

Stabilité Hydrodynamique. MSF2 I J. Hoepffner

Université Pierre et Marie Curie.
Année 2012-2013. TP info: Rayleigh-Taylor

Gerris

Vous disposez de «Gerris», un très bon logiciel open-source collaboratif pour simuler les écoulements (Navier-Stokes avec possibilité d'écoulements diphasiques). Commencez avec le fichier de paramètres qu'on vous fourni «rayleightaylor.gfs». C'est un des exemples donnés sur le site web de Gerris: <http://gfs.sourceforge.net>

Dans le fichier de paramètre, tout est expliqué en détail. Vous pouvez changer toutes les valeurs des paramètres si vous le voulez. Les conditions aux bords sur les côté sont «symétriques», donc ici on simule une seule longueur d'onde, mais c'est équivalent à une onde périodique infinie selon la direction horizontale.

Pour lancer le calcul: tapez dans le terminal et dans le bon répertoire:
gerris2D rayleightaylor.gfs | gfsview2D

gfsview2D est le logiciel qui fait l'interface graphique pour regarder comment ça évolue dans le temps. On peut tout regarder, la pression, le champs de vitesse et ainsi de suite...

Une fois que le calcul est fini, vous disposez dans le répertoire des fichiers de données, par exemple rayta-0.70.gfs pour le temps 0.7. Pour visualiser ces données avec gfsview, tapez dans le terminal:
gfsview2D rayta-%3.2f.gfs
cela vous donne la fenêtre «play» avec forward et backward pour avancer et reculer dans le temps.

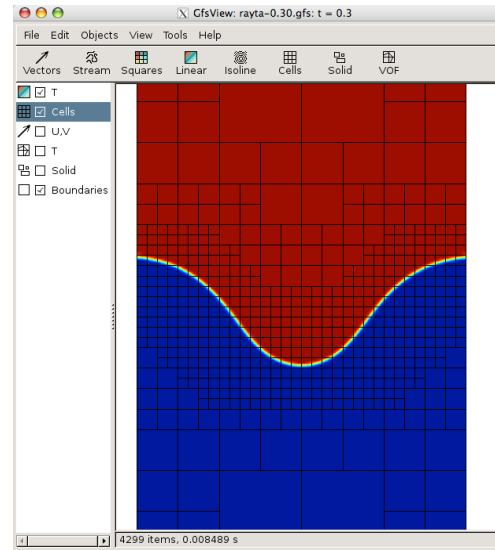
Dans gfsview: rouleur de la souris pour zoomer/dézoomer, clic gauche pour rotation, clic droite pour translation. CTRL+Clic vous indique en bas à gauche de l'image le x et le y et la valeur du champ au point ou vous cliquez (utile pour mesurer l'amplitude de déformation de l'interface par exemple).

Pour en savoir plus allez voir sur le site de Gerris.

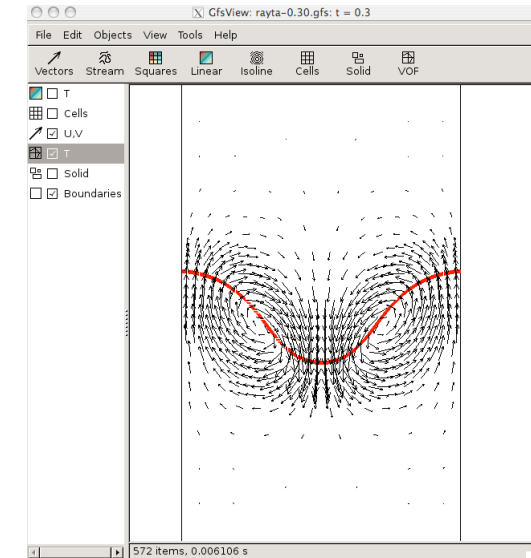


GfsView

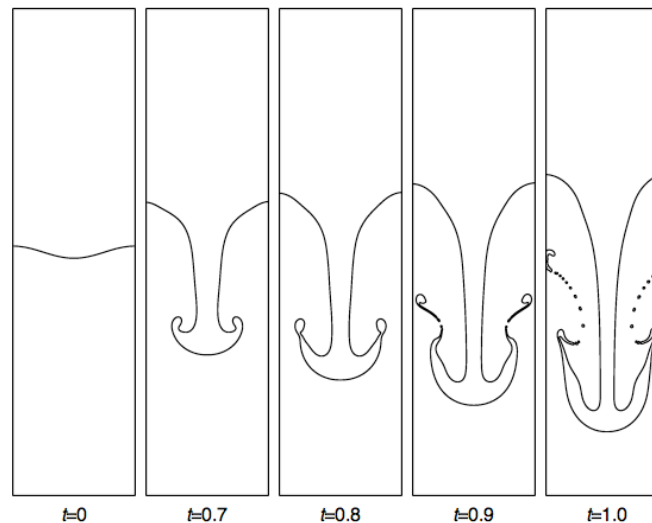
Voilà à quoi ressemble l'interface graphique de Gfsview2D: avec «linear» on trace avec des couleurs les champs comme le T (qui définit les deux fluides, rouge si $T=1$ et bleu si $T=0$) ou U (la vitesse horizontale) ou V (la vitesse verticale) etc. Avec «Cells» on regarde le maillage, c'est utile pour voir si c'est bien résolu. On peut aussi tracer le champ de vitesse avec un champ de vecteurs («vectors»)



Ici je montre l'interface avec une ligne rouge et le champ de vitesse. J'ai changé la couleur du fond en blanc.



Une figure qui montre le calcul par défaut avec le fichier de paramètre qu'on vous donne: le fluide lourd est en haut, et le fluide léger est en bas. Ce qui nous interesse, ici ce n'est pas le comportement fortement non linéaire comme représenté ici, c'est le tout début lorsque l'onde commence à grandir en amplitude.



Votre étude

Utilisez gerris pour vérifier la validité de notre théorie de stabilité linéaire. A garder en tête:

- restez dans les faibles amplitudes de perturbation (ne pas aller trop dans le non linéaire)
- vérifiez que l'amplitude de la perturbation croit exponentiellement dans le temps.
- faites varier les paramètres physiques : par exemple longueur d'onde, rapport de densité des deux fluides, intensité de la gravité. Et mesurez le taux de croissance exponentiel, et éventuellement la vitesse de phase (si la gravité est négative).
- tracez des graphiques qui comparent la théorie linéaire avec vos mesures de simulations.
- Vérifiez que la résolution que vous avez choisie est suffisante.

Première chose à faire: tracez dans le temps l'évolution de l'amplitude de la déformation de l'interface, et comparez avec une exponentielle avec le taux de croissance théorique.

Title Rayleigh-Taylor instability

toutes les lignes qui commencent avec le symbole "dièse" sont des commentaires
pour vous expliquer comment ça marche et comment faire varier les paramètres.

cette ligne débute le fichier de paramètres et précise qu'on utilise 4 boîtes
et qu'il y a 3 connections entre les boîtes. Les boîtes sont les unes au dessus
des autres

4 3 GfsSimulation GfsBox GfsGEdge {}

cette ligne dit que on fini la simulation au temps t=1 et qu'il ne faut pas
avoir un pas de temps plus grand que dtmax

Time { end = 1 dtmax = 5e-3 }

cette ligne décrit la résolution de la simulation pour la condition initiale: par défaut la
plus petite taille de maille c'est la taille d'une boîte divisée par 2 à la puissance 7

Refine 7

T est un traceur passif qui nous sert à suivre l'évolution de l'interface
il est défini dans toutes les mailles du domaine, et s'il a la valeur 1 on est dans
le premier fluide et s'il a la valeur 0 on est dans l'autre fluide

VariableTracerVOF {} T

Voilà comment on initialise les deux fluides: dans les mailles pour

lesquelles la fonction que l'on donne en argument est négative, on met T=0

et si elle est positive, on met T=1. «ty=0.5» sert juste à translater verticalement la fonction

InitFraction {} T (0.05*cos(2.*M_PI*x) + y) { ty = 0.5 }

ici ce sont deux commandes pour établir un maillage adaptatif: on change la

taille des mailles de calcul en fonction de la vorticité et de la distance à l'interface

maxlevel décrit la taille minimum de maille, comme pour "Refine" plus haut

AdaptVorticity { istep = 1 } { maxlevel = 7 cmax = 2e-2 }

AdaptGradient { istep = 1 } { maxlevel = 7 cmax = 1e-2 } T

Voici la valeur de la viscosité dynamique

SourceViscosity {} 0.00313

Une commande qui sert dire à Gerris la taille des boîtes de calcul: ici des

carrés de 1 sur 1, et la densité des deux fluides: ici 1.225 pour le fluide lourd

(dans les mailles pour lesquelles on a T=1) et 0.1694 pour le fluide léger

(dans les mailles telles que T est égal à 0).

fonction of T

PhysicalParams { L=1 alpha = 1./(T*1.225 + (1. - T)*0.1694) }

Ici on donne la valeur de la gravité

Source {} V -9.81

cette ligne sert à afficher à l'écran l'évolution du calcul

tous les dix pas de temps

OutputTime { istep = 10 } stderr

cette ligne sert à donner à gfsview l'évolution de la simulation en temps réel

toutes les 0.05 unités de temps

OutputSimulation { step = 0.05 } stdout

Cette ligne sert à sauver les données de la simulation sur le disque toutes les

0.05 unités de temps. LE fichier est sauvé sous le nom rayta-%3.2f.gfs, ou

%3.2f est remplacé par la valeur du temps.

OutputSimulation { step = 0.05 } rayta-%3.2f.gfs

}

les commandes pour initialiser les 4 boîtes de calculs

GfsBox {}

GfsBox {}

GfsBox {}

GfsBox {}

les commandes pour indiquer comment les boîtes sont connectées entre elles

ici les unes au dessus des autres.

1 2 top

2 3 top

1 4 bottom