Fish Fitness problem

Ayoub Sbai

June 30, 2024

1 Introduction

The fish is peacefully flowing in the river, described by the Stokes equation. However, there is a Gaussian source of pollution spreading in the corner. The fish has now to optimize its location by searching coordinates in the negative gradient direction of the pollution scalar field (fitness landscape) in order to survive. Will the fish survive ? Let's find out!

2 Mathematical Modeling

2.1 General Outline

These equations describe the motion of a fluid (governed by the Stokes equations) and the transport of a pollutant within that fluid (governed by the Burgers equation).

1. **Stokes Equation**:

$$-\nabla^2 u + \nabla p = 0$$

This equation describes the conservation of momentum for an incompressible fluid. Here, u represents the velocity field of the fluid, and p represents the pressure field. The term $-\nabla^2 u$ represents the viscous forces acting on the fluid, and ∇p represents the pressure gradient. In simple terms, this equation states that the change in momentum of the fluid is balanced by the pressure gradient and the viscous forces.

2. **Continuity Equation**:

$$\nabla \cdot u = 0$$

This equation represents the conservation of mass for an incompressible fluid. It states that the divergence of the velocity field u is zero, meaning that the fluid is incompressible and the volume flow rate into any region equals the volume flow rate out of that region.

with boundary conditions: $u_1 = v_l(y - Y_1)(Y_2 - y)$, $u_2 = 0$, $v_l = v_{max} \frac{4}{(Y_2 - Y1)^2}$.
$X_1 = 1.5$, $X_2 = 4$, $Y_1 = 1$, $Y_2 = 2$

3. **Advection-Diffusion Equation**:

$$\frac{\partial \phi}{\partial t} + u \cdot \nabla \phi = \nu \nabla^2 \phi$$

This equation is technically not the Burgers equation because the transport drift velocity is not the unknown solution ϕ, and therefore it presents no discontinuities or non-linearities. However, because it takes a very similar form, we are going to call it the Burgers equation due to absence of ambiguity. This equation describes the transport of a scalar quantity ϕ (in this case, pollution concentration) within the fluid flow. Here, ν represents the diffusion coefficient of the pollutant. The term $u \cdot \nabla \phi$ represents the advection term, which accounts for the transport of the pollutant by the fluid velocity field. The term $\nu \nabla^2 \phi$ represents the diffusion term, which accounts for the spreading of the pollutant due to diffusion processes (or kinematic viscosity).

In the context of a fish trying to relocate to maximize its chances for survival, these equations could be used to model how the fluid flow patterns affect the distribution of pollutants in the water, which in turn could affect the fish's habitat and survival. The fish might navigate through the fluid flow to avoid regions with high pollutant concentrations, as to maximize its fitness landscape through finding space with the minimum polluant concentration.

2.2 Analytic Resolution: Harmonic Analysis

The viscous Burgers' equation is a nonlinear partial differential equation (PDE) that describes the behavior of a viscous fluid in one dimension. It is given by:

$$\frac{\partial \phi}{\partial t} + u \cdot \nabla \phi = \nu \nabla^2 \phi$$

where $u(x,t)$ is the velocity of the fluid at position x and time t, and ν is the viscosity coefficient.

In order to solve the Burgers equation analytically, we have to add two important assumptions as to regularize the problem:

H1: the transport drift velocity u is homogeneous ie; it doesn't vary with space coordinates.

H2: x and y are independent: this assumption is not necessary but computationally convenient as it allows us to solve the Burgers equation in decoupled dynamics in x and y separately, and this will prove handy when taking the Fourier transform of two-variables function (x and t, y and t) rather than 3-variables (x,y,t), or numerically, when using 2-dimensional FFT instead of 3-dimensional FFT.

In the light of these two hypotheses, we substitute the above Burgers equation by the following simplified version of two decoupled, homogeneous transport equations:

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}$$

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial y} = \nu \frac{\partial^2 u}{\partial y^2}$$

We can use the Fourier transform to solve this equation. Let's denote the Fourier transform of $u(x,t)$ with respect to x as $\hat{u}(k,t)$, where k is the wave number. The Fourier transform pair is given by:

$$\hat{u}(\omega,t) = \int_{-\infty}^{\infty} u(x,t) e^{-i} \, dx$$

and its inverse

$$u(x,t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{u}(\omega,t) e^{i} \, d\omega$$

Taking the Fourier transform of the Burgers' equation with respect to x (and y), we get:

$$\frac{\partial \hat{u}}{\partial t} + i\omega \hat{u} = -\nu \omega^2 \hat{u}$$

This is now an ordinary differential equation (ODE) in the variable ω where $\hat{u} = u(\hat{\omega}, t)$. Solving this ODE, we can then apply the inverse Fourier transform to obtain $u(x,t)$.
Solving the ODE using the Ansatz method or Fourier method (via integration), we get:

$$\hat{u}(\omega,t) = \hat{u}(\omega,0) e^{-(iv\omega + \nu\omega^2)t} = \sqrt{\pi} e^{-(i\omega x_0 + iv\omega + \pi^2 \omega 2 + \nu \omega^2)t}$$

By Fourier inversion theorem (existence and unicity between Fourier transform and its inverse), we have:

$u(x,t) = \mathcal{F}^{-1}(\hat{u}(\omega,t) = \mathcal{F}^{-1}(\sqrt{\pi}e^{-(i\omega x_0+iv\omega+\pi^2\omega 2+\nu\omega^2)t})$

By the Convolution theorem, we have:

$\mathcal{F}^{-1}(\hat{u}(\omega,t)) = u(x,0) * \mathcal{F}^{-1}(e^{-(iv\omega+\nu\omega^2)t}) = u(x,0) * \mathcal{F}^{-1}(e^{-\nu\omega^2 t}) * \mathcal{F}^{-1}(e^{-iv\omega t})$

Thus, we get our analytic form:

$\forall x \in [\,-4,4]\ \forall t \geq 0\ \ u(x,t) = \frac{u(x,0)*e^{-\frac{x^2}{4\nu t}}}{\sqrt{4\pi\nu t}} * \delta(x - vt)$

such that $u(x,0) = e^{-10(x-x_s)^2}$

and by symmetric permutation for variable y, we get:

$\forall y \in [\,0,2]\ \forall t \geq 0\ \ u(y,t) = \frac{u(y,0)*e^{-\frac{y^2}{4\nu t}}}{\sqrt{4\pi\nu t}} * \delta(y - vt)$

such that $u(y,0) = e^{-10(y-y_s)^2}$

such that the convolution of f and g is: $(f*g)(t) := \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$

Numerically, we are going to proceed with DFT or FFT algorithms

2.3 Numerical Resolution: Weak Formulation and Forward Euler Schema (FreeFem++)

The pollution concentration dynamics or evolution is described by the deterministic Burgers equation with homogeneous parameters:

$$\frac{\partial \phi}{\partial t} + u \cdot \nabla \phi = \nu \nabla^2 \phi$$

with initial condition:

$\phi(\text{x,y,t=0}) = e^{-10((x-x_s)^2+(y-y_s)^2)}$

Where:

- ϕ is the concentration of pollution.
- u is the velocity vector field.
- t is time.
- ∇ is the gradient operator.
- ∇^2 is the Laplacian operator.
- ν is the kinematic viscosity, representing the diffusion coefficient. (=0.001)

To derive the weak formulation of the Burgers equation, we first need to introduce a test function and integrate the equation over a suitable domain. Let's denote the test function by v, an infinitely differentiable function that cancels out in the boundaries $\gamma(v) = 0$). We also need to define certain boundary conditions for our weak solution ϕ, here we chose the Robin boundary conditions (Dirichlet and Von Neumann boundary conditions) to simplify the calculations:

- $\phi = 0\ \forall x \in \partial\Omega$

- $\nabla\phi \cdot \mathbf{n} = \frac{\partial \phi}{\partial n} = 0 \; \forall x \in \partial\Omega$

Therefore our solution space is the Sobolev space of order 1:

$H_0^1(\Omega) = \{f \in H^1(\Omega), \gamma(f) = 0, \gamma(\nabla f \cdot \mathbf{n}) = 0\}$

The Burgers equation is:

$$\frac{\partial \phi}{\partial t} + u \cdot \nabla\phi = \nu\nabla^2\phi$$

Now, multiply both sides of the equation by the test function v and integrate over the domain Ω:

$$\int_\Omega \frac{\partial \phi}{\partial t} v \, dx + \int_\Omega (u \cdot \nabla\phi) v \, dx = \nu \int_\Omega \nabla^2 \phi \, v \, dx$$

We integrate by parts the second term on the left-hand side and the third term on the right-hand side:

$$\int_\Omega \frac{\partial \phi}{\partial t} v \, dx + \int_\Omega u \cdot \nabla\phi \, v \, dx = -\nu \int_\Omega \nabla\phi \cdot \nabla v \, dx + \nu \int_{\partial\Omega} \frac{\partial \phi}{\partial n} v \, ds$$

where $\partial\Omega$ denotes the boundary of Ω, and $\frac{\partial \phi}{\partial n}$ denotes the normal derivative of ϕ on $\partial\Omega$.

Now, let's define the weak formulation:

Find $\phi \in V$ such that:

$$\int_\Omega \frac{\partial \phi}{\partial t} v \, dx + \int_\Omega u \cdot \nabla\phi \, v \, dx = -\nu \int_\Omega \nabla\phi \cdot \nabla v \, dx + \nu \int_{\partial\Omega} \frac{\partial \phi}{\partial n} v \, ds$$

for all $v \in \hat{V}$, where V and $\hat{V}$ are suitable function spaces satisfying certain boundary conditions.

This weak formulation can be used for numerical discretization methods such as finite element methods, finite difference methods, etc., where it's often easier to work with than the original strong form of the equation.

To apply an explicit time discretization scheme to the weak formulation derived earlier, let's discretize the time derivative using a forward Euler scheme. Denoting the time step as Δt and the time index as n, the time-discretized weak formulation becomes:

$$\int_\Omega \frac{\phi_{n+1} - \phi_n}{\Delta t} v \, dx + \int_\Omega u \cdot \nabla\phi_n \, v \, dx = -\nu \int_\Omega \nabla\phi_n \cdot \nabla v \, dx$$

which can be simplified further as follow:

$$\int_\Omega \phi_{n+1} v \, dx = \int_\Omega \phi_n v \, dx - \Delta t \int_\Omega (u \cdot \; v + \nu \cdot \nabla v \;) \nabla\phi_n \, dx$$

$\Delta t = t_{n+1} - tn \; \forall n \in |0, N-1|$, therefore $\Delta t = \frac{t_{N-1} - t_0}{N}$

where ϕ_n is the approximation of ϕ at time $t_n = t_0 + n\Delta t = n\Delta t$, and ϕ_{n+1} is the approximation of ϕ at time $t_{n+1} = (n+1)\Delta t$.

This discretization scheme is explicit (Forward Euler scheme) because the value of ϕ_{n+1} can be directly computed from the known values at time $t = n\Delta t$.

Now, this discrete equation can be solved iteratively in time starting from the initial condition ϕ_0. At each time step n, the equation can be solved for ϕ_{n+1}.

This explicit time discretization is relatively simple to implement but may suffer from stability issues depending on the choice of time step size and spatial discretization. Careful consideration of stability constraints, such as the CFL (Courant-Friedrichs-Lewy) condition, may be necessary to ensure numerical stability (the same condition used to stabilize molecular dynamics simulations through the constraint $dt \leq T_{min} = \frac{1}{f_{max}}$). But we are going to proceed with FreeFem++ implementation which is optimal in terms of speed.

2.4 Numerical Results

2.4.1 FreeFem++ Simulation

FreeFem++ is a powerful open-source finite element software package designed for solving partial differential equations (PDEs) numerically. It provides a user-friendly environment for defining complex geometries, specifying boundary conditions, and discretizing PDEs into finite element formulations. FreeFem++ supports a wide range of numerical methods, including finite element methods, spectral methods, and meshless methods, making it suitable for solving various types of PDEs encountered in physics, engineering, and other scientific disciplines. Its flexibility and ease of use enable researchers and engineers to efficiently implement and solve PDE problems, visualize results, and analyze the behavior of physical systems, making it a valuable tool for both academic and industrial applications.

Additionally, FreeFem++ allows for the formulation of weak forms of PDEs, facilitating the implementation of variational principles and enabling the efficient solution of problems involving complex boundary conditions or heterogeneous materials.

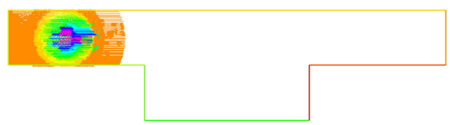

Figure 1: Initial Conditions

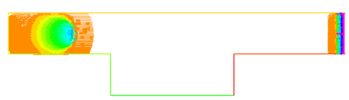

Figure 2: Advection-Diffusion Dynamics

Figure 3: Advection-Diffusion Dynamics

2.4.2 FFT Simulation

Inverse Fast Fourier Transform (FFT) Simulation is a numerical technique utilized to solve partial differential equations (PDEs) by transforming them into the frequency domain using the FFT, allowing for efficient computation and analysis. This method discretizes the spatial and temporal domains of the PDE, representing the solution as a sum of sinusoidal functions. By leveraging the FFT to convert the problem from the spatial domain to the frequency domain, Inverse FFT Simulation facilitates the approximation of complex systems with high accuracy and computational efficiency. It finds widespread applications in diverse fields such as fluid dynamics, heat transfer, signal processing, and

Figure 4: Advection-Diffusion Dynamics

Figure 5: Advection-Diffusion Dynamics

quantum mechanics, enabling researchers and engineers to gain insights into the dynamic behavior of physical systems and processes.

2.5 Stability analysis:

2.5.1 Krein-Rutman theorem

For recall, the Burgers equation of our problem is as follow:

$$\frac{\partial \phi}{\partial t} + u \cdot \nabla \phi = \nu \nabla^2 \phi$$

where u satisfies the Stokes equation:

$$-\nabla^2 u + \nabla p = 0$$

and Continuity equation:

$$\nabla \cdot u = 0$$

Given that $u(x,t) \neq \phi(x,t)$ uniformly on Ω, then the Burgers equation at hand is linear (ie; its solution space is a vector space). Therefore, it can be represented as a linear system of a given linear operator $\mathcal{L}_{\lfloor}$.

such that $\mathcal{L}_{\lfloor} = -u \cdot \nabla - \nu \nabla^2$.

The Burgers equation thus becomes:

$$\frac{\partial \phi}{\partial t} = \mathcal{L}_{\lfloor}(\phi) = -u \cdot \nabla \phi - \nu \nabla^2 \phi$$

Thanks to the system linearity, it makes it convenient to study the stability of equilibrium point of our dynamical system. More specifically, we are interested in the stability analysis of the equilibrium point 0, obviously because we want to assess the conditions on our spectral decomposition (ie; eigenvalue problem of decoupled dynamics) under which the 0 equilibrium point is stable. This can be done thanks to the Krein-Rutman theorem.

To apply the Krein-Rutman theorem, we have to prove that u is Lipshitzian in $\bar{\Omega}$:

u is the fluid velocity satisfying the Stokes equation,

$$-\nabla^2 u + \nabla p = 0$$

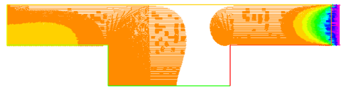

Figure 6: Advection-Diffusion Dynamics

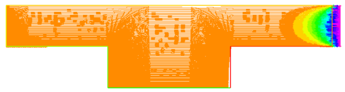

Figure 7: Advection-Diffusion Dynamics

As for physical intuition, u satisfies the Stokes equation ie; Navier-Stokes equation without the advection term of the material derivative that is responsible for discontinuities (notably in the Burgers equation). Therefore, a fluid described by Stokes equation cannot exhibit a brutal change in its velocity ie; it cannot exhibit an infinitely high rate of change (blow up), therefore it must have a bounded rate of change (unlike in the Burgers equation or Navier-Stokes equation). Thus, u is Lipshitzian.

Therefore, $\exists!(\lambda_1, \phi) \in R \times C^2(\hat{\Omega})$ such that:

$$\mathcal{L}_{\lfloor}(\phi) = -u \cdot \nabla\phi - \nu\nabla^2\phi = \lambda_1\phi$$

where:

- $\phi = 0 \ \forall x \in \partial\Omega$
- $\nabla\phi \cdot \mathbf{n} = \frac{\partial\phi}{\partial n} = 0 \ \forall x \in \partial\Omega$
- $\phi > 0 \ \forall x \in \Omega$
- $\max_\Omega \phi = 1$

The operator $\mathcal{L}_\nabla$ has a countably infinite set of real eigenvalues, all real, $\lambda_1 < \lambda_2 \leq \ldots \leq \lambda_k \leq \ldots$, with $\lambda_k \to +\infty$ as $k \to +\infty$. The associated eigenfunctions $(\phi_k)_{k\geq 1}$ form an orthogonal basis of $L^2(\Omega)$. Considering the linearized problem:

We decompose the solution in the basis of eigenfunctions: $\phi(t, x) = \sum_{k=1}^{\infty} a_k(t)\phi_k(x)$.

$a'_k\phi_k = a_k(D\Delta\phi_k + r\phi_k) = -a_k\lambda_k\phi_k$

Thus, $a_k(t) = a_k(0)e^{-\lambda_k t}$ and

$$\phi(t, x) = a_1(0)\phi_1(x)e^{-\lambda_1 t} + a_2(0)\phi_2(x)e^{-\lambda_2 t} + \ldots$$

If $a_1(0) > 0$, $\lambda_1 > 0 \to$ exponential decay of the polluant

and $\lambda_1 < 0 \to$ exponential growth of polluant.

3 Interpretation

The logistic growth equation for the fish population can be expressed as:

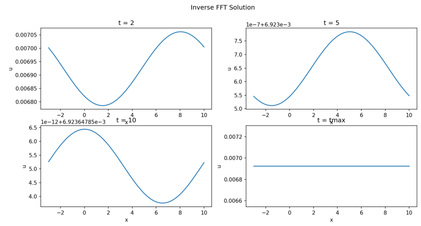

Figure 8: Inverse FFT Solution

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K(x,y)}\right)$$

Where:

- N is the population of fish.
- r is the intrinsic growth rate of the fish population.
- $K(x,y)$ is the carrying capacity of fish species at neighborhood of $(x,y) : \mathcal{N}_\epsilon(x,y) = \{(a,b) \in \Omega, d((x,y),(a,b)) < \epsilon\}$.

The threshold function $\psi_f(x,y)$ comparing fish-specific function $I_f(x,y)$ to pollution level $\phi(x,y)$ can be defined as follows:

$$\psi_f(x,y) = \begin{cases} 0 & \text{if } I_f < \phi(x,y) \\ 1 & \text{otherwise} \end{cases}$$

- I_f: intrinsic immunity level
- $\psi_f(x,y)$: threshold function (1 = survival, 0 = extinction)

For fish to optimize their location by moving towards the negative gradient direction of the pollution scalar field, they would move according to the normalized negative gradient of the pollution function $-\frac{\nabla\phi(x,y)}{\|\nabla\phi(x,y)\|}$. This ensures that fish move in the direction of steepest descent of pollution concentration.

So, the outline for the fish movement direction becomes:

$$e_{d_0} = -\frac{\nabla\phi(x,y)}{\|\nabla\phi(x,y)\|}$$

This movement direction leads the fish towards regions of lower pollution concentration, allowing them to optimize their location based on the negative gradient of the pollution field, in order for their immunity to be higher than the pollution at that spatial aggregation.