Supplementary material: Emergence of Lanes and Turbulent-like Motion in Active Spinner Fluid

Cody J. Reeves[a], Igor S. Aranson[b], Petia M. Vlahovska[a]
[a] *Engineering Sciences and Applied Mathematics, Northwestern University, Evanston, IL 60208, USA. E-mail: petia.vlahovska@northwestern.edu*
[b] *Department of Biomedical Engineering, Pennsylvania State University, University Park, Pennsylvania 16802, USA*

(Dated: October 20, 2020)

MODEL EQUATIONS

$$\frac{\partial \omega^*}{\partial t^*} + (\mathbf{u}^* \cdot \nabla^*)\omega^* = D^* \nabla^{*2} \omega^* + \kappa\Big((\nabla^* \times \mathbf{u}^*)_z - 2\omega^* \Big) + 2\kappa\gamma (\mathbf{P}^* \times \hat{\mathbf{x}})_z \tag{1}$$

$$\mathrm{Re}\bigg(\frac{\partial \mathbf{u}^*}{\partial t^*} + (\mathbf{u}^* \cdot \nabla^*)\mathbf{u}^* \bigg) = \nabla^{*2}\mathbf{u}^* - \nabla^* p^* + 4\alpha \nabla^* \times (\omega^* \hat{\mathbf{z}}) \tag{2}$$

$$\nabla^* \cdot \mathbf{u}^* = 0 \tag{3}$$

$$\frac{\partial \mathbf{P}^*}{\partial t^*} + (\mathbf{u}^* \cdot \nabla^*)\mathbf{P}^* = D_P^* \nabla^{*2} \mathbf{P}^* + \mathbf{P}^*_{eq} - \mathbf{P}^* + \omega^* \hat{\mathbf{z}} \times \mathbf{P}^* \tag{4}$$

where

$$\mathbf{P}^*_{eq} = -\gamma^{-1} \Big(\hat{\mathbf{x}} + \frac{\omega^*}{|\omega^*|} \sqrt{\gamma^2 - 1}\, \hat{\mathbf{y}} \Big) \tag{5}$$

NUMERICAL IMPLEMENTATION

All numerical simulations are run using a square domain of (nondimensionalized) length L, double periodic boundary conditions, and a grid of N by N points, $N = 1024$. Tow box sizes are used to check convergence, $L = 480$ and $L = 300$. The code is highly parallelized using CUDA in order to run on a NVIDIA graphics card. All Fourier and inverse transforms are computed using the Fast Fourier Transform (FFT).

Unless otherwise noted, all spatial derivatives are computed using a pseudo-spectral method. Spectral methods are optimal for smooth systems with several orders of derivatives. For a variable with p defined derivatives, the pseudo-spectral approximation of the qth derivative is of order $(L/2\pi N)^{2p-2q}$. If there is high frequency, it can be resolved by refining the spatial resolution of the grid or by using a low pass filter.

Eqs. (1-4) are solved using first order exponential time differencing. For simplicity, let us first consider a general PDE for a variable A(x,y,t), Eq. (6), such that the function $G(A, B, C)$ includes any nonlinear terms and dependence on other variables B(x,y,t) and C(x,y,t). D and c are the diffusion and linear coefficients respectively.

$$\partial_t A = D\nabla^2 A - cA + G(A, B, C) \tag{6}$$

Let $\mathcal{A}_{l,j}$ be the Fourier transform of A for wavenumbers l and j such that

$$\mathcal{A}_{l,j}(t) = \frac{1}{L^2} \int_{-L/2}^{L/2} \int_{-L/2}^{L/2} A(x, y, t) e^{-i(lx+jy)} dxdy. \tag{7}$$

The wavenumbers, l and j, are defined as $2\pi m_l/L$ and $2\pi m_j/L$ for some integers m_l and m_j. Then, Eq. (8) can be derived by taking the two-dimensional Fourier transform of Eq. (6), such that $k^2 = l^2 + j^2$ and $\mathcal{G}_{l,j}$ is the Fourier transform of the function G.

$$\partial_t \mathcal{A}_{l,j} = -(Dk^2 + c)\mathcal{A}_{l,j} + \mathcal{G}_{l,j} \tag{8}$$

By multiplying Eq. (8) by $\exp(-c_k t)$ and integrating with respect to time from t^n to $t^{n+1} = t^n + \Delta t$, the solution can then be written as

$$\mathcal{A}_{l,j}(t^{n+1}) = \mathcal{A}_{l,j}(t^n) e^{c_k \Delta t} + e^{c_k \Delta t} \int_0^{\Delta t} e^{-c_k t'} \mathcal{G}_{l,j}(t^n + t') dt' \tag{9}$$

such that $c_k = -(Dk^2 + c)$. It is worth noting that Eq. (9) is an exact solution and there have been no numerical approximations yet. However since $\mathcal{G}_{l,j}$ is often too complicated to write explicitly, we will assume the first order approximation that $\mathcal{G}_{l,j}$ is roughly constant. Furthermore, we define $\mathcal{A}^n = \mathcal{A}(t^n)$.

$$\mathcal{A}^{n+1}_{l,j} = \mathcal{A}^n_{l,j} e^{c_k \Delta t} + \mathcal{G}^n_{l,j} \big(e^{c_k \Delta t} - 1 \big)/c_k \tag{10}$$

A^{n+1} can then be evaluated by taking the inverse transform of $\mathcal{A}^{n+1}$ This numerical method has the advantage that the first term, accounting for the diffusion and linear terms, is exact. The second term however is $\mathcal{O}(\Delta t)$

due to the approximation of the integral in Eq. (9). This method can be improved to second order by including a $\mathcal{G}_{l,j}^{n-1}$ term in Eq. (10). In order to ensure stability for Eq. (10), $|c_k \Delta t| < 1$ must be satisfied. If we use the FFT to approximate Eq. (7), then $k < 2\sqrt{2}\pi N/L$ for all values k. In this case, stability is ensured for Eqs. (1-4) provided that

$$\frac{N}{L} < \frac{1}{2\sqrt{2}\pi}\left(\Delta t^{-1} - 2\kappa\right)^{1/2}. \tag{11}$$

In the case, Re $\neq 0$, in order to ensure the incompressibility condition for the fluid flow, we will use the Chorin's Projection method to solve Eqs. (2-3). This is done by breaking Eq. (2) into two different steps with a prediction/intermediate variable $\mathbf{u}^*$.

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -(\mathbf{u}^n \cdot \nabla)\mathbf{u}^n + \frac{1}{\mathrm{Re}}\nabla^2 \mathbf{u}^n + \frac{\alpha}{\mathrm{Re}}(\nabla \times \omega^n) \tag{12}$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\frac{1}{\mathrm{Re}}\nabla p^{n+1} \tag{13}$$

Given $\mathbf{u}^n$, $\mathbf{u}^*$ can be evaluated using Eq. (12). But in order to evaluate $\mathbf{u}^{n+1}$, we must first evaluate p^{n+1}. By Eq. (3), we know that $\nabla \cdot \mathbf{u}^{n+1} = 0$. Therefore, if we take the divergence of Eq. (13) then we get the following

$$\nabla \cdot \mathbf{u}^* = \frac{\Delta t}{\mathrm{Re}}\nabla^2 p^{n+1}. \tag{14}$$

If we define, respectively, $\mathcal{U}^*$, $\mathcal{V}^*$, and $\mathcal{P}^{n+1}$ as the Fourier transform for the x and y component of $\mathbf{u}^*$ and p^{n+1}, then Eq. (14) can be rewritten as

$$\mathcal{P}_{l,j}^{n+1} = i\frac{\mathrm{Re}}{k^2 \Delta t}(l\, \mathcal{U}_{l,j}^* + j\mathcal{V}_{l,j}^*). \tag{15}$$

Since we only care about the gradient of p, we set the zeroth Fourier mode so that $\mathcal{P}_{0,0}^{n+1} = 0$. It is then straightforward to compute $\mathbf{u}^{n+1}$ from Eq. (13).

In the case of Stokes flow, Re $= 0$, we use streamfunction formulation, $\mathbf{u} = \nabla \times (\psi \hat{\mathbf{z}})$, which automatically satisfies the flow incompressibility condition $\nabla \cdot \mathbf{u} = 0$.