This readme file will help replicate the results in the paper titled: "Stability of schooling patterns of a fish pair swimming against a flow".

Directory **“/Code/”** contains all the codes required to replicate the results of this work.

The programs and functions are detailed below:

## “SampleSimulations.m”:

This code calculates and plots sample trajectories for specified initial conditions.

## “Find\_AllRoots\_alph.m”:

This code finds the equilibria of the system of equations at different values of alpha for a given value of lambda.

## “Plot\_equil\_vs\_alph.m”:

This code plots the cross-stream coordinate (xi) of stable/unstable equilibria of the system as a function of the flow parameter alpha.

## “Find\_AllRoots\_lam\_alph\_fine.m”, “Find\_AllRoots\_lam\_alph.m”::

This code finds the equilibria of the system of equations at different values of alpha and lambda.

## “Plot\_heatmap.m”:

This code plots the heatmap of the stable configurations of the system in alpha-lambda plane.

## “Write\_nat\_freq.m”:

This code computes the natural frequencies of stable equilibria at different values of alpha-lambda.

## “Plot\_nat\_freq.m”:

This code plots the contour plots of natural frequencies of stable equilibria in alpha-lambda plane.

## “Stab\_boundary\_alph\_cr.m”:

This code finds the alpha critical values for different Lambda.

## “Plot\_Stab\_boundary\_alphcr.m”:

This code plots alpha critical vs Lambda for different Kappa (fixed rho) and for different rho (fixed Kappa)

## “Write\_zero\_eigenvec\_lam.m”:

This code finds the component of zero eigenvector of stable equilibria along lambda.

## “Plot\_zero\_eigenvec.m”:

This code plots the component of zero eigenvector of stable equilibria along lambda.

## “Stab\_boundary\_Lam1\_alph.m”:

This code finds the stability boundary curves Lambda\_1,2,3 for every alpha.

## “Plot\_Stab\_boundary\_curves.m”:

This code plots the stability boundary curves Lambda\_1,2,3 as functions of alpha for different Kappa and rho values.

## “uniqueroots.m”:

This is a function that finds the unique solutions of the system of equations among all the solutions numerically obtained

## “equationmap\_Tdip.m”:

This function checks whether a given state of the system is an equilibrium solution of the system and returns 1 if it is and 0 if not.

## “ODEfive\_Tdip.m”:

This function returns the system governing equations.

## “func\_jacob\_Tdip.m”:

This function returns the Jacobian matrix at a given state of the system in a vectorized form.

## “jac\_Tdip.m”:

This function calculates and classifies the eigenvalues of the Jacobian matrix of the system at a given state.

## “plot\_lines.m”:

This function plots the stable (green) and unstable (red) equilibria.

## “func\_SOLVE5eq\_var4\_Tdip.m”:

This function returns the system five equations for finding the solutions/equilibria of the system.

## “Model\_derivation.nb”:

This Mathematica code shows the complete derivation of the model equations for non-dimensionalized position and orientation of both the fish.

## “Governing\_equations.nb”:

This Mathematica code contains the five governing equations of the dynamical system and some manipulations to determine the terms in the form expressed in the manuscript.