

Simulations de fluides en 2D

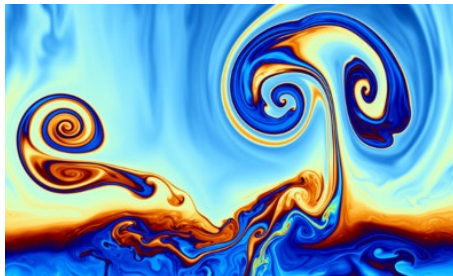
Encadrant : Marc Hartley (marc.hartley@umontpellier.fr)

Objectif du projet

Les phénomènes de mécanique des fluides sont omniprésents dans la nature (fumée, écoulement d'eau, diffusion de chaleur...) et constituent un domaine majeur de la simulation numérique.

L'objectif de ce projet est de concevoir et développer une simulation 2D interactive de fluide reposant sur une modélisation simplifiée des équations de la dynamique des fluides (type Navier-Stokes).

Le système devra permettre de visualiser le comportement du fluide au cours du temps, d'injecter des particules ou des forces, et d'observer la diffusion, l'advection et la dissipation du mouvement.



Objectifs fonctionnels

Les objectifs de ce projet sont :

- Représenter un champ de fluide 2D sur une grille régulière (vitesse, densité, pression).
- Simuler l'évolution temporelle du fluide selon des étapes typiques : advection (transport du fluide par sa propre vitesse), diffusion (effet de viscosité), projection (ajustement pour garantir un champ de vitesse incompressible).
- Permettre l'interaction avec la simulation par l'ajout de matière ou de couleur à l'écran et l'application de forces locales (souffle, vortex, obstacles).
- Visualiser le résultat sous forme d'animation 2D en temps réel (champ de densité, vecteurs de vitesse, lignes de courant, etc.).
- Analyser les performances et la stabilité de la simulation (influence du pas de temps, de la taille de grille, de la viscosité).

Extensions possibles

Le projet peut naturellement donner lieu à de nombreuses extensions pour les groupes souhaitant aller plus loin :

- Visualisation : affichage du champ de vitesse, lignes de flux, couleurs dynamiques, export vidéo.
- Simulation de phénomènes dérivés : fumée, chaleur, diffusion thermique, fluides multi-couleurs.
- Version web ou API : serveur proposant des rendus ou animations à la demande.
- Approche alternative : simulation de fluide par particules (Smoothed Particle Hydrodynamics).

Contraintes techniques

Le projet pourra être développé en C++ (recommandé), Python, Javascript ou autre.

Le C++ est conseillé pour ses performances supérieures, utiles aux simulations interactives ou à grande grille, mais ce n'est pas une contrainte.

Les autres langages restent acceptés s'ils permettent une solution claire, fonctionnelle et bien structurée.

Quel que soit le choix, le code devra être modulaire, documenté et accompagné d'une analyse des performances (temps de calcul, stabilité, limites).