

# BAL<sup>L</sup>AÏTO<sub>U</sub>S :

un code de simulations numériques  
locales pour la dynamique des fluides stellaires

F. Rincon  
F. Lignières  
M. Rieutord

Laboratoire d'Astrophysique  
Observatoire Midi-Pyrénées  
UMR 5572 - TOULOUSE



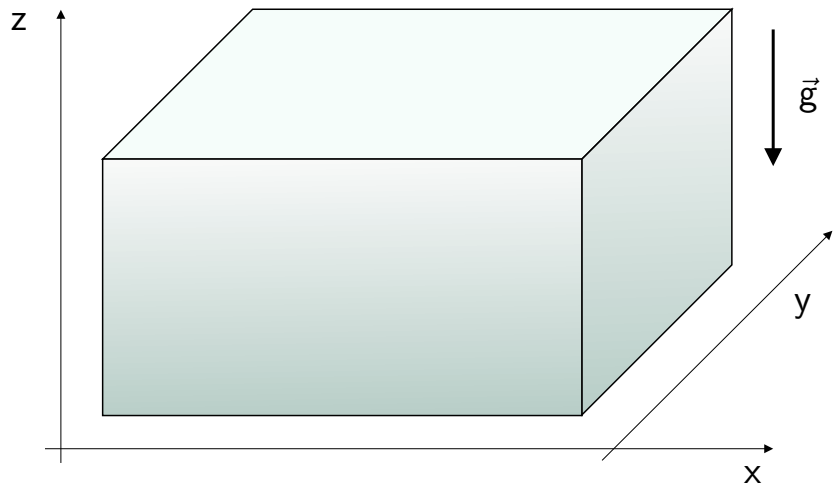
# Simulations numériques locales pour la dynamique des fluides stellaires

- ✓ Présentation générale
  - ✗ Motivations
  - ✗ Historique et évolutions récentes
  - ✗ Types de simulations réalisables
  - ✗ Structure du code
- ✓ Méthodes numériques
  - ✗ Avancement en temps
  - ✗ Différentiation spatiale
  - ✗ Parallélisation
- ✓ Exemples d'utilisations récentes
  - ✗ Tests, simulations à grand rapport d'aspect
- ✓ Conclusions, évolutions possibles

# Motivations

- ✓ Un **outil numérique polyvalent** et **performant** pour la modélisation des écoulements fluides 3D dans les **atmosphères** et les **intérieurs stellaires** ;
- ✓ **simulations locales** avec **une direction inhomogène** (celle de la gravité) :

- ✗ **géométrie cartésienne**,
- ✗ **deux directions horizontales périodiques**,
- ✗ **direction verticale avec conditions aux limites** ;



- ✓ codes similaires : HPS (Chicago, Boulder), code du DAMTP (Cambridge).

# Historique et évolutions récentes

- ✓ Initié par [A. Mangeney](#) et [F. Califano](#) pour étudier des écoulements incompressibles [Califano, 1996];
- ✓ développé par [F. Lignières](#) : passage en fortran 90, parallélisation MPI, portage sur IBM SP3/4 [Lignières, 1998];
- ✓ récemment : intégration des écoulements compressibles, portage sur SGI, conception de tests de performance et de précision.

# Types de simulations réalisables

✓ Résolution en 2D ou 3D :

✗ de l'équation de continuité,

✗ de l'équation de Navier-Stokes,

✗ éventuellement de l'équation pour la température  
(dans l'approximation de diffusion),

✗ éventuellement de l'équation d'induction,

pour des conditions aux limites de type champ fixé ou dérivée fixée.

⇒ flots incompressibles ou dans l'approximation de Boussinesq,

⇒ écoulements complètement compressibles,

⇒ simulations purement HD ou MHD,

⇒ écoulements forcés ou avec rotation ;

✓ code sans viscosité artificielle :  $\nu \Delta \vec{v}$  seulement.

# Structure du programme

- ✓ Langage : **fortran 90**, interfacé avec du C ;
- ✓ **Modularité** : possibilité de changer les méthodes de différentiation spatiale sans affecter l'ensemble du code :
  - ⇒ utilisation d'**interfaces génériques** ;
- ✓ **preprocessing** du compilateur pour la **sélection des méthodes** numériques ;
- ✓ configuration d'une partie du problème par des namelist.
  
- ✓ **Installation** sur diverses architectures et **benchs** de performances et d'exactitude **automatisés en Perl**.

# Avancement en temps

- ✓ Complètement **explicite** ;
- ✓ prédicteur correcteur de type **Runge-Kutta** [Demuren et al., 2001] :
  - ✗ faible stockage mémoire,
  - ✗ ordre 3 ou 4 ;
- ✓ **avancement** dans **l'espace réel** (différence avec HPS).
  
- ✓ Différents algorithmes :
  - ✗ en **incompressible**, résolution intermédiaire de **l'équation de Poisson** pour la pression ;
  - ✗ en **compressible**, avancement **direct** des différents champs, mais un **champ supplémentaire  $\rho$**  à calculer.

## Différentiation spatiale dans la direction verticale – différences finies compactes

- ✓ Schéma possédant une **précision quasi-spectrale** [Lele, 1992] :
  - ✗ **résolution** satisfaisante des **petites échelles**, adapté aux problèmes de **turbulence** ;
  - ✗ **stable** pour une équation d'advection [Carpenter, 1993] ;
  - ✗ choix entre ordre 6 et ordre 8 ;
  - ✗ intégration de **plusieurs schémas de bords** pour la gestion de **différents types de conditions aux limites** ;
  
- ✓ en pratique, résolution de **systèmes linéaires tri- ou penta-diagonaux** :
  - ✗ peu ou pas parallélisable ;
  - ✗ choix d'une bibliothèque : **LAPACK (PC)**, **SCS (SGI)**, **ESSL (IBM)**.



# Différentiation spatiale dans les directions horizontales – transformée de Fourier rapide

## ✓ Précision spectrale :

- ✗ utile pour les simulations de turbulence ;
- ✗ moins de points !
- ✗ facteurs de sécurité plus petits...

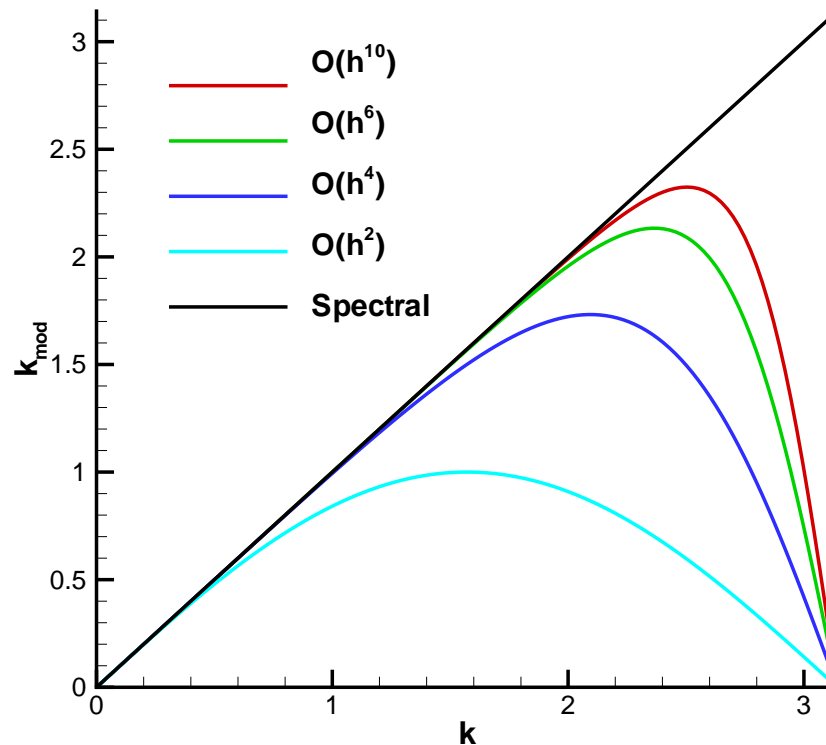
Équation	RK <sub>3</sub>			RK <sub>4</sub>		
	DFC <sub>6</sub>	DFC <sub>8</sub>	FFT	DFC <sub>6</sub>	DFC <sub>8</sub>	FFT
Diffusion	0.36	0.33	0.25	0.42	0.39	0.29
Advection	0.87	0.78	0.55	1.43	1.29	0.9

- ✓ hautement parallélisable ;
- ✓ utilisation de la bibliothèque **FFTW** reconnue pour ses performances comparées aux FFT propriétaires (interface C/F90).

# Précision des schémas et dealiasing

✓ Différentiation :

$$f'_k{}^{\text{SC}} \hat{=} i k_{\text{mod}} f_k^{\text{SC}}$$

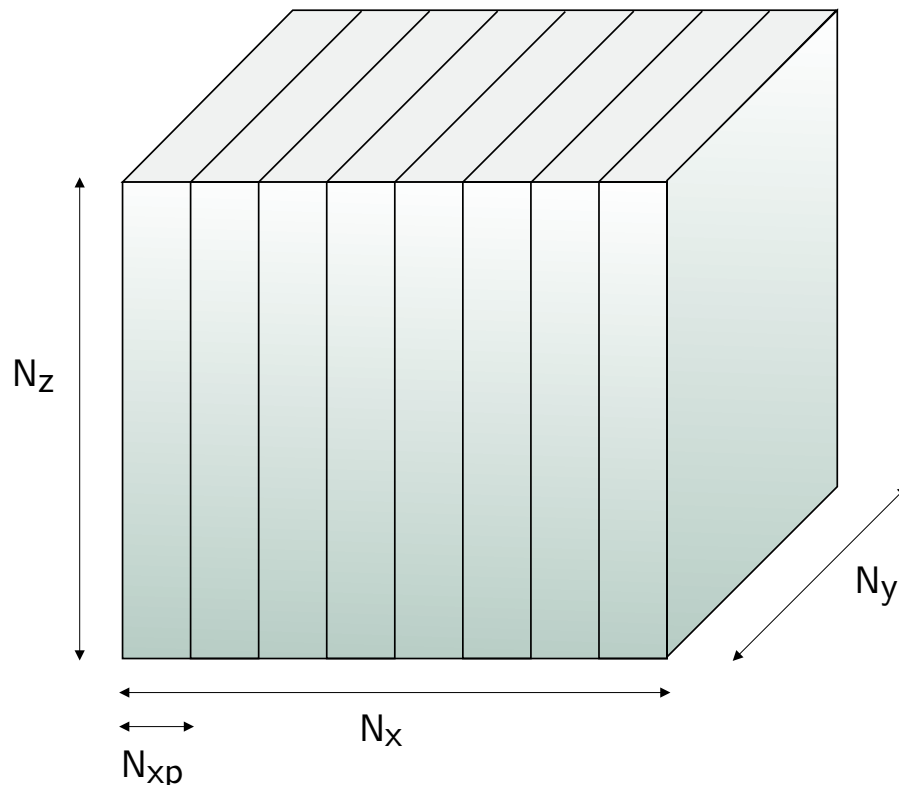


[R. Samtaney]

- ✓ DFC : filtre passe-bas « naturel » ;
- ✓ FFT : **filtre de dealiasing** pour supprimer les modes spurieux liés à la collocation dans l'espace réel.

# Stratégie de parallélisation

- ✓ les dérivées verticales dans chaque tranche sont calculées sur un processeur ;
- ✓ Parallélisation des FFT pour le calcul des dérivées horizontales ;

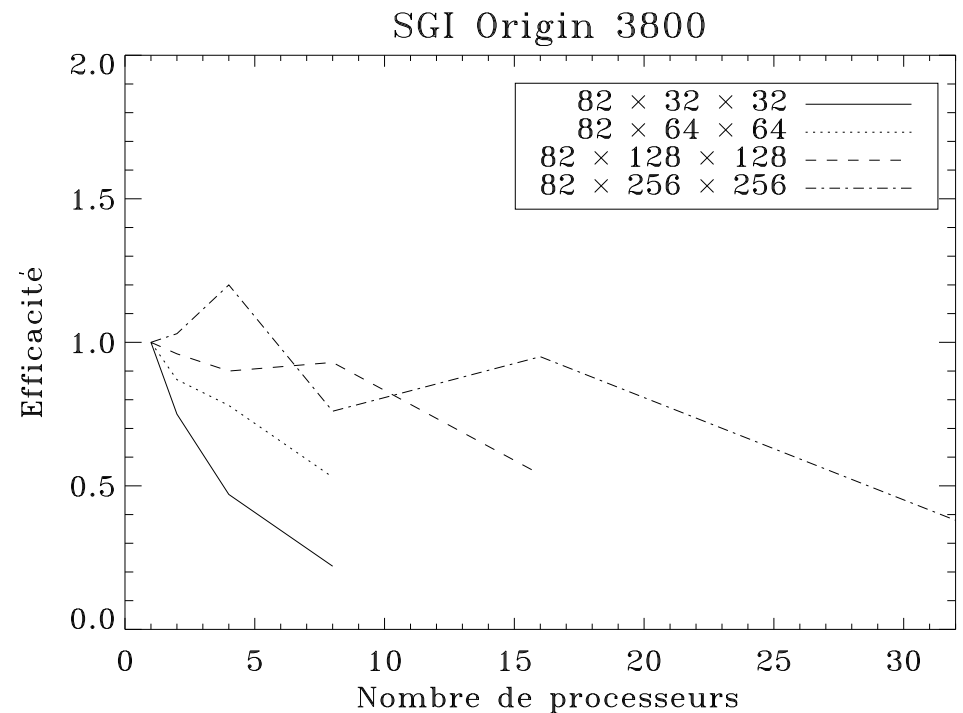
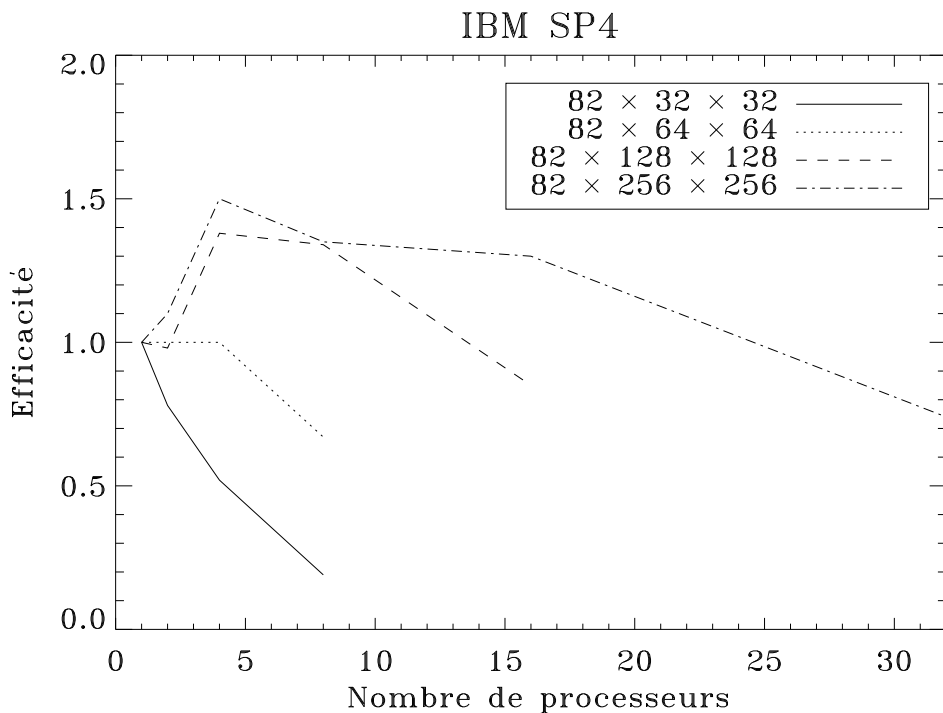


⇒  $N_{xp} \times N_y \times N_z$  points par processeur

- ✓ portabilité : utilisation de la bibliothèque MPI ;
- ✓ implémentation des communications interprocesseur intégrée à FFTW.

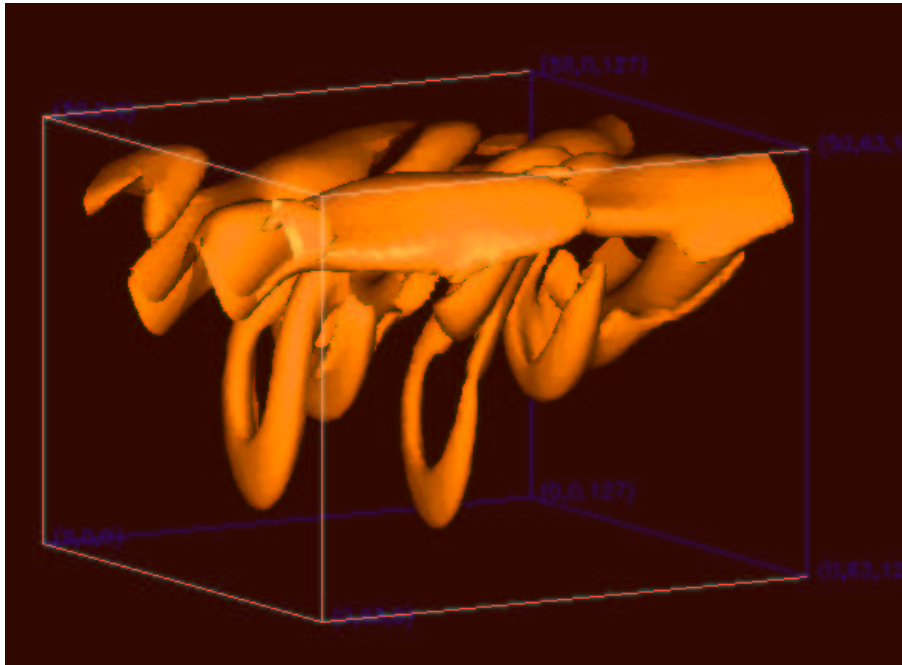
# Performances

- ✓ BENCHS de parallélisation effectués à l'IDRIS et au CINES :



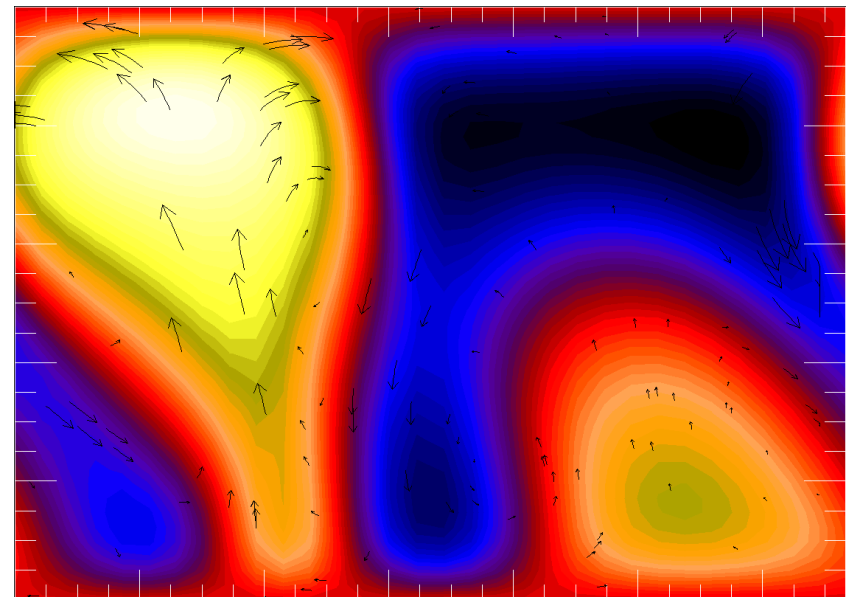
- ✓ Efficacité assurée jusqu'à 32 processeurs pour des résolutions  $82 \times 256 \times 256$  ;
- ✓ utilisation de plus de processeurs envisageable à des résolutions supérieures, résolutions horizontales  $N_p^2$  idéales ;
- ✓ performances de l'IBM supérieures pour le parallélisme.

## Exemples d'utilisations récentes



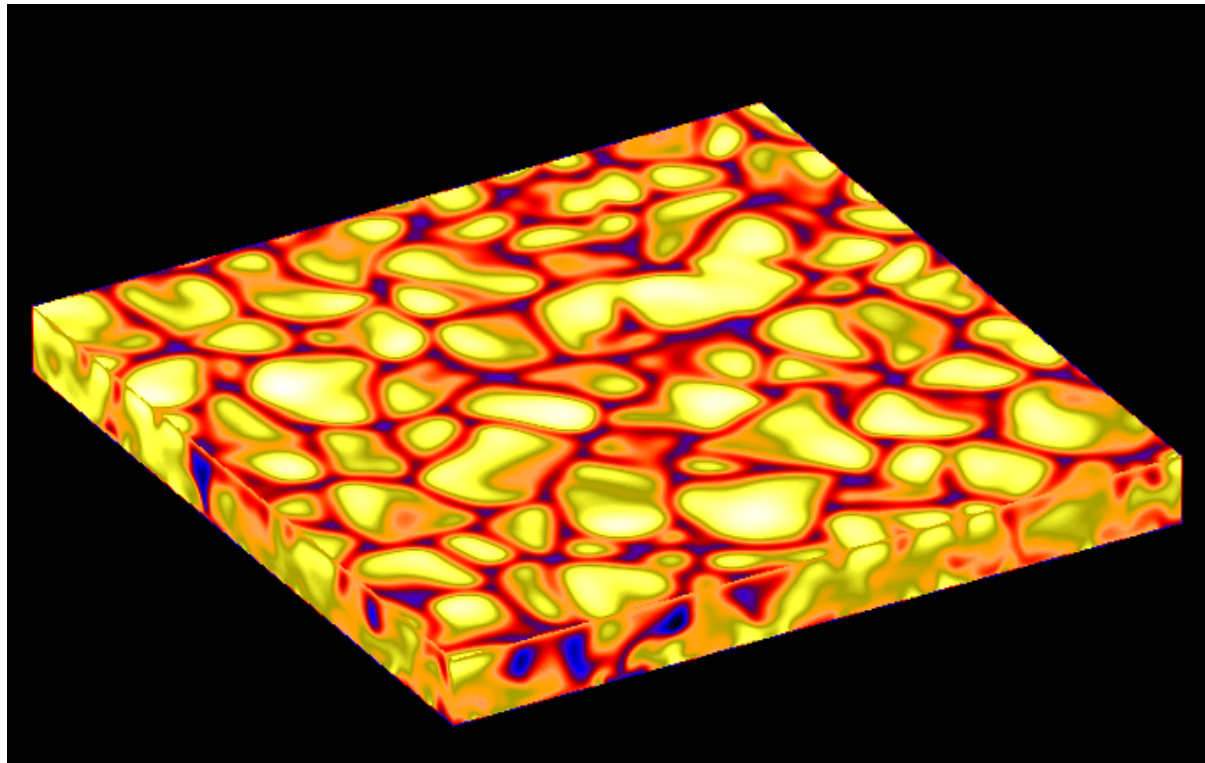
✓ Déstabilisation 3D d'une couche de cisaillement (F. Lignières)

✓ Tests d'exactitude sur la convection oscillante MHD compressible [Weiss, 1991]



## Exemples d'utilisations récentes (2)

- ✓ Simulations à grand rapport d'aspect de convection compressible en milieu fortement stratifié (F. Rincon)



## Conclusions, évolutions possibles

- ✓ Code adapté à la modélisation d'un grand nombre d'écoulements ;
- ✓ méthodes numériques précises et performantes sur plusieurs architectures de calcul scientifique ;
- ✓ très bonne parallélisation, hautes résolutions  $\mathcal{O}(500^3)$  possibles ;
- ✓ « user friendly ».
  
- ✓ Un futur plus ou moins proche/lointain :
  - ✗ comparer les performances avec d'autres codes ;
  - ✗ codage implicite des termes visqueux ;
  - ✗ parallélisation verticale : différences finies classiques ;
  - ✗ inclusion de transfert radiatif ?
  - ✗ géométrie sphérique ? cf. [Nishikawa, 2002] pour une utilisation des FFTs en sphérique.