" dans l'océan. 2006. Available at `http://hdl.handle.net/2078.1/155304`.

[2] Eric Deleersnijder, Jean-Michel Campin, and Eric JM Delhez. The concept of age in marine modelling: I. theory and preliminary model results. *Journal of Marine Systems*, 28(3):229–267, 2001.

[3] Darya Spivakovskaya, Arnold W. Heemink, and Eric Deleersnijder. Lagrangian modelling of multi-dimensional advection-diffusion with space-varying diffusivities: theory and idealized test cases. *Ocean Dynamics*, 57(3):189–203, 2007.

[4] D Spivakovskaya, AW Heemink, and Eric Deleersnijder. The backward îto method for the lagrangian simulation of transport processes with large space variations of the diffusivity. *Ocean Science Discussions*, 4(4):623–652, 2007.

[5] VAN STIJN. Positive advection schemes for environmental studies. *Numerical methods in laminar and turbulent flow*, 1987.

[6] Y Yang, LT Wilson, ME Makela, and MA Marchetti. Accuracy of numerical methods for solving the advection–diffusion equation as applied to spore and insect dispersal. *Ecological modelling*, 109(1):1–24, 1998.

[7] Chunmiao Zheng, Gordon D Bennett, et al. *Applied contaminant transport modeling*, volume 2. Wiley-Interscience New York, 2002.

[8] JR Hunter. The application of lagrangian particle-tracking techniques to modelling of dispersion in the sea. *North-Holland Mathematics Studies*, 145:257–269, 1987.

[9] JR Hunter, PD Craig, and HE Phillips. On the use of random walk models with spatially variable diffusivity. *Journal of Computational Physics*, 106(2):366–376, 1993.

[10] GT Yeh. A lagrangian-eulerian method with zoomable hidden fine-mesh approach to solving advection-dispersion equations. *Water Resources Research*, 26(6):1133–1144, 1990.

[11] Eric M LaBolle, Graham E Fogg, and Andrew FB Tompson. Random-walk simulation of transport in heterogeneous porous media: Local mass-conservation problem and implementation methods. *Water Resources Research*, 32(3):583–593, 1996.

[12] C. W. Gardiner. Stochastic methods. *Springer-Verlag*, 1985.

[13] Arnold Heemink and et al. Modeling transport processes using stochastic differential equations. Slides, 2011.

[14] Roland Keunings. Mathematical modeling of physical problems. Lecture notes, 2016.

[15] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Springer-Verlag, Berlin Heidelberg New York, 1995.

[16] Eric M LaBolle, Jeremy Quastel, Graham E Fogg, and Janko Gravner. Diffusion processes in composite porous media and their numerical integration by random walks: Generalized stochastic differential equations with discontinuous coefficients. *Water Resources Research*, 36(3):651–662, 2000.

[17] Pere Colet and Raúl Toral. *Stochastic numerical methods: an introduction for students and scientists*. John Wiley & Sons, 2014.

[18] Thomas A Prickett, Carl G Lonnquist, Thomas G Naymik, et al. A" random-walk" solute transport model for selected groundwater quality evaluations. *Bulletin/Illinois State Water Survey; no. 65*, 1981.

[19] Andrew FB Tompson, Efstratios G Vomvoris, and Lynn W Gelhar. Numerical simulation of solute transport in randomly heterogeneous porous media: motivation, model development, and application. Technical report, Lawrence Livermore National Lab., CA (USA), 1987.

[20] Eric Deleersnijder. Test cases for advection-diffusion equations with a first-order decay term. 2011. Available at `http://hdl.handle.net/2078.1/155372`.

[21] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3):75–174, 2010.

[22] J-C Delvenne, Sophia N. Yaliraki, and Mauricio Barahona. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, 107(29):12755–12760, 2010.

[23] Renaud Lambiotte, J-C Delvenne, and Mauricio Barahona. Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770*, 2009.

[24] Jean-Charles Delvenne, Michael T. Schaub, Sophia N. Yaliraki, and Mauricio Barahona. The stability of a graph partition: A dynamics-based framework for community detection. In *Dynamics On and Of Complex Networks, Volume 2*, pages 221–242. Springer, 2013.

[25] Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[26] Sergey Grin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[27] Marina Meilă. Comparing clusterings – an information based distance. *Journal of multivariate analysis*, 98(5):873–895, 2007.

[28] AM Riddle. The specification of mixing in random walk models for dispersion in the sea. *Continental Shelf Research*, 18(2):441–456, 1998.

[29] Bernard W. Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[30] MP Wand and MC Jones. Kernel smoothing. 1995. *Chapman&Hall, London*, 1995.

[31] Catherine Timmermans. Ventilation et rétention dans un bassin océanique idéalisé. Master's thesis, UCL, 2006.

[32] Renaud Lambiotte, Roberta Sinatra, J-C Delvenne, Tim S. Evans, Mauricio Barahona, and Vito Latora. Flow graphs: Interweaving dynamics and structure. *Physical Review E*, 84(1):017102, 2011.

Appendix A

Numerical considerations

A.1 Stieltjes integral

The Riemann-Stieltjes integral is a generalization of the Riemann integral. Let f and g be real-valued functions defined on a closed interval $[a,\, b]$. The Riemann-Stieltjes integral of f with respect to g is denoted

$$\int_a^b f(t)\mathrm{d}g(t). \tag{A.1}$$

Consider a partition of the interval

$$a = t_0 < t_1 < \cdots < t_{n-1} < t_n = b, \tag{A.2}$$

and define

$$h_n \triangleq \max_{i\in\{1,2,\ldots,n\}} (t_i - t_{i-1}). \tag{A.3}$$

Now take the Riemann sum

$$\sum_{i=1}^{n} f(\tau_i)[g(t_i) - g(t_{i-1})], \tag{A.4}$$

with $\tau_i \in [t_{i-1},\, t_i]$. If the sum tends to a fixed number I as $n \to \infty$ and $h_n \to 0$, then

$$\int_a^b f(t)\mathrm{d}g(t) = I. \tag{A.5}$$

A.2 Mean-square limit

Suppose that we have a probability space Ω, and a sequence of random variables X_n defined on Ω. We say that X_n converges to X in the mean-square sense if

$$\lim_{n\to\infty} \left\langle (X_n - X)^2 \right\rangle = 0, \tag{A.6}$$

and we note

$$\operatorname*{ms-lim}_{n\to\infty} X_n = X. \tag{A.7}$$

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve **www.uclouvain.be/epl**