

BONNET/TERRAS Fanny  
GM5  
INSA de Rouen

**ETUDE**  
**DE LA STABILITE NUMERIQUE**  
**DU CODE OPA-TRC-NNPZDDOM**

*pour le compte du*

Pôle de modélisation  
de l'Institut Pierre Simon Laplace

Décembre 1999-juin 2000

# Résumé

Ce projet de fin d'étude vise à étudier la stabilité numérique du code OPA-TRC-NNPZDDOM au moyen de la bibliothèque CADNA.

Le code OPA-TRC-NNPZDDOM est un code de simulation de la circulation océanique (OPA version 8.1) couplé à un code de traceurs passifs, qui dans cette étude sont des traceurs biologiques. Ces deux codes ont été créés au LODYC (le Laboratoire d'Océanographie DYnamique et de Climatologie), laboratoire géré par l'IPSL (l'Institut Pierre-Simon-Laplace des Sciences de l'Environnement Global).

Après avoir présenté le LODYC et l'IPSL, le code OPA-TRC-NNPZDDOM est étudié du point de vue théorique, en particulier la méthode du gradient conjugué préconditionné (utilisé pour calculer une fonction de courant) et les critères de stabilité des différents schémas numériques.

Dans la partie traceurs passifs du code, deux schémas numériques peuvent être utilisés : le schéma d'Arakawa et le schéma de Smolarkiewicz.

La bibliothèque CADNA, basée sur la méthode CESTAC (Contrôle et Estimation Stochastique des Arrondis de Calculs), est ensuite présentée.

Une dernière partie présente l'ensemble des résultats obtenus. L'algorithme du gradient conjugué préconditionné 'calcule bien' du point de vue informatique (mise à part lors du calcul de la fonction de courant barotropique d'une île) même si la solution obtenue n'atteint pas toujours la précision maximale.

D'autre part les variables du code possèdent une bonne précision, que ce soit les variables dynamiques (traceurs actifs), les variables biologiques (traceurs passifs) avec le schéma numérique d'Arakawa ou les variables biologiques avec le schéma numérique de Smolarkiewicz. Il est intéressant de noter que les concentrations des variables biologiques ont environ un chiffre significatif exact de moins avec Smolarkiewicz qu'avec Arakawa.

# Table des matières

|   |           |
|---|-----------|
| <b>Remerciements</b>  | <b>4</b>  |
| <b>Introduction</b>   | <b>5</b>  |
| <b>1 Cadre de l'étude</b>                                       | <b>6</b>  |
| 1.1 Présentation de l'IPSL . . . . .                            | 6         |
| 1.2 Présentation du LODYC . . . . .                             | 6         |
| <b>2 Le code OPA, version 8.1</b>                               | <b>8</b>  |
| 2.1 Le domaine . . . . .  | 8         |
| 2.2 Les équations primitives . . . . .                          | 9         |
| 2.3 Repère local . . . . .                                      | 11        |
| 2.4 La méthode du gradient conjugué préconditionné . . . . .    | 13        |
| 2.5 Discrétisation . . . . .                                    | 16        |
| 2.6 Critères de stabilité . . . . .                             | 17        |
| 2.7 Fichiers associés . . . . .                                 | 17        |
| 2.8 CRAY SV1 . . . . .  | 18        |
| <b>3 Le code NNPZDDOM</b>                                       | <b>19</b> |
| 3.1 Le schéma d'Arakawa . . . . .                               | 20        |
| 3.2 Le schéma de Smolarkiewicz . . . . .                        | 22        |
| <b>4 La bibliothèque CADNA</b>                                  | <b>24</b> |
| 4.1 Le LIP6 . . . . .   | 24        |
| 4.2 L'arithmétique à virgule flottante . . . . .                | 25        |
| 4.3 La méthode CESTAC . . . . .                                 | 28        |
| 4.4 La bibliothèque CADNA . . . . .                             | 32        |
| <b>5 Instrumentation de CADNA dans le code OPA-TRC-NNPZDDOM</b> | <b>33</b> |
| 5.1 Méthode utilisée . . . . .                                  | 33        |
| 5.2 Erreur globale . . . . .                                    | 36        |
| 5.3 Convergence du gradient conjugué préconditionné . . . . .   | 37        |
| 5.4 Précision des différents schémas . . . . .                  | 41        |
| 5.5 Conservation des traceurs . . . . .                         | 45        |
| 5.6 Planning . . . . .  | 46        |
| <b>Conclusion</b>   | <b>47</b> |

# Remerciements

Je tiens à remercier tout particulièrement les personnes qui ont encadré ce projet de fin d'étude à Jussieu, Marie-Alice Foujols, Fabienne Jézéquel, Laurent Mémery et Marina Lévy, pour leur aide, leur soutien et leur sympathie.

Je tiens également à remercier M. Gleyse qui a encadré ce projet de fin d'étude, dans le cadre de mon école, pour les directions de recherche et les conseils judicieux qu'il m'a donnés.

Je tiens également à remercier toutes les personnes de l'Institut Pierre Simon Laplace (IPSL), du Laboratoire d'Océanographie DYnamique et de Climatologie (LODYC) et de l'équipe "Calcul à Haute Performance et Validation" du LIP6, ainsi que Clémentine Saïd pour leur aide et leur accueil chaleureux.

# Introduction

Les océans, qui couvrent 70% du globe et contiennent 97% de l'eau présente sur Terre, jouent un rôle primordial dans le système climatique. Le Laboratoire d'Océanographie DYnamique et de Climatologie (LODYC) a créé d'importants codes, à la pointe de la recherche, permettant de simuler les océans. La propagation des erreurs d'arrondi due à l'arithmétique à virgule flottante des ordinateurs peut fausser les résultats et engendrer des instabilités numériques. Pour pouvoir valider les codes de simulation des océans, il est important de connaître leur qualité numérique.

Une première étude [8] en 1998 a montré que l'on pouvait implanter la bibliothèque CADNA, qui permet de valider numériquement tout programme de calcul scientifique, dans une version simplifiée du code de simulation océanique OPA 8.0.

Une deuxième étude en 1999 [14] a permis de valider, du point de vue numérique, le code ORCA, version océan mondial du code OPA 8.1, grâce à cette bibliothèque. En effet, toutes les variables observées possédaient, en moyenne, plus de sept chiffres significatifs exacts et les instabilités numériques détectées n'avaient pas de conséquence grave.

Suite à ces deux études nous étudions ici le code OPA-TRC-NNPZDDOM c'est-à-dire le code OPA étudié précédemment, couplé avec un code simulant de la biologie marine : NNPZDDOM. Dans un premier temps le modèle correspondant au code OPA sera évoqué, puis celui concernant le code NNPZDDOM. Ensuite nous décrirons l'outil utilisé pour étudier la stabilité numérique du code : la bibliothèque CADNA. Enfin nous présenterons les différents résultats obtenus.

# Chapitre 1

## Cadre de l'étude

Le code OPA-TRC-NNPZDDOM a été développé au Laboratoire d'Océanographie DYnamique et de Climatologie (LODYC), et de l'Institut Pierre Simon Laplace (IPSL).

### 1.1 Présentation de l'IPSL

L'Institut Pierre-Simon-Laplace des Sciences de l'Environnement Global (IPSL) a été créé en tant que Fédération d'Unités par le CNRS le 1er Janvier 1994. Son rôle est de mettre en place des services scientifiques permettant une rationalisation des moyens et une meilleure efficacité dans la conduite des activités de recherche. Il regroupe six laboratoires de recherche localisés en région parisienne :

- Le Centre d'études des Environnements Terrestre et Planétaires (CETP)
- Le Laboratoire de Météorologie Dynamique (LMD)
- Le Laboratoire de Physique et Chimie Marines (LPCM)
- **Le Laboratoire d'Océanographie DYnamique et de Climatologie (LODYC)**
- Le Laboratoire des Sciences du Climat et de l'Environnement (LSCE)
- Le Service d'Aéronomie (SA)

### 1.2 Présentation du LODYC

Le LODYC est une Unité Mixte de Recherche (UMR 7617) dépendant du CNRS, département des Sciences de l'Univers, de l'Université Pierre et Marie Curie (Paris VI), et de l'Institut de Recherche pour le Développement (IRD). Il regroupe quelque 80 chercheurs, enseignants-chercheurs, ingénieurs, techniciens, administratifs et thésitifs localisés à Jussieu.

La vocation première du LODYC est l'étude des processus dynamiques gouvernant la circulation océanique, la compréhension des mécanismes gouvernant l'évolution du système climatique terrestre où l'océan joue un rôle important et l'étude des cycles biogéochimiques océaniques, en particulier celui du carbone, qui mettent en jeu, entre autres, la biosphère marine.

Les principaux projets du laboratoire relèvent :

- d'une activité expérimentale fondée sur l'implication forte du LODYC dans les campagnes à la mer ;
- d'une activité de développement instrumental ;
- de l'interprétation conjointe des observations spatiales et in situ ;
- d'une activité de modélisation théorique, conceptuelle et statistique avancée, liée à l'interprétation des données ;
- **d'une activité de modélisation numérique, le laboratoire ayant développé un modèle de circulation générale océanique : OPA.**

Le laboratoire participe activement à l'enseignement supérieur et l'accueil de thésitifs dans les domaines de l'océanographie, de la climatologie et de l'étude des cycles biogéochimiques océaniques.

# Chapitre 2

## Le code OPA, version 8.1

L'océanographie dynamique est une science récente. La structure détaillée des courants, les déplacements de masse d'eau, et la répartition des propriétés physiques et chimiques dans la mer sont loins d'être bien compris. Les processus physiques qui conduisent les courants et déterminent les propriétés physiques de l'eau de mer sont nombreux, complexes et se produisent au travers d'un vaste spectre spatio temporel. Ils résultent de la circulation induite par l'action du vent sur la surface de la mer et de la circulation liée à l'hétérogénéité de la température et de la salinité.

Comprendre et simuler l'océan possède des similitudes avec le problème de prévision du temps. C'est sans doute pour cela que, historiquement, beaucoup de techniques utilisées dans le domaine de la météorologie ont été appliquées au problème océanographique. Le modèle OPA permet d'étudier sur un vaste spectre spatio temporel la circulation océanique dans un cadre général. Ce code permet sous certaines contraintes de calculer l'évolution dans le temps de paramètres tels que la vitesse du courant, la température, la pression hydrostatique, la salinité.

### 2.1 Le domaine

Le domaine étudié dans le cadre de cette étude est un canal zonal périodique de 168 km de large, sur 504 km de long et 4,3 km de profondeur. On fait circuler l'eau qui arrive à un bord de l'autre côté. Le domaine est discrétisé en mailles de 6 km de large, par 6 km de long. Le pas d'espace en largeur et en longueur est constant. Par contre le découpage en profondeur n'est pas régulier : les mailles vont de 10 à 300 m de profondeur. On a donc une grille de  $28 \times 84 \times 30$  points.



Le découpage en profondeur se fait de la façon suivante :

| numéro de la maille | Profondeur prise au milieu de la maille |
|---------------------|---|
| 1                   | 4.98                                    |
| 2                   | 15.10                                   |
| 3                   | 25.54                                   |
| 4                   | 36.45                                   |
| 5                   | 48.07                                   |
| 6                   | 60.73                                   |
| 7                   | 74.94                                   |
| 8                   | 91.42                                   |
| 9                   | 111.22                                  |
| 10                  | 135.84                                  |
| 11                  | 167.35                                  |
| 12                  | 208.50                                  |
| 13                  | 262.83                                  |
| 14                  | 334.49                                  |
| 15                  | 427.90                                  |
| 16                  | 547.03                                  |
| 17                  | 694.55                                  |
| 18                  | 871.10                                  |
| 19                  | 1075.10                                 |
| 20                  | 1303.24                                 |
| 21                  | 1551.23                                 |
| 22                  | 1814.70                                 |
| 23                  | 2089.75                                 |
| 24                  | 2373.16                                 |
| 25                  | 2662.49                                 |
| 26                  | 2955.93                                 |
| 27                  | 3252.20                                 |
| 28                  | 3550.40                                 |
| 29                  | 3849.90                                 |
| 30                  | 4150.28                                 |

## 2.2 Les équations primitives

L'océan est un fluide qui est bien approché par les équations primitives, ie les équations de Navier-Stokes avec une équation d'état non-linéaire qui relie la température et la salinité à la vitesse du fluide par l'intermédiaire de la densité.

Le modèle OPA est défini avec les approximations suivantes :

- l'approximation de la sphéricité de la Terre
- l'approximation "couche mince" : la profondeur de l'océan est négligée par rapport au rayon de la Terre
- hypothèse de la fermeture turbulente : les tenseurs turbulents sont supposés être proportionnels aux gradients dynamiques et thermodynamiques à grande échelle

- hypothèse de Boussinesq : les variations de la masse volumique sont négligés dans l'équation de conservation de la quantité de mouvement horizontale
- hypothèse hydrostatique <sup>1</sup> : seuls la gravité et le gradient vertical de pression sont pris en compte dans l'équation de conservation de la quantité de mouvement verticale.
- hypothèse d'incompressibilité (d'où une vitesse non divergente)

Il est fréquent d'utiliser un système unitaire  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$  de vecteurs orthogonaux liés à la Terre du fait de l'importance de la force gravitationnelle dans les équations de mouvement à grande échelle. Ainsi  $\mathbf{k}$  est le vecteur local dirigé vers le haut et  $(\mathbf{i}, \mathbf{j})$  deux vecteurs orthogonaux à  $\mathbf{k}$ .

Les différentes variables sont définies de la manière suivante :

- $U$  le vecteur vitesse,  $U_h$  le vecteur vitesse sur l'horizontale (sur le plan  $(\mathbf{i}, \mathbf{j})$ ) et  $w$  le vecteur vitesse sur la verticale avec  $U = U_h + wk$
- $T$  la température
- $S$  la salinité
- $\rho$  la densité in-situ
- $p$  la pression

Voici les principales équations du modèle dans le repère  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$  :

L'équation de conservation du moment :

$$\frac{\partial U_h}{\partial t} = -[(\nabla \wedge U) \wedge U + \frac{1}{2} \nabla(U^2)]_h - f * k * U_h - \frac{1}{\rho_o} \nabla_h p + D^U \quad (2.1)$$

L'équilibre hydrostatique :

$$\frac{\partial p}{\partial z} = -\rho g \quad (2.2)$$

L'incompressibilité :

$$\nabla \cdot U = 0 \quad (2.3)$$

La conservation de la chaleur :

$$\frac{\partial T}{\partial t} = -\nabla \cdot (TU) + D^T \quad (2.4)$$

La conservation de la salinité :

$$\frac{\partial S}{\partial t} = -\nabla \cdot (SU) + D^S \quad (2.5)$$

L'équation d'état :

$$\rho = \rho(T, S, p) \quad (2.6)$$

avec  $\nabla$  le gradient dans les directions  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ ,  $t$  le temps,  $z$  la coordonnée verticale,  $\rho_o$  une densité de référence,  $f$  l'accélération de Coriolis ( $f = 2\Omega \cdot k$ , avec  $\Omega$  le vecteur de vitesse angulaire de rotation de la Terre),  $g$  l'accélération de la gravité et  $D^i$  le terme de diffusion pour la variable  $i$ .

---

<sup>1</sup>Un fluide est hydrostatique si on néglige ses mouvements verticaux

## 2.3 Repère local

Soit  $(i, j, k)$  un système de coordonnées curvilignes orthogonales sur la sphère, associé aux vecteurs orthogonaux  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ .

Soit  $(\lambda, \varphi, z)$  un système de coordonnées géographiques dans lequel une position est définie par la latitude  $\varphi(i, j)$ , la longitude  $\lambda(i, j)$  et la distance du centre de la Terre  $a + z(k)$  avec  $a$  le rayon de la Terre et  $z$  l'altitude au dessus d'un niveau de mer de référence.

La déformation locale du système de coordonnées curvilignes est donné par  $e_1$ ,  $e_2$  et  $e_3$ , les trois facteurs d'échelle :

$$e_1 = (a + z) \left[ \left( \frac{\partial \lambda}{\partial i} \cos \varphi \right)^2 + \left( \frac{\partial \varphi}{\partial i} \right)^2 \right]^{\frac{1}{2}} \quad (2.7)$$

$$e_2 = (a + z) \left[ \left( \frac{\partial \lambda}{\partial j} \cos \varphi \right)^2 + \left( \frac{\partial \varphi}{\partial j} \right)^2 \right]^{\frac{1}{2}} \quad (2.8)$$

$$e_3 = \left( \frac{\partial z}{\partial k} \right) \quad (2.9)$$

Comme la profondeur de l'océan est négligeable devant le rayon de la Terre,  $a + z$  peut être remplacé par  $a$ . Les facteurs d'échelle horizontaux,  $e_1$  et  $e_2$  sont donc indépendants de  $k$ . Le facteur d'échelle vertical,  $e_3$ , n'est fonction que de  $k$  puisque  $\mathbf{k}$  est parallèle à  $z$ . Dans notre cas, le domaine est assez petit pour considérer  $e_1$  et  $e_2$  constants. On se ramène donc à un système orthogonal.

Les opérateurs scalaires et vectoriels qui apparaissent dans les équations primitives (Equations (1.1) à (1.6)) peuvent être réécrits dans le formalisme tensoriel. Il est alors nécessaire de calculer la composante horizontale des termes visqueux et non-linéaires de l'équation du moment (1.1). On pose  $U = (u, v, w) = U_h + wk$  la vitesse dans le système de coordonnées  $(i, j, k)$ .

On obtient dans ce formalisme :

\* Avec  $\zeta$  la vorticité relative et  $\chi$  la divergence du champ horizontal de vitesse.

$$\chi = -\frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 u)}{\partial i} + \frac{\partial(e_1 v)}{\partial j} \right] \quad (2.10)$$

$$\zeta = \frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 v)}{\partial i} - \frac{\partial(e_1 u)}{\partial j} \right] \quad (2.11)$$

\* L'équation de conservation du moment :

$$\frac{\partial u}{\partial t} = +(\zeta + f)v - \frac{w}{e_3} \frac{\partial u}{\partial k} - \frac{1}{2e_1} \frac{\partial(u^2 + v^2)}{\partial i} - \frac{1}{\rho_o e_1} \frac{\partial p_h}{\partial i} - \frac{1}{\rho_o e_1} \frac{\partial p_s}{\partial i} + D_u^{lU} + D_u^{vU} \quad (2.12)$$

$$\frac{\partial v}{\partial t} = +(\zeta + f)u - \frac{w}{e_3} \frac{\partial v}{\partial k} - \frac{1}{2e_2} \frac{\partial(u^2 + v^2)}{\partial j} - \frac{1}{\rho_o e_2} \frac{\partial p_h}{\partial j} - \frac{1}{\rho_o e_2} \frac{\partial p_s}{\partial j} + D_v^{lU} + D_v^{vU} \quad (2.13)$$

avec  $D_i^{lU}$  et  $D_i^{vU}$  la diffusion horizontale, respectivement verticale, de la composante  $i$  de la vitesse et où  $\zeta$  est donné par (1.1) et le gradient de la pression de surface par :

$$\frac{1}{\rho_o e_1} \frac{\partial p_s}{\partial i} = \overline{M}_u + \frac{1}{H e_2} \frac{\partial}{\partial j} \left( \frac{\partial \Psi}{\partial t} \right) \quad (2.14)$$

et

$$\frac{1}{\rho_o e_2} \frac{\partial p_s}{\partial j} = \overline{M}_v - \frac{1}{H e_1} \frac{\partial}{\partial i} \left( \frac{\partial \Psi}{\partial t} \right) \quad (2.15)$$

Ici  $M = (M_u, M_v)$  représente la contribution des termes non-linéaire, visqueux et de gradient de pression hydrostatique aux équations (1.12) et (1.13). Et le surligné indique une moyenne verticale sur la colonne d'eau entière (i.e. de  $z=-H$ , le fond de l'océan, à  $z=0$ , le "toit rigide"<sup>2</sup>).

$$\overline{M}_u = \frac{1}{H} \int_{-H}^0 \left[ \frac{\partial u}{\partial t} + \frac{1}{\rho_o e_1} \frac{\partial p_s}{\partial i} \right] e_3 dk \quad (2.16)$$

$$\overline{M}_v = \frac{1}{H} \int_{-H}^0 \left[ \frac{\partial v}{\partial t} + \frac{1}{\rho_o e_2} \frac{\partial p_s}{\partial j} \right] e_3 dk \quad (2.17)$$

La dérivée temporelle de  $\psi$  est la solution de **l'équation elliptique** suivante :

$$\left[ \nabla \wedge \left[ \frac{1}{H} k \wedge \nabla \left( \frac{\partial \psi}{\partial t} \right) \right] \right]_z = [\nabla \wedge \overline{M}]_z \quad (2.18)$$

soit dans le formalisme tensoriel :

$$\frac{\partial}{\partial i} \left[ \frac{e_2}{H e_1} \frac{\partial}{\partial i} \left( \frac{\partial \psi}{\partial t} \right) \right] + \frac{\partial}{\partial j} \left[ \frac{e_1}{H e_2} \frac{\partial}{\partial j} \left( \frac{\partial \psi}{\partial t} \right) \right] = \frac{\partial}{\partial i} (e_2 \overline{M}_v) - \frac{\partial}{\partial j} (e_1 \overline{M}_u) \quad (2.19)$$

\* La vitesse verticale  $w$  et la pression hydrostatique  $p_h$  sont diagnostiqués par les équations suivantes :

$$\frac{\partial w}{\partial k} = -\chi e_3 \quad (2.20)$$

$$\frac{\partial p_h}{\partial k} = -\rho g e_3 \quad (2.21)$$

où  $\chi$  est donné par (1.10)

---

<sup>2</sup>on considère que la surface de l'océan ne varie pas suivant  $k$

\* Les équations de conservation de la chaleur de la salinité :

$$\frac{\partial T}{\partial t} = -\frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 T u)}{\partial i} + \frac{\partial(e_1 T v)}{\partial j} \right] - \frac{1}{e_3} \frac{\partial(T w)}{\partial k} + D_T \quad (2.22)$$

$$\frac{\partial S}{\partial t} = -\frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 S u)}{\partial i} + \frac{\partial(e_1 S v)}{\partial j} \right] - \frac{1}{e_3} \frac{\partial(S w)}{\partial k} + D_S \quad (2.23)$$

avec  $D_T$  le terme de diffusion pour la température et  $D_S$  le terme de diffusion pour la salinité.

## 2.4 La méthode du gradient conjugué préconditionné

Un algorithme de gradient conjugué préconditionné est utilisé pour résoudre l'équation elliptique((2.18)et(2.19)). Cette équation est de la forme  $Ax = b$ . avec A matrice symétrique définie positive. Trouver la solution x revient à minimiser la fonction [27]

$$f(x) = \frac{1}{2} x \cdot A \cdot x - b \cdot x \quad (2.24)$$

En effet, cette fonction est minimum quand son gradient,  $\nabla f = A \cdot x - b$  s'annule, ce qui revient à (2.24).

Le principe de la méthode du gradient conjugué est de chercher, à chaque étape, une quantité  $\alpha_k$  qui minimize  $f(x_k + \alpha_k p_k)$ . On a alors

$$\frac{df}{d\alpha} = 0$$

donc

$$\alpha_k = \frac{\langle r_k, r_k \rangle}{\langle A d_k, d_k \rangle}$$

où

$$r_k = b - A \cdot x_k = A(x - x_k)$$

est l'erreur au rang k. Le choix de la direction  $p_k$  dépend de l'erreur :

$$p_k = r_k + \beta_k p_{k-1}.$$

$\beta_k$  est tel que  $\langle A p_k, p_k \rangle = 0$  d'où

$$\beta_k = \frac{\langle r_k, r_k \rangle}{\langle r_{k-1}, r_{k-1} \rangle}$$

On obtient l'algorithme suivant :

\* Initialisations

$$x_0 \quad (2.25)$$

$$p_0 = r_0 = b - A x_0 \quad (2.26)$$

\* Pour  $k=0,1,\dots$

$$\alpha_k = \frac{\langle r_k, r_k \rangle}{\langle Ap_k, p_k \rangle} \quad (2.27)$$

$$x_{k+1} = x_k + \alpha_k p_k \quad (2.28)$$

$$r_{k+1} = r_k - \alpha_k Ap_k \quad (2.29)$$

$$\beta_k = \frac{\langle r_{k+1}, r_{k+1} \rangle}{\langle r_k, r_k \rangle} \quad (2.30)$$

$$p_{k+1} = r_{k+1} + \beta_{k+1} p_k \quad (2.31)$$

**L'algorithme du gradient conjugué converge en au plus  $n$  itérations, avec  $n * n$  la dimension de  $A$  [28]. Pratiquement, à cause des erreurs d'arrondis, cela n'est plus vérifié : on a une méthode itérative [28].**

Le test d'arrêt utilisé [7] est le suivant :

$$\frac{\langle r_{k+1}, r_{k+1} \rangle}{\langle b, b \rangle} \leq \epsilon \quad (2.32)$$

avec  $\epsilon$  choisi a priori.

### Le gradient conjugué préconditionné

L'algorithme utilisé dans OPA est en fait un gradient conjugué préconditionné. Il a pour avantage de converger en moins d'itérations si la matrice de préconditionnement est bien choisie [27].

On remplace la résolution de  $Ax = b$  par  $Q^{-1}Ax = Q^{-1}b$ .  $Q^{-1}$  doit être choisi avec l'objectif que  $K(Q^{-1}A) \ll K(A)$  avec  $K(A)$  le conditionnement de la matrice  $A$ .

En théorie le meilleur choix est donc  $Q^{-1} = A^{-1}$ . Dans ce cas  $K(Q^{-1}A) = 1$ .

En pratique, il faut trouver  $Q^{-1}$  le plus proche de  $A^{-1}$ , sans que les calculs de  $Q^{-1}$  ne soient trop coûteux. Dans notre cas **Q est la diagonale de A**.  $Q$  est alors trivialement inversible.

On ne peut appliquer directement l'algorithme du gradient conjugué à  $Q^{-1}A$ , car il faut que  $Q^{-1}A$  soit symétrique, ce qui est faux en général, même si  $Q^{-1}$  est symétrique.

Si  $Q^{-1}$  est symétrique définie positive, ce qui est notre cas, on peut définir  $Q^{-1/2}$  symétrique et définie positive telle que  $(Q^{-1/2})^2 = Q^{-1}$ .

Or

$$Q^{1/2}(Q^{-1}A)Q^{-1/2} = Q^{-1/2}AQ^{-1/2}$$

est symétrique définie positive. De plus  $Q^{-1}A$  est semblable à  $Q^{-1/2}AQ^{-1/2}$ . Donc au lieu d'utiliser le système  $Q^{-1}Ax = Q^{-1}b$ , on considère

$$Q^{1/2}(Q^{-1}A)Q^{-1/2}Q^{1/2}x = Q^{-1/2}b$$

On pose  $y = Q^{1/2}x$ . On doit alors trouver  $y$  tel que  $Q^{1/2}(Q^{-1}A)Q^{-1/2}y = Q^{-1/2}b$ .

La méthode du gradient conjugué est appliquée à ce nouveau système de matrice

$$\tilde{A} = Q^{-1/2} A Q^{-1/2}$$

On a alors

$$\begin{aligned} y_k &= Q^{1/2} x_k \\ \tilde{r}_k &= Q^{-1/2} b - \tilde{A} y_k = Q^{-1/2} r_k \\ \tilde{p}_k &= Q^{1/2} p_k \end{aligned}$$

avec  $r_k = b - A.x_k$

| Algorithme appliqué à $\tilde{A}$   | Idem en tenant compte des relations précédentes   |
|---|---|
| $\tilde{\alpha}_k = \frac{\langle \tilde{r}_k, \tilde{r}_k \rangle}{\langle \tilde{A} \tilde{p}_k, \tilde{p}_k \rangle}$  | $\tilde{\alpha}_k = \frac{\langle Q^{-1} r_k, r_k \rangle}{\langle A p_k, p_k \rangle}$     |
| $y_{k+1} = y_k + \tilde{\alpha}_k \tilde{p}_k$  | $x_{k+1} = x_k + \tilde{\alpha}_k p_k$  |
| $\tilde{r}_{k+1} = \tilde{r}_k - \tilde{\alpha}_k \tilde{A} \tilde{p}_k$  | $r_{k+1} = r_k - \tilde{\alpha}_k p_k$  |
| $\tilde{\beta}_{k+1} = \frac{\langle \tilde{r}_{k+1}, \tilde{r}_{k+1} \rangle}{\langle \tilde{r}_k, \tilde{r}_k \rangle}$ | $\beta_k = \frac{\langle Q^{-1} r_{k+1}, r_{k+1} \rangle}{\langle Q^{-1} r_k, r_k \rangle}$ |
| $\tilde{p}_{k+1} = \tilde{r}_{k+1} + \tilde{\beta}_{k+1} \tilde{p}_k$   | $p_{k+1} = Q^{-1} r_{k+1} + \tilde{\beta}_{k+1} p_k$  |

Dans OPA, c'est l'algorithme suivant, qui équivaut à ceux du tableau ci-dessus, qui est utilisé :

\* Initialisations

$$x_0 \tag{2.33}$$

$$z_0 = d_0 = b - Q^{-1} A x_0 \tag{2.34}$$

$$\gamma_0 = \langle z_0, Q z_0 \rangle \tag{2.35}$$

\* Pour  $k=0,1,\dots$

$$\alpha_k = \frac{\gamma_k}{\langle Q^{-1} A, Q p_k \rangle} \tag{2.36}$$

$$x_{k+1} = x_k + \alpha_k p_k \tag{2.37}$$

$$z_{k+1} = z_k - \alpha_k Q^{-1} A \tag{2.38}$$

$$\gamma_{k+1} = \langle z_{k+1}, Q z_{k+1} \rangle \tag{2.39}$$

$$\beta_k = \frac{\gamma_{k+1}}{\gamma_k} \tag{2.40}$$

$$p_{k+1} = z_{k+1} + \beta_{k+1} p_k \tag{2.41}$$

Tant que

$$\frac{\gamma_{k+1}}{\langle b, Q b \rangle} \leq \epsilon \tag{2.42}$$

avec  $\epsilon$  choisi a priori.

Il est à noter que le  $z_k$  utilisé ici correspond à  $r_k$  dans le code<sup>3</sup>, de même  $p_k$  correspond à  $d_k$  et il y a une variable supplémentaire dans le code :  $Q^{-1}Ap_k$  (donc  $Q^{-1}Ad_k$ ) qui est notée ' $z_k$ ' (à ne pas confondre!).

## 2.5 Discrétisation

Les techniques numériques utilisés pour résoudre les équations primitives sont basées sur la traditionnelle **méthode des différences finies** à l'ordre deux.

Le domaine est discrétisé sur la grille "C" dans la classification d'Arakawa.

$e_1x$ ,  $e_2x$  et  $e_3x$  correspondent à la largeur, respectivement la longueur, respectivement la hauteur de la maille ayant pour centre la variable  $x$ , pour  $i=1,2,3$ .

Notations :

Etant donné les valeurs d'une variable scalaire  $q$  en deux points adjacents de la grille du modèle, les opérateurs de moyenne et de différentiation appliqués à cette variable sont définis au point intermédiaire par :

$$\delta_i[q] = q \left( i + \frac{1}{2} \right) - q \left( i - \frac{1}{2} \right) \quad (2.43)$$

$$\bar{q}^i = \frac{1}{2} \left[ q \left( i + \frac{1}{2} \right) + q \left( i - \frac{1}{2} \right) \right] \quad (2.44)$$

La forme semi-discrète, en espace, de l'équation du moment est la suivante :

$$\begin{aligned} \frac{\partial u}{\partial t} = & + \frac{1}{e_{1u}} \overline{(\zeta + f)^j} \overline{(e_{1v}v)^{i+1/2,j}} - \frac{1}{2e_{1u}} \delta_{i+1/2} [\overline{u^2} + \overline{v^2}] \\ & - \frac{1}{e_{1u}e_{2u}e_{3u}} \overline{e_{1T}e_{2T}w^{i+1/2}} \delta_{k+1/2}[u]^k \\ & - \frac{1}{\rho_0 e_{1u}} \delta_{i+1/2} [p^h] - \left( M_u + \frac{1}{H_u e_{2u}} \delta_j \left( \frac{\partial \Psi}{\partial t} \right) \right) \\ & + D_u^{lU} + D_u^{vU} \end{aligned} \quad (2.45)$$

$$\begin{aligned} \frac{\partial v}{\partial t} = & + \frac{1}{e_{2v}} \overline{(\zeta + f)^i} \overline{(e_{2u}u)^{i,j+1/2}} - \frac{1}{2e_{2v}} \delta_{j+1/2} [\overline{u^2} + \overline{v^2}] \\ & - \frac{1}{e_{1v}e_{2v}e_{3v}} \overline{e_{1T}e_{2T}w^{j+1/2}} \delta_{k+1/2}[v]^k \\ & - \frac{1}{\rho_0 e_{2v}} \delta_{j+1/2} [p^h] - \left( M_v + \frac{1}{H_v e_{1v}} \delta_i \left( \frac{\partial \Psi}{\partial t} \right) \right) \\ & + D_v^{lU} + D_v^{vU} \end{aligned} \quad (2.46)$$

La forme semi-discrète, en espace, des équations de conservation de la chaleur et de la salinité sont les suivantes :

$$\frac{\partial T}{\partial t} = -\frac{1}{e_{1T}e_{2T}} \left( \delta_i [e_{2u}u\bar{T}] + \delta_j [e_{1v}v\bar{T}] \right) - \frac{1}{e_{3T}} \delta_k [w\bar{T}] + D_T^{lT} + D_T^{vT} \quad (2.47)$$

$$\frac{\partial S}{\partial t} = -\frac{1}{e_{1S}e_{2S}} \left( \delta_i [e_{2u}u\bar{S}] + \delta_j [e_{1v}v\bar{S}] \right) - \frac{1}{e_{3S}} \delta_k [w\bar{S}] + D_S^{lS} + D_S^{vS} \quad (2.48)$$

<sup>3</sup>Plus exactement dans l'algorithme énoncé au début de la subroutine solpcg, du fichier solpcg.F, qui contient l'algorithme du gradient conjugué préconditionné



avec  $e_{2u} = e_{2u_{i-1/2,j,k}}$ , même chose pour  $e_{2v}$  et  $u = u_{i-1/2,j,k}$ , même chose pour  $v$  et  $w$ .

Le schéma temporel utilisé pour les termes d'advection est le schéma centré, d'ordre deux, saute-mouton ou "leapfrog". Ainsi pour un traceur  $T$  advecté avec la vitesse  $U$  on a :

$$T^{t+\Delta t} = T^{t-\Delta t} + 2\Delta t F[T(t), U(t)] \quad (2.49)$$

Pour assurer la stabilité d'un tel schéma on utilise un terme de diffusion temporelle sous la forme d'un filtre d'Asselin :

$$T^{t+\Delta t} = \alpha T^{t+\Delta t} + (1 - \alpha) T^t \quad (2.50)$$

Ce schéma spatial et temporel est celui utilisé pour le schéma d'Arakawa. Il sera vu plus en détail par la suite dans la partie 'NNPZDDOM'.

Le calcul de la diffusion horizontale est basée sur le schéma temporel avant centré d'ordre 2 suivant :

$$T^{t+\Delta t} = T^{t-\Delta t} + 2\Delta t D^{t-\Delta t} \quad (2.51)$$

Le calcul de la diffusion verticale est basée sur le schéma temporel arrière centré d'ordre 2 suivant :

$$T^{t+\Delta t} = T^{t-\Delta t} + 2\Delta t D^{t+\Delta t} \quad (2.52)$$

## 2.6 Critères de stabilité

- Pour les termes d'advection, le critère de stabilité à respecter est le suivant (voir la partie 'code NNPZDDOM' pour plus de détails) :

$$\left( \frac{|u| + |v|}{\Delta x} + \frac{|v|}{\min(\Delta z)} \right) 2\Delta t \leq 1$$

Un message d'erreur apparaît dans le fichier de sortie si ce critère n'est pas respecté.

- Pour les termes de diffusion horizontale, on doit avoir [15] :

$$|D^l| \leq \frac{(\Delta x)^4}{64\Delta t}$$

soit  $|D^l| \leq 5,625E + 10$  or dans notre cas  $|D^l| = 3,5E + 10$  donc ce critère est respecté

- Les termes de diffusion verticale sont inconditionnellement stable [15]
- Le terme de Coriolis (f) est stable pour  $\Delta t \leq 1h56$  et dans notre cas  $\Delta t = 360$  secondes.

## 2.7 Fichiers associés

Le code OPA8.1 a été écrit en Fortran 77, compatible Fortran 90, sauf les fichiers du répertoire ioipsl qui créent des sorties au format netcdf. Ceux-ci sont écrit

en Fortran 90. Ce programme comprend **20000 à 30000 lignes de codes** réparties en **9 fichiers ayant l'extension .f** - fichiers source Fortran -, **103 fichiers .F** - fichiers avant l'appel au précompilateur cpp - et **77 fichiers .h** - fichiers qui sont inclus dans les fichiers source par le précompilateur cpp. Le précompilateur cpp (standard sous UNIX) est utilisé non seulement pour inclure les fichiers d'extension .h dans des fichiers d'extension.F, mais aussi pour compiler ou non certaines parties des codes sources. En effet, l'utilisateur peut choisir différentes clés qui vont déterminer le contenu de certains fichiers d'extension .f.

## 2.8 CRAY SV1

Les simulations avec le code OPA sont effectuées sur le CRAY SV1, calculateur destiné au calcul scientifique vectoriel. Voici ses principales caractéristiques :

- Nombre de processeurs : 8 (Fréquence = 300 Mhz)
- Architecture : Vectorielle 64 bits
- Capacité mémoire centrale : 4 gigaOctets
- Capacité disque : 95 gigaOctets
- Unités de calcul par processeur : 2
- Puissance crête par processeur : 1,2 gigaFlops

L'étude de la stabilité numérique du code ORCA utilisait un CRAY J90 dont la capacité mémoire était quatre fois inférieure au CRAY SV1 et dont la puissance crête par processeur était six fois plus lente.

# Chapitre 3

## Le code NNPZDDOM

Le code OPA peut être couplé à un code de biologie marine NNPZDDOM qui calcule l'évolution de la concentration des variables suivantes :

- N pour Amonium (NH<sub>4</sub>)
- N pour Nitrate (NO<sub>3</sub>)
- P pour Phytoplancton (PHY)
- Z pour Zooplancton (ZOO)
- D pour Detritus (DET)
- DOM pour Dissolved Organic Matter (DOM)

Ces variables, ou traceurs, sont dites passives car elles sont transportées par la circulation océanique mais ne l'affecte pas, contrairement aux traceurs dits actifs, tels que la température et la salinité, qui eux agissent sur la dynamique du modèle.

Plusieurs schémas numériques peuvent être utilisé pour résoudre l'équations d'advection de ces traceurs passifs. Nous envisagerons dans cette étude le schéma d'Arakawa (qui est déjà utilisé pour les traceurs actifs) et le schéma de Smolarkiewicz dont le principal avantage est de conserver la positivité des traceurs, contrairement au schéma d'Arakawa.

**Fichiers associés** Le code NNPZDDOM est couplé au code OPA en y ajoutant les fichiers correspondant et en activant la clé 'key\_passivetr' pour indiquer qu'on utilise des traceurs passifs et la clé 'key\_trc\_nnpzddom' pour indiquer qu'on utilise le code NNPZDDOM. Le code NNPZDDOM comprend 28 fichiers .F et 14 fichiers .h.

Le schéma de Smolarkiewicz est pris par défaut <sup>1</sup>.

Pour utiliser le schéma d'Arakawa, il faut en plus ajouter la clé 'key\_trc\_arakawa'.

---

<sup>1</sup>Dans ce cas, voici la totalité des clés utilisées pour cette configuration : 'key\_monotasking', 'key\_linear', 'key\_temtda', 'key\_trahdfbilap', 'key\_dynhdfbilap', 'key\_hpgimplicit', 'key\_islands', 'key\_passivetr', 'key\_trc\_nnpzddom', 'key\_trc\_diabio', 'key\_trc\_diaadd', 'key\_trc\_diatrd', 'key\_fluxsr', 'key\_zdftke' et 'key\_diainstant'.

### 3.1 Le schéma d'Arakawa

Dans le repère local (i, j, k) défini précédemment, l'équation d'advection d'un traceur passif T par un champ de vitesse U s'écrit :

$$\frac{\partial T}{\partial t} = -\frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 T u)}{\partial i} + \frac{\partial(e_1 T v)}{\partial j} \right] - \frac{1}{e_3} \frac{\partial(T w)}{\partial k} \quad (3.1)$$

soit en discrétisant en espace :

$$\frac{\partial T}{\partial t} = -\frac{1}{e_{1T} e_{2T}} \left( \delta_i [e_{2u} u \bar{T}] + \delta_j [e_{1v} v \bar{T}] \right) - \frac{1}{e_{3T}} \delta_k [w \bar{T}] \quad (3.2)$$

avec  $e_{2u} = e_{2u_{i-1/2,j,k}}$ , même chose pour  $e_{2v}$  et  $u = u_{i-1/2,j,k}$ , même chose pour v et w.

Nous rappelons que le schéma temporel utilisé est le schéma saute mouton ou "leapfrog" (1.38).

Avec les notations vu en (1.24) et (1.25), on a :

$$\begin{aligned} \delta_i [\bar{T}] &= \bar{T} \left( i + \frac{1}{2} \right) - \bar{T} \left( i - \frac{1}{2} \right) \\ &= \frac{1}{2} [T(i+1) + T(i)] - \frac{1}{2} [T(i) + T(i-1)] \\ &= \frac{1}{2} [T(i+1) - T(i-1)] \end{aligned}$$

Le schéma discrétisé possède donc la forme suivante :

$$\begin{aligned} \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^{n-1}}{2\Delta t} &= -\frac{1}{e_{1T} e_{2T}} \left[ \frac{1}{2} (e_{2u_{i,j,k}} u_{i,j,k} T_{i+1,j,k}^n - e_{2u_{i-1,j,k}} u_{i-1,j,k} T_{i-1,j,k}^n) \right. \\ &\quad \left. + \frac{1}{2} (e_{2v_{i,j,k}} v_{i,j,k} T_{i,j+1,k}^n - e_{2v_{i-1,j,k}} v_{i-1,j,k} T_{i,j-1,k}^n) \right] \\ &\quad - \frac{1}{2e_{3T}} (v_{i,j,k} T_{i,j,k+1}^n - v_{i-1,j,k} T_{i,j,k-1}^n) \end{aligned} \quad (3.3)$$

avec  $e_{1T} = e_{1T_{i,j,k}}$ ,  $e_{2T} = e_{2T_{i,j,k}}$  et  $e_{3T} = e_{3T_{i,j,k}}$

Pour notre domaine, une maille fait 6km en largeur sur 6km en longueur et en profondeur de 10 à 300m. Une maille au point u à la même largeur et longueur qu'une maille au point v ou T. Donc on a

$$e_{1v} = e_{2u} = e_{1T} = e_{2T} = 6000m = \Delta x = \Delta y$$

et

$$10m \leq e_{3T} = \Delta z \leq 300m$$

avec  $\Delta z$  fonction de z.

L'équation (3.3) devient donc

$$\begin{aligned} & \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^{n-1}}{2\Delta t} = \\ & - \left[ \frac{(u_{i,j,k}^n T_{i+1,j,k}^n - u_{i-1,j,k}^n T_{i-1,j,k}^n)}{2\Delta x} + \frac{(v_{i,j,k}^n T_{i,j+1,k}^n - v_{i,j,k}^n T_{i,j-1,k}^n)}{2\Delta x} \right] \\ & \quad - \frac{(w_{i,j,k}^n T_{i,j,k+1}^n - w_{i,j,k}^n T_{i,j,k-1}^n)}{2\Delta z} \end{aligned} \quad (3.4)$$

D'après [13], p.22 et 23, le critère de stabilité est ici :

$$|(u+v)\frac{2\Delta t}{\Delta x}\sin(r\Delta x) + w\frac{2\Delta t}{\Delta z}\sin(r\Delta z)| \leq 1$$

soit

$$|(u+v)\frac{2\Delta t}{\Delta x}\sin(r\Delta x)| + |w\frac{2\Delta t}{\Delta z}\sin(r\Delta z)| \leq 1$$

Il faut donc que l'on ait

$$\left( \frac{|\mathbf{u}| + |\mathbf{v}|}{\Delta \mathbf{x}} + \frac{|\mathbf{w}|}{\min(\Delta \mathbf{z})} \right) 2\Delta \mathbf{t} \leq 1$$

avec  $\Delta x = 6000\text{m}$ ,  $\min(\Delta z) = 10\text{m}$  et  $\Delta t = 360$  secondes. Cette condition est communément appelée critère de stabilité de Courant-Friedrichs-Lewy ou de CFL car elle a été trouvée pour la première fois en 1928 par Courant, Friedrich et Lewy.

Ce test a été mis en place dans le programme à chaque pas de temps. Par exemple, pour le soixantième pas de temps on a :

$$\max(u) = 0.87614735389 \text{ m.s}^{-1}$$

$$\max(v) = 0.1707980336 \text{ m.s}^{-1}$$

$$\max(w) = 0.382930849 \text{ E} - 003 \text{ m.s}^{-1}$$

$$\text{Le critère de stabilité vaut : } 0.1532044676 \leq 1$$

S'il n'est pas respecté, on affiche un message d'erreur.

**Erreur de troncature** Le schéma utilisé est d'ordre 2 en temps et en espace donc l'erreur de troncature ou l'erreur absolue de méthode est la suivante :

$$e_m \approx \frac{1}{(n_{itmin})^2} + \frac{1}{(n_x)^2} + \frac{1}{(n_y)^2} + \frac{1}{(n_z)^2}$$

avec  $n_{it}$  le nombre minimum de pas de temps : 60,

$n_x$  le nombre de maille sur la largeur : 28,

$n_y$  le nombre de maille sur la longueur : 84,

$n_z$  le nombre de mailles sur la profondeur : 30,

**L'erreur de méthode**  $e_m \approx 2,8E - 3$

## 3.2 Le schéma de Smolarkiewicz

Le schéma de Smolarkiewicz (ou méthode FCT pour Flux Corrected Transport) utilisé sur un traceur passif  $T$  consiste à advecter une première fois ce traceur avec la vitesse  $U$  au moyen d'un schéma "upstream". Ce schéma possède une forte diffusion implicite. On obtient alors de nouveaux flux de masse et de nouvelles concentrations de traceurs à partir desquels on calcul un flux de masse antidiffusif  $A_m^{(1)}$ . On advecte ensuite les nouvelles concentrations obtenues par un autre schéma 'upstream' avec les flux de masse  $A_m^{(1)}$ , et ainsi de suite. (1) représente le numéro de la phase corrective. Dans notre cas, on utilise une seule phase corrective.

On a toujours la même équation d'advection d'un traceur  $T$  par une vitesse  $U$  :

$$\frac{\partial T}{\partial t} = -\frac{1}{e_1 e_2} \left[ \frac{\partial(e_2 T u)}{\partial i} + \frac{\partial(e_1 T v)}{\partial j} \right] - \frac{1}{e_3} \frac{\partial(T w)}{\partial k} \quad (3.5)$$

Pour la discrétisation spatiale, nous introduisons la notation suivante :

$$q^+ = \frac{1}{2}(q + |q|) \quad (3.6)$$

$$q^- = \frac{1}{2}(q - |q|) \quad (3.7)$$

D'autre part on définit les flux de masses suivants :

$$A_{1u} = e_{2u} e_{3u} u$$

$$A_{2v} = e_{1v} e_{3v} v$$

$$A_{3w} = e_{1w} e_{2w} w$$

L'équation semi-disrète avec le schéma spatial 'upstream' donne :

$$\begin{aligned} \frac{\partial T}{\partial t} = & -\frac{1}{e_{1T} e_{2T} e_{3T}} \left( \delta_i \left[ A_{1u_{i-\frac{1}{2},j,k}}^+ T_{i-\frac{1}{2},j,k} + A_{1u_{i-\frac{1}{2},j,k}}^- T_{i+\frac{1}{2},j,k} \right] \right) \\ & -\frac{1}{e_{1T} e_{2T} e_{3T}} \left( \delta_j \left[ A_{2v_{i,j-\frac{1}{2},k}}^+ T_{i,j-\frac{1}{2},k} + A_{2v_{i,j-\frac{1}{2},k}}^- T_{i,j+\frac{1}{2},k} \right] \right) \\ & -\frac{1}{e_{1T} e_{2T} e_{3T}} \left( \delta_k \left[ A_{3w_{i,j,k-\frac{1}{2}}}^+ T_{i,j,k-\frac{1}{2}} + A_{3w_{i,j,k-\frac{1}{2}}}^- T_{i,j,k+\frac{1}{2}} \right] \right) \end{aligned} \quad (3.8)$$

avec  $e_{lT} = e_{lT_{i,j,k}}$ ,  $l = 1, 2, 3$

Avec ce schéma spatiale, on utilise le schéma temporel centré avant suivant :

$$T^{t+\Delta t} = T^{t-\Delta t} + 2\Delta t F[T(t-\Delta t), U(t)] \quad (3.9)$$

En fait, on calcule dans un premier temps

$$\begin{aligned} T_{i,j,k}^n = & -\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_i \left[ A_{1u_{i-\frac{1}{2},j,k}}^{+n} T_{i-\frac{1}{2},j,k}^{n-1} + A_{1u_{i-\frac{1}{2},j,k}}^{-n} T_{i+\frac{1}{2},j,k}^{n-1} \right] \right) \\ & -\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_j \left[ A_{2v_{i,j-\frac{1}{2},k}}^{+n} T_{i,j-\frac{1}{2},k}^{n-1} + A_{2v_{i,j-\frac{1}{2},k}}^{-n} T_{i,j+\frac{1}{2},k}^{n-1} \right] \right) \end{aligned}$$

$$-\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_k \left[ A_{3w,i,j,k-\frac{1}{2}}^{+n} T_{i,j,k-\frac{1}{2}}^{n-1} + A_{3w,i,j,k-\frac{1}{2}}^{-n} T_{i,j,k+\frac{1}{2}}^{n-1} \right] \right) \quad (3.10)$$

avec  $e_{lT} = e_{lT,i,j,k}^n, l = 1, 2, 3$

Puis on intègre temporellement :

$$T_{i,j,k}^{(1)^n} = T_{i,j,k}^{n-1} + \Delta t T_{i,j,k}^n \quad (3.11)$$

On calcule ensuite les flux de masse antidiffusif avec  $T^{(1)}$  :

$$A_{1u}^{(1)^n} = \frac{1}{\overline{T}_{i+\frac{1}{2},j,k}^{(1)^n}} \left( |A_{1u}| - \Delta t * \frac{(A_{1u})^2}{e_{1u} e_{2u} e_{3u}} \right) * \delta_{i+\frac{1}{2}} [T^{(1)^n}] \quad (3.12)$$

avec  $e_{lu} = e_{lu,i,j,k}^n$  et  $A_{lu} = A_{lu,i,j,k}^n$  pour  $l = 1, 2, 3$

De même pour  $A_{2v}^{(1)}$  et  $A_{2w}^{(1)}$

Enfin, on calcule à nouveau l'advection avec les termes en  $A^{(1)}$  et  $T^{(1)}$  à la place des termes en  $A$ , respectivement en  $T$  :

$$\begin{aligned} T_{i,j,k}^{n+1} = & -\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_i \left[ A_{1u,i-\frac{1}{2},j,k}^{+(1)^n} T_{i-\frac{1}{2},j,k}^{(1)^n} + A_{1u,i-\frac{1}{2},j,k}^{-(1)^n} T_{i+\frac{1}{2},j,k}^{(1)^n} \right] \right) \\ & -\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_j \left[ A_{2v,i,j-\frac{1}{2},k}^{+(1)^n} T_{i,j-\frac{1}{2},k}^{(1)^n} + A_{2v,i,j-\frac{1}{2},k}^{-(1)^n} T_{i,j+\frac{1}{2},k}^{(1)^n} \right] \right) \\ & -\frac{1}{e_{1T}^n e_{2T}^n e_{3T}^n} \left( \delta_k \left[ A_{3w,i,j,k-\frac{1}{2}}^{+(1)^n} T_{i,j,k-\frac{1}{2}}^{(1)^n} + A_{3w,i,j,k-\frac{1}{2}}^{-(1)^n} T_{i,j,k+\frac{1}{2}}^{(1)^n} \right] \right) \quad (3.13) \end{aligned}$$

avec  $e_{lT} = e_{lT,i,j,k}^n, l = 1, 2, 3$

Par manque de temps et d'information, le critère de stabilité et l'erreur de troncature ne seront pas exposés. On suppose toutefois que le critère de stabilité de ce schéma est toujours respecté.

**Diffusion horizontale** Le schéma d'Arakawa et de Smolarkiewicz peuvent être utilisés, ou non, avec un terme de diffusion horizontale. Ici nous travaillons avec un opérateur bilaplacien.

On calcule tout d'abord le laplacien :

$$D_{i,j,k}^n = \frac{1}{e_{1T} * e_{2T}} \left( \delta_{i-1} \left[ \frac{e_{2u}}{e_{1u}} \delta_i(T_{i,j,k}^n) \right] + \delta_{j-1} \left[ \frac{e_{1v}}{e_{2v}} \delta_j(T_{i,j,k}^n) \right] \right) \quad (3.14)$$

on multiplie ensuite le résultat par un coefficient de diffusion  $B_h$

$$D_{i,j,k}^n = D_{i,j,k}^n * B_h \quad (3.15)$$

On calcule enfin le bilaplacien :

$$D_{i,j,k}^n = \frac{1}{e_{1T} * e_{2T}} \left( \delta_{i-1} \left[ \frac{e_{2u}}{e_{1u}} \delta_i(D_{i,j,k}^n) \right] + \delta_{j-1} \left[ \frac{e_{1v}}{e_{2v}} \delta_j(D_{i,j,k}^n) \right] \right) \quad (3.16)$$

avec  $e_{lT} = e_{lT,i,j,k}^n, l = 1, 2, 3$

# Chapitre 4

## La bibliothèque CADNA

La bibliothèque CADNA est utilisée pour étudier la stabilité numérique du code OPA-TRC-NNPZDDOM. Cette bibliothèque, qui applique la méthode CESTAC, est développée par l'équipe CHPV du Laboratoire d'Informatique de Paris VI (LIP6).

### 4.1 Le LIP6

Le Laboratoire d'Informatique de Paris VI (LIP6) est une unité mixte de recherche de l'Université Pierre et Marie Curie et du CNRS (UMR 7606). Il rassemble en une seule entité l'essentiel de la recherche en informatique à l'Université Pierre et Marie Curie. Avec un effectif d'environ 320 personnes (doctorants compris), il est structuré en neuf thèmes scientifiques :

1. **Algorithmique numérique et parallélisme (ANP)**
2. Apprentissage et acquisition de connaissances (APA)
3. Architecture des systèmes intégrés et micro-électronique (ASIM)
4. Calcul formel (CALFOR)
5. Objets et Agents pour Systèmes d'Information et de Simulation (OASIS)
6. Réseaux et performances (RP)
7. Sémantique, preuve et implantation (SPI)
8. Systèmes répartis et coopératifs (SRC)
9. Systèmes d'aide à la décision et à la formation (SYSDEF)

Le thème Algorithmique Numérique et Parallélisme comprend deux équipes :

- **Calcul à Haute Performance et Validation (CHPV)**
- Optimisation et Modélisation

L'équipe Calcul à Haute Performance et Validation (CHPV) travaille sur deux projets :

1. Projet PAN (Parallélisation d'Algorithmes Numériques)



## 2. Projet VAN (Validation des Algorithmes Numériques).

Ce dernier projet consiste à contrôler et valider les logiciels scientifiques, c'est-à-dire à faire en cours d'exécution du programme :

- L'analyse de la propagation des erreurs d'arrondi
- La détection des instabilités numériques
- Le contrôle des tests et des branchements
- L'estimation de la précision de tout résultat de calcul
- L'estimation de l'influence des incertitudes des données sur les résultats fournis

Ce problème est abordé grâce à l'approche stochastique. Celle-ci permet de tenir compte de la compensation des erreurs d'arrondi et de donner une bonne estimation des résultats.

Ce projet vise comme application tout type de calcul scientifique en virgule flottante sur machine séquentielle, parallèle ou vectorielle.

## 4.2 L'arithmétique à virgule flottante

Tout réel  $x \in \mathbb{R}$ , non nul peut s'écrire :

$$x = \epsilon.b^e.m \text{ avec } b \in \mathbb{N}, \epsilon \in \{-1, +1\}, e \in \mathbb{Z} \text{ et } m \in [0, b[$$

où  $b$  est la base,  $\epsilon$  le signe,  $e$  l'exposant et  $m$  la mantisse.

Si on pose  $m \in [1, b[$ , le triplet  $\{\epsilon, e, m\}$  est unique. La représentation est dite normalisée. C'est cette représentation qui est utilisée dans l'arithmétique à virgule flottante sur ordinateur. Coder un réel sur ordinateur revient donc à coder le triplet  $\{\epsilon, e, m\}$ .

Depuis 1985, une norme internationale est reconnue pour l'arithmétique à virgule flottante et respectée par la quasi-totalité des constructeurs : la norme IEEE754 [16]. Cette norme prévoit, entre autres, les codages simple précision et double précision. Elle utilise la base 2. Pour  $e$  et  $m$ , on utilise donc le développement binaire de ces nombres :

$$e = \sum_{i=0}^p b_i.2^i \text{ et } m = \sum_{i=0}^{\infty} a_i.2^{-i} \text{ avec } (a_i, b_i) \in \{0, 1\}$$

Le codage de  $\epsilon$  tient sur un bit (appelé "bit de signe") et vaut 0 si  $x > 0$  et 1 si  $x < 0$ . Un nombre en double précision codé suivant cette norme possède 52 bits de mantisse plus un bit caché, soit 53 bits de mantisse en tout ce qui correspond à 15 chiffres significatifs exacts au maximum.

Les deux calculateurs CRAY (CRAY J90 et CRAY SV1) ne respectent pas la norme IEEE754 de 1985 [16]. En effet les réels simple précision sont codés sur 64 bits dont 48 bits de mantisse, ce qui correspond à **14 chiffres significatifs exacts au maximum**.

On appelle ensemble des flottants, noté  $\mathbb{F}$ , l'ensemble des réels codables exactement sur ordinateur. Il dépend de la précision utilisée, mais c'est toujours un ensemble fini, donc borné et discret. Le fait qu'il soit borné entraîne les phénomènes

de sous-capacité ou de sur-capacité. Le fait qu'il soit discret et que les opérateurs arithmétiques ne soient pas des lois internes dans  $\mathbb{F}$  entraîne le phénomène d'erreur d'arrondi.

Ainsi tout réel qui n'est pas un flottant est approché par un élément de  $\mathbb{F}$ . Plus précisément, soit  $X_{min}$  (resp.  $X_{max}$ ) le plus petit (resp. le plus grand) nombre flottant, pour tout réel  $x$  dans l'intervalle  $(X_{min}, X_{max})$ , il existe  $\{X^-, X^+\}$  dans  $\mathbb{F}$  tels que  $X^- < X^+$  et  $(X^-, X^+) \cap \mathbb{F} = \emptyset$ .  $x$  est toujours représenté soit par  $X^-$  soit par  $X^+$ .

Déterminer la règle qui, en fonction de  $x$ , fournit  $X^-$  ou  $X^+$ , c'est choisir le mode d'arrondi de la machine. Il y a quatre types d'arrondi :

- l'arrondi vers 0 :  $x$  est représenté par le flottant le plus près de  $x$ , compris entre 0 et  $x$
- l'arrondi au plus près :  $x$  est représenté par le flottant le plus près de  $x$  pour la distance usuelle de  $\mathbb{R}$
- l'arrondi vers plus l'infini :  $x$  est systématiquement représenté par  $X^+$
- l'arrondi vers moins l'infini :  $x$  est systématiquement représenté par  $X^-$

L'arrondi s'effectue d'une part, à chaque lecture d'une donnée ou affectation d'une variable par une valeur numérique explicitée et, d'autre part lors de l'exécution de toute opération arithmétique élémentaire à virgule flottante (addition, soustraction, multiplication et division).

### La propagation des erreurs d'arrondi

Nous avons vu que tout réel  $x \in \mathbb{R}$ , non nul peut s'écrire :

$$x = \epsilon.b^e.m \quad \text{avec} \quad \frac{1}{b} \leq m < 1 \quad (4.1)$$

où  $b$  est la base,  $\epsilon$  le signe de  $x$ ,  $e$  l'exposant et  $m$  la mantisse.

\* Erreur due à l'opération d'affectation

Si  $X$  est la représentation de  $x$  sur un ordinateur qui utilise la base 2, on peut écrire :

$$X = \epsilon.M.2^E \quad \text{avec} \quad X = x - \epsilon.2^{E-p}.\alpha \quad (4.2)$$

$2^{E-p}.\alpha$  représente l'erreur absolue faite sur la mantisse finie  $M$ ,  $p$  le nombre de bits de cette mantisse (le bit caché compris) et  $E$  l'exposant du résultat informatique  $X$ . Suivant le mode d'arrondi considéré on a :

- en arrondi au plus près,  $\alpha \in [-0.5, +0.5[$ ;
- en arrondi vers zéro,  $\alpha \in [0, 1[$ ;
- en arrondi vers plus ou moins l'infini,  $\alpha \in [-1, +1[$ ;

\* erreur due à l'addition informatique

Soient  $\oplus$  l'addition informatique et  $x_1, x_2$  deux réels d'images informatique  $X_1$  et  $X_2$ , on a

$$X_i = x_i - \epsilon_i.2^{E_i-p}.\alpha_i \quad \text{pour} \quad i = 1, 2 \quad (4.3)$$

et

$$X_1 \oplus X_2 = x_1 + x_2 - \epsilon_1.2^{E_1-p}.\alpha_1 - \epsilon_2.2^{E_2-p}.\alpha_2 - \epsilon_3.2^{E_3-p}.\alpha_3 \quad (4.4)$$

$E_3$  étant l'exposant,  $\epsilon_3$  le signe et  $\alpha_3$  l'arrondi du résultat informatique  $X_1 \oplus X_2$

\* erreur due à la soustraction informatique

De façon similaire, en notant  $\ominus$  la soustraction informatique, on obtient

$$X_1 \ominus X_2 = x_1 - x_2 - \epsilon_1.2^{E_1-p}.\alpha_1 + \epsilon_2.2^{E_2-p}.\alpha_2 + \epsilon_3.2^{E_3-p}.\alpha_3 \quad (4.5)$$

\* erreur due à la multiplication informatique

De même, soit  $\otimes$  la multiplication informatique,

$$\begin{aligned} X_1 \otimes X_2 &= x_1.x_2 - \epsilon_1.2^{E_1-p}.\alpha_1.x_2 - \epsilon_2.2^{E_2-p}.\alpha_2.x_1 \\ &+ \epsilon_1.\epsilon_2.2^{E_1+E_2-2p}.\alpha_1.\alpha_2 - \epsilon_3.2^{E_3-p}.\alpha_3 \end{aligned} \quad (4.6)$$

Le quatrième terme en  $2^{-2p}$  est du deuxième ordre. En le négligeant, on obtient :

$$X_1 \otimes X_2 = x_1.x_2 - \epsilon_1.2^{E_1-p}.\alpha_1.x_2 - \epsilon_2.2^{E_2-p}.\alpha_2.x_1 - \epsilon_3.2^{E_3-p}.\alpha_3$$

\* erreur due à la division informatique

Soit  $\oslash$  la division informatique, toujours en négligeant les termes d'ordre supérieur ou égal à deux :

$$X_1 \oslash X_2 = \frac{x_1}{x_2} - \epsilon_1.2^{E_1-p}.\frac{\alpha_1}{x_2} + \epsilon_2.2^{E_2-p}.\alpha_2.\frac{x_1}{x_2^2} - \epsilon_3.2^{E_3-p}.\alpha_3 \quad (4.7)$$

**Phénomène de cancellation** Dans le cas de l'addition de termes de même signes, l'exposant  $E_3$  du résultat est égal à  $Sup(E_1, E_2) + \delta$ ,  $\delta$  valant 0 ou 1. Les deux termes résultant des erreurs d'arrondi sur  $X_1$  et  $X_2$  sont d'un ordre de grandeur inférieur ou égal à  $2^{E_3-p}$ . L'erreur relative sur  $X_1 \oplus X_2$  reste de l'ordre de  $2^{-p}$ . Cette opération est donc "relativement stable" en ce sens où elle ne génère pas de perte brutale de précision.

Pour la multiplication, puisque  $E_3 = E_1 + E_2 + \delta$ ,  $\delta$  valant 0 ou -1 et que pour la division,  $E_3 = E_1 - E_2 + \delta$ ,  $\delta$  valant 0 ou +1, les conclusions sont les mêmes.

Par contre, dans le cas de la soustraction de termes de même signe, on a seulement  $E_3 = Sup(E_1, E_2) - k$ . Si  $X_1$  est très voisin de  $X_2$ ,  $k$  peut valoir plusieurs unités alors que l'erreur absolue reste de l'ordre de  $2^{Sup(E_1, E_2)-p}$  du fait de  $\alpha_1$  et  $\alpha_2$ . L'erreur relative sur  $X_1 \ominus X_2$ ,  $2^{Sup(E_1, E_2)-p-E_3}$ , se trouve multipliée par  $2^k$ . En une opération on a perdu  $k$  bits significatifs exacts. Nous retrouvons là l'instabilité de la soustraction, appelé **phénomène de cancellation**, mise en évidence dans le code ORCA [14].

Un programme de calcul est une suite ordonnée d'opérations arithmétiques. Supposons pour simplifier, qu'il fournisse un résultat unique au bout de  $n$  opérations. Soit  $r \in \mathbb{R}$  le résultat exact, l'ordinateur fournit un résultat  $R \in \mathbb{F}$  entaché d'erreur. Il a été montré [17] que  $R$  peut se modéliser au premier ordre en  $2^{-p}$  par :

$$R \approx r + \sum_{i=1}^n g_i(d).2^{-p}.\alpha_i \quad (4.8)$$

$g_i(d)$  étant des coefficients ne dépendant que des données et  $\alpha_i$  étant des quantités perdues lors des arrondis. De plus, on a supposé que les exposants et signes des résultats intermédiaires étaient indépendants des  $\alpha_i$ . Il est alors aisé de formaliser le nombre de chiffres significatifs exacts  $C_R$  du résultat informatique R. En effet :

$$C_R \approx \log_{10} \left| \frac{r}{R-r} \right| \quad (4.9)$$

### 4.3 La méthode CESTAC

La méthode CESTAC (Contrôle et Estimation Stochastique des Arrondis de Calculs) a été définie par M. La Porte et J. Vignes en 1974, puis généralisée par ce dernier [18, 19, 20].

On peut résumer très simplement l'idée majeure sur laquelle repose la méthode CESTAC. Elle consiste à exécuter plusieurs fois le même programme de calcul en propageant différemment les erreurs d'arrondi. On obtient ainsi pour un même calcul, des résultats différents. La partie commune à tous les résultats représente la partie fiable, l'autre étant la partie non significative.

**L'arithmétique aléatoire** Comme il a été vu précédemment, tout résultat R d'une opération arithmétique est encadré par deux flottants consécutifs  $R^-$  et  $R^+$ . Déterminer le mode d'arrondi, c'est d'établir la règle qui, en fonction de R, choisit de le représenter soit par  $R^-$  soit par  $R^+$ . L'arithmétique aléatoire consiste à retenir aléatoirement  $R^-$  et  $R^+$  avec la même probabilité.

Avec ce mode d'arrondi,  $\alpha_i \in ]-1, +1[$ . Bien sûr ce nouveau mode d'arrondi nécessite un générateur aléatoire. Ainsi, en utilisant l'arithmétique aléatoire, un même programme supposé fournir un résultat unique, exécuté N fois, fournira N valeurs différentes du résultat.

Le but de l'arithmétique aléatoire n'est donc pas d'améliorer la précision du résultat mais seulement de faire propager différemment les erreurs d'arrondi pour ensuite, à partir de N échantillons du résultats, pouvoir estimer son nombre de chiffres significatifs exacts.

**Estimation de la précision** En exécutant N fois le programme de calcul, on obtient donc N représentants  $R_{i,i=1,N}$  de la variable aléatoire R modélisée par (4.8) où les  $\alpha_i$  sont des variables aléatoires indépendantes et équidistribuées. La distribution commune des  $\alpha_i$  est uniforme sur  $[-1, +1]$  donc centrée. Les deux conséquences majeures sont :

1. l'espérance mathématique de la variable R est le résultat mathématique exact r
2. la distribution de R est quasi-gaussienne.

Il s'agit donc d'estimer la moyenne d'une variable aléatoire gaussienne à partir d'un échantillon. C'est exactement le but du test de Student. Celui-ci fournit un intervalle de confiance pour l'espérance d'une gaussienne à partir d'un échantillon sous une probabilité donnée. Appliqué à notre problème, on sait donc que :  $\forall \beta \in$

$[0, 1], \exists \tau_\beta \in \mathbb{R}$ ,  $\tau_\beta$  étant le seuil de confiance de la distribution de Student pour une probabilité de  $(1 - \beta)$ ,

$$P\left(|\bar{R} - r| \leq \frac{s \cdot \tau_\beta}{\sqrt{N}}\right) = \beta \quad (4.10)$$

avec  $\bar{R}$ , la moyenne des  $R_i$  :

$$\bar{R} = \frac{1}{N} \sum_{i=1}^N R_i \quad (4.11)$$

et s l'écart-type des  $R_i$ ,

$$s^2 = \frac{1}{N-1} \sum_{i=1}^N (R_i - \bar{R})^2 \quad (4.12)$$

Sous une probabilité  $\beta$ , le nombre de chiffres significatifs exacts de  $\bar{R}$ , c'est-à-dire le nombre de chiffres décimaux significatifs communs à  $\bar{R}$  et à  $r$ , est minoré par :

$$C_{\bar{R}} = \log_{10} \left( \frac{\sqrt{N} \cdot |R|}{s \cdot \tau_\beta} \right) \quad (4.13)$$

Ainsi, l'application synchrone<sup>1</sup> de la méthode CESTAC à un programme consiste, à chaque étape intermédiaire de calcul, à :

1. effectuer N fois le calcul en parallèle en utilisant le mode d'arrondi aléatoire pour obtenir N résultats intermédiaires  $R_i$  différents
2. choisir la moyenne  $\bar{R}$  de cette population comme représentant informatique du résultat intermédiaire
3. calculer le nombre de chiffres significatifs exacts de  $\bar{R}$  en utilisant la formule (4.13)
4. recommencer l'étape 1 pour le calcul suivant avec les N représentants  $R_i$

En pratique,  $N = 2$ , ou 3 et  $\beta = 0,95$ .

Pour  $N = 2$ ,  $\tau_\beta = 12,706$ .

Pour  $N = 3$ ,  $\tau_\beta = 4,303$ .

**Le zéro informatique** Le concept de zéro informatique [21] est essentiel dans la méthode CESTAC.

*Définition : Un résultat informatique  $R$  est un zéro informatique si  $R=0$  en étant significatif ou bien si  $R$  est quelconque mais non significatif.*

Concrètement, avec la méthode CESTAC, un résultat  $R$ , représenté par N résultats  $R_i$ , sera un zéro informatique si l'une des deux conditions suivantes est remplie :

1.  $\forall i, R_i = 0$
2.  $C_{\bar{R}} \leq 0$

---

<sup>1</sup>L'application asynchrone consiste à exécuter N fois le programme complet avec à chaque fois un résultat final  $R_i$ . On calcule alors la moyenne et le nombre de chiffres significatifs exacts des différents résultats  $R_i$ .

En pratique, l'ordinateur ne peut distinguer le zéro informatique du zéro mathématique. Avec ce nouveau concept, une arithmétique stochastique a été développée avec de nouvelles définitions pour les relations d'ordre et d'égalité, qui prennent en compte la qualité numérique des opérandes.

**Contrôle et validation de la méthode CESTAC [12]** L'étude théorique [17, 22, 23, 24] a montré la validité de la méthode CESTAC sur la modélisation mathématique. Sa validité pratique repose donc entièrement sur la réalité physique ou non des hypothèses et des approximations faites dans l'étude théorique.

Très précisément, ces hypothèses sont :

- a) Les signes et exposants des résultats intermédiaires sont indépendants des perturbations.
- b) Les erreurs d'arrondi  $\alpha_i$  se comportent comme des variables aléatoires indépendantes équidistribuées.
- c) L'approximation au premier ordre en  $2^{-p}$  dans l'expression de R est valide.
- d) L'approximation aux quatre premiers termes de son développement de Edgeworth de la distribution de R dans l'étude du test de Student sur cette distribution est valide.
- e) La régularité des coefficients  $g_i(d)$  est respectée. Ceci signifie qu'aucun d'entre eux n'est prépondérant devant la somme des autres en valeur absolue.

Dans la pratique, les hypothèses a), d) et e) ont peu d'influence sur l'efficacité de la méthode [12].

Les hypothèses b) et c) sont des hypothèses beaucoup plus importantes qui concernent les fondements même de la méthode CESTAC. Ces hypothèses sont des approximations. Par définition, elles ne sont jamais strictement vérifiées dans la réalité.

La conséquence majeure de l'hypothèse b) est la distribution uniforme sur  $[-1, +1[$  des variables aléatoires  $z_i$ , ce qui entraîne que la variable aléatoire Z est centrée sur le résultat mathématique exact. Parce qu'il s'agit d'un modèle, dans la pratique il existe un biais, que nous appellerons de type I, dû à l'approximation faite dans l'hypothèse b). On montre dans [25] qu'un biais de l'ordre de quelques  $s$ -écart-type de R- entraîne une erreur inférieure à un chiffre décimal, voire à un bit, sur l'estimation du nombre de chiffres significatifs exacts. Cette robustesse s'explique principalement par la structure même de la formule  $C_{\overline{R}}$ -utilisation du logarithme- et par la robustesse naturelle du test de Student. La question est de savoir si la différence entre le modèle théorique et la réalité physique due à l'hypothèse b) est susceptible de générer un biais supérieur à quelques dizaines ou quelques centaines de  $s$ .

L'écart-type de R est donné par  $s^2 = \frac{1}{N-1} \sum_{i=1}^N (g_i(d))^2$  et le biais réel est majoré par  $\sum_{i=1}^N |g_i(d)|$ . Il faut donc beaucoup d'opérations avec des coefficients qui soient tous du même ordre de grandeur et un comportement des erreurs d'arrondi systématiquement biaisé pour voir apparaître un rapport important entre l'écart-type théorique et le biais réel. Or, comme nous l'avons rappelé dans le paragraphe précédent, l'expérience a toujours montré que la compensation des erreurs d'arrondi est un phénomène important lors de calculs scientifiques longs. Dans son principe même, la méthode CESTAC, parce qu'elle injecte de l'aléatoire pur dans le calcul,

valide d'autant la modélisation des erreurs des erreurs d'arrondis par des variables aléatoires. Il y a une autovalidation de la méthode CESTAC.

En conclusion, on peut dire que soit il y a peu de calcul et le biais est toujours de l'ordre de quelques  $s$  au plus, soit le phénomène de compensation de l'erreur d'arrondi en arithmétique virgule flottante, amplifié par la méthode CESTAC, joue pleinement. Ceci explique pourquoi dans la pratique, le biais de type I ne met jamais en défaut le nombre de chiffres significatifs exacts fourni par CESTAC.

L'hypothèse c) est la seule hypothèse qui, si elle n'est pas satisfaite, peut réellement faire échouer la méthode CESTAC. L'hypothèse c), rappelons-le, stipule que les termes d'ordre supérieur ou égal à  $2^{-2p}$  sont négligeables devant ceux du premier ordre. Si tel n'est pas le cas, on ne peut plus affirmer que le test de Student est applicable à la distribution de  $Z$  et surtout, l'espérance mathématique des termes d'ordre supérieur ou égal à  $2^{-2p}$  n'étant pas nulle, il en résulte un biais que nous appellerons de type II avec un ordre de grandeur devant lequel l'écart-type de  $Z$  peut-être négligeable. Ceci est illustré par l'exemple de la suite de J.-M. Muller [26] où CESTAC surestime le nombre de chiffres significatifs.

Pour éviter cette surestimation possible et utiliser correctement la méthode CESTAC, il faut analyser ce qui peut invalider l'approximation au premier ordre. Les additions et soustractions ne génèrent pas de nouveaux termes d'ordre supérieur à 2 [17, 24]. Une déviation brutale de la trajectoire mathématique ne peut pas venir de ces opérations (nous supposons, pour simplifier, que le calcul ne fait intervenir que les quatre opérations arithmétiques élémentaires). Par contre, les multiplications et les divisions génèrent des termes d'ordre supérieur (voir le paragraphe 'la propagation des erreurs d'arrondi').

Pour la multiplication, le produit  $\epsilon_1 \cdot \epsilon_2$  devient prépondérant devant les termes du premier ordre si, à la fois,  $\epsilon_1$  et  $\epsilon_2$  sont prépondérants devant  $x_1$  et  $x_2$ . Il s'agit de la multiplication de deux résultats non significatifs. Remarquons que dès que l'une des deux opérands est significative, l'approximation au premier ordre pour la multiplication reste valable.

Dans le cas de la division de  $X_1$  par  $X_2$ , les termes du résultat d'ordre supérieur à 2 seront prépondérant si et seulement si  $\epsilon_2$  est prépondérant devant  $x_1$ , ie, si  $X_2$  est non significatif. La division par un résultat non significatif, de même que précédemment est susceptible de dévier totalement la trajectoire informatique de la trajectoire mathématique.

Il faut donc détecter les apparitions éventuelles de ce type d'opérations. Ceci ne pose aucune difficulté à condition d'utiliser la programmation synchrone de la méthode CESTAC qui seule permet d'évaluer à tout instant la précision de tout résultat intermédiaire. On dispose en permanence de  $N$  représentants de tout résultat intermédiaire ce qui permet d'en estimer la précision. On peut dès lors tester systématiquement la non signification de deux opérands pour les multiplications et du dénominateur pour les divisions. Il y a là une véritable autovalidation de la méthode CESTAC puisque la méthode est capable de détecter le moment où ses conditions de validité ne sont plus satisfaites.

C'est sur cette capacité que repose toute la fiabilité de la bibliothèque CADNA.

## 4.4 La bibliothèque CADNA

Le logiciel CADNA (Control of Accuracy and Debugging for Numerical Applications) conçu par *Jean VIGNES* et *Jean-Marie CHESNEAUX* [8, 6], tous deux membres de l'équipe Calcul à Haute Performance et Validation (CHPV), permet de valider les résultats de tout programme de calcul scientifique exécuté sur ordinateur. CADNA se présente comme une librairie écrite en Fortran 90, utilisable après la compilation lors de l'édition de liens. Elle met en œuvre automatiquement et de manière synchrone la méthode CESTAC.

Un nouveau type, **le nombre stochastique**, a été créé : il n'est rien d'autre qu'un N-uplet (avec  $N=2$  pour la version utilisée dans notre étude) d'éléments en virgule flottante perturbés. Tous les opérateurs arithmétiques ( $+$ ,  $-$ ,  $*$ ,  $/$ ) ont été surchargés de telle sorte que, lorsqu'un tel opérateur est exécuté, les opérandes sont des N-uplets et le résultat rendu est un N-uplet aléatoirement perturbé. Les opérateurs dits de relation ( $>$ ,  $\geq$ ,  $<$ ,  $\leq$ ,  $=$ ) ont été également surchargés en respectant les propriétés de relation d'ordre [6, 9]. Toutes les fonctions utilisables dans des programmes écrits en Fortran 77 (SIN, COS, EXP,...) ont été également surchargées. De même, les instructions d'entrées-sorties ont été modifiées, notamment l'instruction d'impression qui fournit comme résultat la moyenne du N-uplet écrite avec uniquement son nombre de chiffres décimaux significatifs exacts. De plus, afin de pouvoir évaluer l'influence des erreurs de données sur les résultats de programme de calcul scientifique, une fonction permet de perturber les données [6, 9].

En outre, CADNA détecte les éventuelles instabilités numériques causées par un zéro informatique. Par exemple, CADNA détecte les divisions instables car elles sont dues à un dénominateur non significatif. Un zéro informatique est affiché par la bibliothèque CADNA sous la forme **@.0**.



# Chapitre 5

## Instrumentation de CADNA dans le code OPA-TRC-NNPZDDOM

### 5.1 Méthode utilisée

On est parti du code ORCA déjà “cadnatisé” [13]. Voici un rappel des principales étapes de l’instrumentation de CADNA dans ORCA :

#### 1) Initialisation

Cette initialisation a pour but :

- d’ouvrir le fichier `cadna_stability_f90.lst` (correspondant à l’unité logique 58) qui contient les messages éventuels d’instabilités numériques
- d’initialiser l’utilisation de l’arithmétique aléatoire.

Cette initialisation est déjà rajoutée à la suite des déclarations du programme principal (*modele.f*). Elle est réalisée par l’appel d’un sous-programme :  
`call CADNA_INIT(K, ind_div, ind_mul, ind_test, ind_intr, ind_math, ind_perte, SEUIL)`

Le premier argument, `K`, de type entier, permet de limiter la taille du fichier `cadna_stability_f90.lst`.

- Si `K = -1`, le fichier contiendra tous les messages des instabilités détectées.
- Si `K = 0`, le fichier ne contiendra aucun message d’instabilité.
- Si `K = N`, le fichier contiendra les `N` premiers messages.

Les autres arguments sont de types logiques et valent vrai si l’on veut détecter les instabilités correspondantes :

**ind\_div** : pour des divisions par un “zéro informatique”

**ind\_mul** : pour des multiplications où les deux opérandes sont des “zéros informatiques”

**ind\_test** : pour la détection dans les relations d’ordre où la différence entre les deux opérandes est un “zéro informatique”

**ind\_intr** : pour le cas où les fonctions intrinsèques *int*, *aint*, *abs*, *mod*, *sign* génèrent des instabilités

**ind\_math** : pour le cas où les fonctions mathématiques *sqrt*, *exp*, *log*, *log10*, *sin*, *cos*, *tan*, *asin*, *acos*, *atan*, *atan2*, *sinh*, *cosh*, *tanh* génèrent des instabilités.

**ind\_perte** : pour détecter la perte brutale de précision d'une opération arithmétique élémentaire

Le dernier argument, S, de type entier, représente le nombre de chiffres significatifs exacts à partir duquel on considère qu'il y a perte brutale de précision. Sa valeur par défaut est 4.

Dans un premier temps K a été initialisé à 0 pour ne pas allonger le temps d'exécution.

2) Modification des types

On utilise le type défini par cette bibliothèque, TYPE (SINGLE\_ST), qui déclare un stochastique simple précision. Il remplace le type REAL et REAL\*8.

3) Modification des fonctions on-line

CADNA n'accepte pas les fonctions on-line car l'opération d'égalité qui correspond à la définition d'une fonction avec des arguments n'a pas été surdéfinie. Ces fonctions ont donc été réécrites.

4) Modification de certaines fonctions standard

Seules certaines fonctions standard sont reconnues par CADNA. Par exemple CADNA reconnaît la fonction MIN avec deux arguments mais pas avec trois. Il faut donc dans ce cas réécrire l'appel à la fonction min :

$$\text{min}(a, b, c) \Rightarrow \text{min}(\text{min}(a, b), c)$$

5) Modification de certaines fonctions du compilateur Fortran sur le CRAYJ90

Le code OPA utilise des fonctions qui possèdent des arguments réels, propres au compilateur Fortran sur CRAY : *cvmgp*, *cvmgm*, *cvmgz*, *sdot*. Elles n'ont pas été redéfinies dans CADNA avec des arguments stochastiques. Elles ont donc été redéfinies sous un autre nom pour que les fonctions correspondantes du CRAY ne soient pas appelées en premier.

6) Modification des opérateurs vectoriels

la surcharge des opérations sur les tableaux n'a pas encore été définie dans la version actuelle de CADNA. Les opérations sur les tableaux doivent donc être effectuées à l'aide de boucles.

7) Modification des entrées

La fonction de lecture READ ne permet pas de lire directement des variables de type stochastique. Il faut rajouter une variable locale réelle, que l'on va lire, et on affecte ensuite la variable stochastique à cette valeur.

Le fichier *trclec.F*, fichier du code NNPZDDOM va lire des variables dans le fichier *namelist.passivetrc.norst* du répertoire BIOSCRIPTS (en dehors

du répertoire BIOORCA où se trouvent tous les autres fichiers). On a donc changé le nom de la variable correspondante dans le fichier namelist. Ainsi on avait dans *trlec.F*

```
REAL rsc,rtrn,ahtrb0
namelist/natnum/rsc,rtrn,ncor,ndttrc,crosster,lhdf,ahtrb0
READ(numnat,natttrc)
```

et dans le fichier *namelist.passivetric.norst*

```
&NATNUM
  ndttrc=1,
  lhdf=.TRUE.,
  rsc=1.,
  rtrn=1.e-15,
  ncor=1,
  crosster=.FALSE.,
  ahtrb0=0.,
&END
```

Après “cadnatisation”, on a dans *trlec.F*

```
REAL zrsc,zrtrn,zahtrb0
namelist/natnum/zrsc,zrtrn,ncor,ndttrc,crosster,lhdf,zahtrb0
READ(numnat,natttrc)
rsc= zrsc
rtrn=zrtrn
ahtrb0=zahtrb0
```

et dans le fichier *namelist.passivetric.norst*

```
&NATNUM
  ndttrc=1,
  lhdf=.TRUE.,
  zrsc=1.,
  zrtrn=1.e-15,
  ncor=1,
  crosster=.FALSE.,
  zahtrb0=0.,
&END
```

#### 8) Modification des sorties

De façon à fournir les résultats avec leur précision associée, il convient de modifier les ordres d’écriture en utilisant la fonction **STR()** de CADNA. Cette fonction traite un argument de type stochastique et retourne une chaîne de caractère contenant la représentation décimale avec exposant de l’argument. Seuls les **chiffres décimaux significatifs exacts** sont contenus dans la chaîne. La précision est alors immédiatement lisible. Lorsque l’argument est un “zéro informatique”, la chaîne retournée est : **@.0**.

Pour instrumenter CADNA dans les nouvelles sous-routines correspondant au code de biologie marine, on a procédé de deux manières :

1. Si le code de la sous-routine était proche du code initial “cadnatisé”, on a pris la version non-cadnatisée de cette sous-routine et on a rajouté les éléments nouveaux dans en les “cadnatisant” si nécessaire.
2. Si le code de la sous-routine version NNPZDDOM différait beaucoup du code version OPA ou si c’était une nouvelle sous-routine, on a ajouté cette nouvelle (version de) sous-routine puis on l’a entièrement “cadnatisée”.

## 5.2 Erreur globale

La méthode des différences finies, utilisée pour résoudre les équations primitives est une méthode approchée : elle fournit une approximation de la solution exacte. La solution obtenue est donc toujours entachée d’une **erreur de méthode** que l’on appelle  $\mathbf{e}_m$  [6, 9].

Du point de vue informatique, lorsqu’une telle méthode est mise en œuvre sur ordinateur, elle fournit toujours une solution entachée d’une **erreur globale**,  $\mathbf{e}_g$ . Cette erreur résulte de l’erreur de méthode  $e_m$ , inhérente à la méthode mathématique utilisée et de l’erreur due à la propagation des erreurs d’arrondi, appelée ici **erreur de calcul**,  $\mathbf{e}_c$ . Il est bien connu [6, 9] que, lorsque le pas de discrétisation de ces méthodes décroît,  $e_m$  décroît et qu’au contraire, quand il croît,  $e_m$  croît aussi. On appellera ici  $h$  le pas de discrétisation,  $h$  représentant le pas temporel ( $\Delta t$ ) et les pas spatiaux ( $\Delta x, \Delta y = \Delta x$  et  $\Delta z$ ). Il a été montré [30], que lorsque  $h$  croît,  $e_c$  décroît et que, lorsqu’il décroît,  $e_c$  croît. De telle sorte que  $e_m$  et  $e_c$  varient en sens opposé et, de ce fait, l’erreur globale  $e_g$  est une fonction qui présente un minimum en fonction de  $h$ .

Ainsi la meilleure approximation de la solution que l’on peut obtenir sur ordinateur correspond à un pas de discrétisation optimal  $h^*$  tel que  $\frac{de_g}{dh} = 0$ . Cet optimum est obtenu lorsque  $e_m \approx e_c$ .

L’erreur de méthode dans le cas d’une méthode de différences finies, appliquée à schéma numérique, est aussi appelé **erreur de troncature**.

L’erreur de troncature du schéma numérique d’Arakawa est

$$e_m \approx \frac{1}{(n_{itmin})^2} + \frac{1}{(n_x)^2} + \frac{1}{(n_y)^2} + \frac{1}{(n_z)^2}$$

avec  $n_{it}$  le nombre minimum de pas de temps : 60,

$n_x$  le nombre de maille sur la largeur : 28,

$n_y$  le nombre de maille sur la longueur : 84,

$n_z$  le nombre de mailles sur la profondeur : 30,

D’où  $\mathbf{e}_m \approx \mathbf{2,8E} - \mathbf{3}$

L’erreur de calcul  $e_c$  peut être estimée par la bibliothèque CADNA. Le nombre de chiffres significatifs exacts des traceurs sera vu en détail par la suite. Mais l’on peut d’ores et déjà dire que les traceurs biologiques ont une concentration qui possède environ 10 chiffres significatifs exacts (le maximum étant à 14), ce qui est

une très bonne précision. Prenons l'exemple du phytoplancton, au soixantième pas de temps dont la concentration est de l'ordre de  $10^{-2}$  mmole. $m^{-3}$  avec plus de 10 chiffres significatifs exacts. L'erreur de calcul (absolue) est donc de l'ordre de  $10^{-12}$  mmole. $m^{-3}$ . On a donc

$$e_c \ll e_m$$

L'erreur de calcul la plus importante qui ait été observée sur 480 pas de temps affecte les nitrates : leur concentration est de l'ordre de 9,2 mmole. $m^{-3}$  avec au moins 10 chiffres significatifs exacts. Ceci correspond à une erreur de calcul de l'ordre de  $10^{-10}$  mmole. $m^{-3}$ .

Donc on peut affirmer que pour les six concentrations biologiques étudiées, quand on lance au moins 60 pas de temps et jusqu'à 480 (ce qui correspond à deux jours de simulation), l'erreur de calcul est nettement inférieure à l'erreur de méthode.

On peut en conclure que, de ce point de vue, **les calculs ne sont pas effectués avec le pas optimal en espace et en temps. A condition de respecter les conditions de stabilité, les pas pourraient être diminués de façon à diminuer l'erreur de méthode et donc l'erreur globale sur les résultats.** L'allure de la courbe représentant l'erreur de méthode ne sera pas étudiée ici<sup>1</sup>, donc on ne peut calculer les pas optimaux.

### 5.3 Convergence du gradient conjugué préconditionné

Comme il a été vu au chapitre 2, à cause des erreurs d'arrondi, la méthode du gradient conjugué préconditionné est une méthode itérative. On a le test d'arrêt suivant :

$$\frac{\gamma_{k+1}}{\langle b, Qb \rangle} \leq \epsilon \quad (5.1)$$

avec  $\epsilon$  choisi à priori (voir le paragraphe 2.4 pour les notations).

En fait le test réellement utilisé dans le code est le suivant :

$$\gamma_{k+1} \leq \tilde{\epsilon} \quad \text{avec} \quad \tilde{\epsilon} = \epsilon * \langle b, Qb \rangle \quad (5.2)$$

C'est une relation du type  $F(x_k) \leq \tilde{\epsilon}$  avec  $x_k$  la suite générée pour approcher la solution mathématique du problème. Si on choisit une valeur  $\epsilon$  trop grande, le processus itératif peut s'arrêter avant que l'on ait obtenu la meilleure solution numérique possible. Si la valeur est trop petite [30] :

- on peut effectuer un grand nombre d'itérations avant l'arrêt de l'algorithme sans amélioration de la solution ;
- l'algorithme peut boucler ou produire un itéré très loin de la solution cherchée par accumulation d'erreurs de calcul (d'arrondi).

---

<sup>1</sup>principalement par manque de temps

Il faut donc arrêter correctement le processus itératif, c'est-à-dire dès qu'une solution numérique est informatiquement satisfaisante pour le problème.

Ainsi avec CADNA, on regarde à chaque itération si, en notant @.0 le zéro informatique (voir paragraphe 4.3) :

1.  $\|x_k - x_{k-1}\|_\infty = @.0$  <sup>2</sup> ;  
La différence entre un itéré et le suivant ne représente que des erreurs d'arrondi.
2. le nombre moyen de chiffres significatifs de  $x_k$  est  $\leq 1$  ;  
Dans ce cas, les  $x_k$  sont, en moyenne<sup>3</sup>, entachés d'une erreur d'arrondi au moins de l'ordre de leur propre valeur 'informatique' et il est donc inutile de poursuivre le calcul.
3.  $\gamma_{k+1} = @.0$  ;  
Les itérations doivent s'arrêter car la solution obtenue est informatiquement satisfaisante.
4.  $n \geq nmax$ .  
La suite informatique est considérée comme non-convergente et on arrête les itérations.

**C'est seulement dans le cas 3) qu'on a une solution informatiquement satisfaisante.**

D'autre part il faut prendre en compte le fait que CADNA peut surestimer le nombre de chiffres significatifs exacts d'un résultat dont le calcul comprend des multiplications instables.

Il faut également noter que l'on a redonné toute sa précision au  $x_0$  initial utilisé dans cet algorithme. En effet il n'y pas de condition a priori sur ce  $x_0$  et s'il possède un mauvaise précision (ce qui était le cas car il résulte de calculs),  $\gamma_{k+1}$  devient non-significatif avant que l'algorithme n'ait eu le temps de converger.

**Premier appel du gradient conjugué préconditionné** L'algorithme du gradient conjugué se situe dans le fichier *solpcg.F*. La subroutine correspondante, *solpcg*, est appelée la première fois par la subroutine *islbsf* (du fichier *islbsf.F*) qui calcule la fonction de courant barotropique de l'île. On a

$$\tilde{\epsilon} = \textit{epsisl} * \textit{epsisl} * < b, Qb >$$

avec  $\textit{epsisl} = 10^{-10}$ ,  
et  $< b, Qb > = 0.16250091347E - 002$   
soit  $\tilde{\epsilon} = 1.6250091347E - 23$ .

Avec ce test, le gradient conjugué s'arrête à la 144<sup>ème</sup> itération.

Avec les tests d'arrêt, cités au paragraphe précédent, mis en place au moyen de la bibliothèque CADNA, l'algorithme s'arrête à la 116<sup>ème</sup> itération car les deux

---

<sup>2</sup>la norme infinie est utilisée pour ne pas faire d'erreur d'arrondi sur le calcul de la norme (car dans ce cas ce test n'aurait plus de sens)

<sup>3</sup>il est à noter que la moyenne est calculée sur le nombre de chiffres significatifs exacts de  $x_k$  et non sur les valeurs de  $x_k$  elles-mêmes

traces du résidu  $\gamma_{k+1}$  valent 3.885840850491544E-16 et 3.128489322589447E-16 soit @.0 (ce nombre n'est pas significatif).

Cependant en regardant de plus près, c'est-à-dire en regardant les résultats intermédiaires du calcul de  $\gamma_{k+1}$ , on s'aperçoit que la précision de celui-ci est totalement surestimée dès la 20<sup>ème</sup> itération. Nous rappelons que :

$$\gamma_{k+1} = \langle z_{k+1}, Q z_{k+1} \rangle$$

et que  $z_{k+1}$  est un tableau de 28\*84 soit 2352 termes. Donc le calcul de  $\gamma_{k+1}$  est une somme de 2352 multiplications de type  $z_{i,j}^{k+1} * (q_i * z_{i,j}^{k+1})$ , avec  $q_i$ , le i<sup>ème</sup> élément diagonal de Q (Q est diagonale), i variant de 1 à 28 et j de 1 à 84.

Après avoir fait afficher, à chaque itération, la somme des multiplications instables pour voir à partir d'où CADNA surestimait la précision du résultat, on a fait afficher toutes les multiplications instables rentrant dans ce calcul à l'itération 20 car c'est à partir de là que les résultats semblaient être surestimés.

A l'itération 20, pour le calcul de  $z_{28,3}^{21} * (q_{28} * z_{28,3}^{21})$  on a, avec CADNA :

La somme des multiplications calculées jusqu'ici pour ce calcul qui vaut :

3.757569404032225E-9 et 3.428899276920191E-9, avec 1 chiffre significatif exact

L'opérande gauche ( $z_{28,3}^{21}$ ) vaut :

1.93843965120987E-3 et -1.851723521891071E-3, avec **0 chiffre significatif exact**

L'opérande droite ( $q_{28} * z_{28,3}^{21}$ ) vaut :

1.938450547895535E-6 et -1.851733931113237E-6, avec **0 chiffre significatif exact**

et le résultat de cette multiplication instable vaut :

3.757569403950169E-9 et 3.428899276526215E-9, avec 1 chiff. sign. exact au lieu

de 0. Donc on voit nettement que CADNA surestime le résultat de cette multiplication. D'autre part, quand on additionne ce résultat à la somme déjà calculée, on obtient :

7.515138807982394E-9, 6.857798553446433E-9, avec 1 chiff. sign.

Donc on ajoute quelque chose de non significatif de l'ordre de  $10^{-9}$  à quelque chose de significatif de l'ordre de  $10^{-9}$ . Le résultat est donc quelque chose de l'ordre de  $10^{-9}$  **qui n'est pas significatif**. Il en est de même pour tout le calcul des  $z_{i,j}^{21} * (q_i * z_{i,j}^{21})$  suivants pour cette 20<sup>ème</sup> itération.

On a montré que  $\gamma_{k+1}$  était non significatif à la 20<sup>ème</sup> itération.

Donc on est dans le cas d'arrêt 3) vu précédemment. La solution est 'informatiquement satisfaisante'. A l'itération 19,  $\gamma_{k+1}$  est de l'ordre de 7.5E-6, et à l'itération 20 il est de l'ordre de 4.8E-6. Donc pour que l'algorithme s'arrête à cette itération sans CADNA, il faudrait mettre *epsil* à 5,547E-02 de manière à ce que  $\tilde{\epsilon}$  vaille 5E-06.

**Autres appels au gradient conjugué** L'algorithme du gradient conjugué pré-conditionné, donc la subroutine *solpcg*, est aussi appelé par la subroutine *dynspg* du fichier *dynspg.F* (qui utilise *dynspg.rigidlid.h* dans notre configuration). Et dans ce cas on a

$$\tilde{\epsilon} = eps * eps * \langle b, Qb \rangle$$

avec  $\text{eps}=1\text{E-}06$  et  $\langle b, Qb \rangle$  qui varie suivant les appels à la sous-routine *solpcg*. Cet algorithme, appelé par la sous-routine *dynspg*, est utilisé une fois avant le premier pas de temps et ensuite une fois par pas de temps.

Quand on met les tests d'arrêt de CADNA, le gradient conjugué préconditionné, quand il est appelé avant le premier pas de temps, s'arrête avec plus d'itérations qu'avec le test d'arrêt utilisant  $\tilde{\epsilon}$ . Au premier pas de temps on a  $\gamma_{k+1} \approx \tilde{\epsilon}^4$ , donc il effectue le bon nombre d'itération. A partir du deuxième pas de temps, il effectue quasiment toujours plus d'itérations sauf pour certains pas de temps où l'on s'arrête avant la cinquième itération.

Ainsi on a :

- Pour le premier appel avant le premier pas de temps, on s'arrête à la 190<sup>ème</sup> itération avec  $\gamma_{k+1}$  de l'ordre de  $10^{-18}$  alors que  $\tilde{\epsilon}$  est de l'ordre de  $10^{-8}$
- Pour le 1<sup>er</sup> pas de temps, on s'arrête également à la 190<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 2 * 10^{-17}$  et  $\tilde{\epsilon} \approx 2 * 10^{-17}$
- Pour le 2<sup>ème</sup> pas de temps, on s'arrête à la 195<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-18}$  alors que  $\tilde{\epsilon} \approx 10^{-14}$
- Pour le 3<sup>ème</sup> pas de temps, on s'arrête à la 227<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-28}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- Pour le 4<sup>ème</sup> pas de temps, on s'arrête à la 227<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-28}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- Pour le 5<sup>ème</sup> pas de temps, on s'arrête à la 231<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-28}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- ...
- mais au 18<sup>ème</sup> pas de temps, on s'arrête à la 5<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-8}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- Puis on recommence : pour le 19<sup>ème</sup> pas de temps, on s'arrête à la 231<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-31}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- ...
- mais au 22<sup>ème</sup> pas de temps, on s'arrête à la 1<sup>ère</sup> itération avec  $\gamma_{k+1} \approx 10^{-7}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- Puis on recommence : pour le 23<sup>ème</sup> pas de temps, on s'arrête à la 234<sup>ème</sup> itération avec  $\gamma_{k+1} \approx 10^{-31}$  alors que  $\tilde{\epsilon} \approx 10^{-13}$
- ...
- ...

Pour tous ces résultats, nous avons vérifié que CADNA ne surestimait pas le nombre de chiffres significatifs exacts des résultats.

En conclusion, dans la majorité des cas, sans CADNA, le gradient conjugué s'arrête trop tôt, on n'a pas obtenu la meilleure solution numérique, donc *eps* pourrait être plus petit. Mais dans certains cas, le gradient conjugué s'arrête trop tard, donc *eps* devrait être plus grand, ce qui signifie que certaines itérations ont été

---

<sup>4</sup>le choix de *eps* a dû se faire sur ce test d'arrêt



effectuées 'pour rien' (sans amélioration de la solution) ou que la solution obtenue est loin de la solution mathématique par accumulation d'erreurs d'arrondi. On n'est vraisemblablement pas dans ce dernier cas puisque la solution obtenue n'est pas, en moyenne, non-significative.

## 5.4 Précision des différents schémas

**Rappel des résultats obtenus avec ORCA** Dans l'étude de la stabilité numérique du code ORCA [13], la précision des traceurs dynamiques a été étudiée dans la cas d'une configuration 'océan globale' (on prenait tous les océans en compte). Il faut noter également que les données étaient lues dans un fichier *restart* (elles découlaient de simulations déjà effectuées sur une période d'un an par exemple), ce qui n'est pas le cas ici.

**Les variables dynamiques (courant, température, salinité,..) avaient une très bonne précision** : plus de 10 chiffres significatifs exacts pour  $tn$  (température),  $sn$  (salinité),  $rdn$  (densité),  $bn2n$  (fréquence de Brunt-Vaisala), et  $q$  (condition de flux de chaleur à la surface) et plus de 7 chiffres significatifs exacts pour  $un$  (première composante horizontale du courant),  $vn$  (deuxième composante horizontale du courant),  $wn$  (composante verticale du courant),  $rotn$  (la vorticit  relative),  $spgu$  (le gradient horizontal de pression de surface),  $spgv$  (le gradient vertical de pression de surface).

**Les simulations** Dans cette  tude, nous avons regard  la pr cision des principales variables, actives et passives, du code OPA-TRC-NNPZDDOM, sur 480 pas de temps, ce qui correspond   deux jours de simulation. Dans l' tude sur ORCA, les simulations n'allaient pas au-del  de 30 it rations car cela prenait d j  une semaine ! Heureusement, les calculs  taient ici plus rapides et cela est d  au fait que :

- nous utilisons le CRAY SV1 du CCR (Centre de calcul Recherche et R seau de Jussieu) qui est deux fois plus rapide que le CRAY J90 qui  tait utilis  pour  tudier ORCA
- le domaine  tudi  est plus petit et m me si son d coupage est plus fin, au total il comporte quand m me moins de maille donc moins de points   calculer
- les fonctions de CADNA ont  t  "inlin es", cela signifie que lors de l' dition de liens le code des fonctions est recopi  dans le programme source qui les utilise. On gagne donc le temps d'appel   ces fonctions, ce qui n'est pas n gligeable.

Avec notre configuration, **480 it rations ont pris 7h18mn (en temps de calcul) et avec le temps d'attente 12h41mn.**

**Les variables dynamiques** La pr cision des traceurs actifs (ou variables dynamiques), ne d pend pas du sch ma num rique (Arakawa ou Smolarkiewicz) utilis  pour les traceurs passifs (variables biologiques), ce qui est normal puisque, par d finition, les traceurs passifs n'agissent pas sur les traceurs actifs.

Les traceurs actifs ont une bonne pr cision : ils ont au moins huit chiffres significatifs exacts apr s la phase d'initialisation.

Dans la suite on ne tient pas compte du fait que certaines variables sont initialisées avec 14 chiffres significatifs exacts (leurs valeurs d'initialisation ne sont pas calculées).

On peut regrouper les traceurs actifs selon trois allures de courbes pour leur précision :

- Première allure de courbe

Les trois composantes du courant ( $un$ ,  $vn$  et  $wn$ ), la vorticité relative ( $rotn$ ) et la divergence horizontale ( $hdivn$ ) ont une courbe qui commence assez bas (autour de 6 ou 7 chiffres significatifs exacts) puis augmente rapidement et semble se stabiliser autour de : 8,7 chiffres significatifs en moyenne pour  $un$  (soit 1 chiffre de moins qu'avec ORCA), 9,2 pour  $vn$  (soit 0,2 chiffre de moins qu'avec ORCA), 8,3 pour  $wn$  (même précision environ qu'avec ORCA), 8,7 pour  $rotn$  (soit 0,8 chiffre de moins) et 8,5 pour  $hdivn$  (soit 0,2 chiffre de plus) La courbe de la divergence horizontale est toutefois moins régulière que les autres (voir ANNEXE 2,3 et 5).

- Deuxième allure de courbe

La température  $tn$ , la salinité  $sn$ , l'énergie  $en$  et la fréquence de Brunt-Vaisala ( $bn2n$ ) ont une courbe qui commence à 14 chiffres significatifs exacts et qui diminue rapidement dans un premier temps puis de façon de plus en plus progressive et semble tendre vers une valeur : 11,9 chiffres significatifs en moyenne pour  $tn$  (soit 0,2 chiffre de plus qu'avec ORCA) et  $sn$  (soit 1,1 chiffre de plus) , 12,9 (soit 0,6 chiffre de moins) pour  $en$  et 10,6 pour  $bn2n$  (soit 0,2 chiffre de moins) (voir ANNEXE 4 et 6).

On peut noter que la courbe de la salinité est très bruitée pour les 200 premières itérations, que celle de l'énergie possède une certaine irrégularité et que celle de la fréquence de Brunt-Vaisala se stabilise très rapidement.

- Troisième allure de courbe

Le gradient horizontal ( $spgu$ ) et le gradient vertical ( $spgv$ ) de pression de surface ont une allure de courbe qui se rapproche de la première mais elle part d'environ 9 chiffres significatifs exacts, augmente plus doucement et de façon plus bruitée. Ces courbes semblent tendre respectivement vers 10 (soit 2,3 chiffres de plus qu'avec ORCA) et 9,5 chiffres significatifs exacts (soit 1,7 chiffre de plus) (voir ANNEXE 8).

La densité ( $rdn$ ) possède une courbe qui se rapproche de celle de la température, avec une différence majeure cependant, elle ne semble pas converger vers une valeur mais continue toujours de descendre au 480<sup>ème</sup> pas de temps (voir ANNEXE 6) alors qu'avec ORCA c'était la variable la plus régulière : elle avait environ 11,9 chiffres significatifs exacts.

Quant au nombre de chiffres significatifs des radiations solaires ( $q$ ) et des conditions de flux de chaleur à la surface ( $qsr$ ), il reste à 14 (voir ANNEXE 7), ces variables ne sont pas calculées.

**Précision des variables biologiques avec Arakawa** Toutes les concentrations, sauf mention contraire, sont exprimés en  $\text{mmole.m}^{-3}$ .

De même que l'on a regroupé les courbes pour les traceurs actifs, on peut regrouper les courbes des traceurs passifs (variables biologiques) en trois catégories.

- Première allure de courbe

Mis à part les nitrates, toutes les courbes de précision des concentrations des traceurs passifs ont exactement la même allure. Après être initialisée à 14 chiffres significatifs exacts, la précision de ces traceurs descend instantanément à 9,5 ou 10 chiffres, remonte très rapidement (en environ 30 pas de temps) à 10,75 ou 11 chiffres puis diminue doucement pour sembler tendre vers :

- **10,1 chiffres significatifs exacts** pour l'ammonium (voir ANNEXE 9)
- **10,5** pour le phytoplancton, le zooplancton, les détritiques (voir ANNEXE 10, 11)
- **10,6** pour la matière organique dissoute (voir ANNEXE 11)

- Deuxième allure de courbe

La précision des nitrates sur les 480 pas de temps forme une courbe un peu à part. Après avoir été initialisée à 14 chiffres significatifs exacts, elle diminue tout d'abord très rapidement puis plus progressivement pour se retrouver à **10,7** chiffres au bout de 480 pas de temps (voir ANNEXE 9).

- Troisième allure de courbe

Le calcul des variables biologiques nécessite celui de variables intermédiaires. Leur précision suit une courbe d'une allure semblable. Après avoir été initialisée à 14 chiffres significatifs exacts, elle diminue tout d'abord très rapidement puis plus progressivement de façon plus ou moins linéaire pour atteindre les valeurs suivantes au bout de 480 pas de temps :

- **12,6 chiffres significatifs exacts** pour  $zlnO3$  (voir ANNEXE 12) qui correspond à la limitation par les nitrates de la production de phytoplancton<sup>5</sup> (c'est une valeur comprise entre 0 et 1)
- **12,5** chiffres significatifs exacts pour  $zlnH4$  (voir ANNEXE 12) qui correspond à la limitation par l'ammonium de la production de phytoplancton
- **13,1** chiffres significatifs exacts pour  $zlle$  (voir ANNEXE 13) qui correspond à la limitation par la lumière de la production de phytoplancton
- **13,1** chiffres significatifs exacts pour  $xpar$  (voir ANNEXE 13) qui correspond à la lumière en  $\text{Watt.m}^{-2}$ .
- **12,3** chiffres significatifs exacts pour  $zno3phy$  (voir ANNEXE 14) qui correspond au débit d'ammonium utilisé par les phytoplanctons en  $\text{mmole.m}^{-3}.\text{s}^{-1}$ .
- **12,3** chiffres significatifs exacts pour  $znh4phy$  (voir ANNEXE 14) qui correspond au débit de nitrate utilisé par les phytoplanctons en  $\text{mmole.m}^{-3}.\text{s}^{-1}$ .
- **12,6** chiffres significatifs exacts pour  $znh4phy$  (voir ANNEXE 15) qui correspond au débit de phytoplanctons brouté par les zooplanctons en

---

<sup>5</sup> $Production_{phytoplancton} = zlnO3 * zlnH4 * zlle * Production_{phytoplancton}$

mmole.m<sup>-3</sup>.s<sup>-1</sup>.

On peut voir en ANNEXE 26 la concentration du phytoplancton sur notre domaine, à 91 mètres de profondeur, au bout de 480 pas de temps, et en ANNEXE 27 la précision qui lui associée. On peut difficilement établir une corrélation entre l'importance de la concentration et la précision car celle-ci varie trop peu.

On peut voir en ANNEXE 28 la concentration du phytoplancton mais cette fois-ci il s'agit d'une coupe en profondeur, de la surface à -260 mètres environ, sur toute la largeur de notre domaine pour la longueur  $y = 252$  km (moitié de la longueur totale), toujours au bout de 480 pas de temps. On peut noter que le phytoplancton est surtout présent dans la zone des 110 mètres. On peut voir, en ANNEXE 29, la précision qui est associée à ce schéma. La précision paraît être plus faible en profondeur (on avait trouver le même résultat avec ORCA).

**Smolarkiewicz** Le schéma de Smolarkiewicz avec diffusion horizontale explose à partir de la 55<sup>ème</sup> itération environ et les variables gardent une très bonne précision. L'explosion du schéma n'est donc pas dû aux erreurs d'arrondis mais plutôt au modèle mathématique utilisé.

Le schéma de Smolarkiewicz sans diffusion horizontale n'explose pas. Ces variables biologiques ont une bonne précision. L'évolution de cette précision diffère quelque peu par rapport aux mêmes variables calculées avec le schéma d'Arakawa. On peut regrouper les variables biologiques de la même façon que précédemment en trois catégories de courbes :

- Première catégorie de courbe

La précision des variables part de 14 chiffres significatifs et se stabilise très rapidement autour d'une valeur :

- **8,9 chiffres significatifs exacts (soit 1,2 chiffre en moins qu'avec Arakawa)** pour l'ammonium (voir ANNEXE 16)
- **9,3 (soit 1,2 chiffre en moins qu'avec Arakawa)** pour le phytoplancton et le zooplancton (voir ANNEXE 17)
- **9,6 (soit 0,9 chiffre en moins qu'avec Arakawa)** pour les détritiques (voir ANNEXE 18)
- **9,4 (soit 1,2 chiffre en moins qu'avec Arakawa)** pour la matière organique dissoute (voir ANNEXE 18)

- Deuxième catégorie de courbe

Ici aussi, la précision des nitrates forme une courbe un peu à part. Après avoir été initialisée à 14 chiffres significatifs exacts, elle diminue tout d'abord très rapidement puis plus progressivement pour se retrouver à **10,9** chiffres au bout de 480 pas de temps (voir ANNEXE 16).

- Troisième catégorie de courbe

Les courbes de précision des variables biologiques intermédiaires ont exactement la même allure qu'avec le schéma d'Arakawa, certaines courbes sont juste légèrement décalées à la baisse, leur précision atteint les valeurs suivantes au bout de 480 pas de temps :

- **12,4 chiffres significatifs exacts** (soit 0,2 chiffre en moins qu'avec Arakawa) pour *zlno3* (voir ANNEXE 20) (soit 0,2 chiffre en moins)

- **13,2** chiffres significatifs exacts (moins d’un dixième de différence) pour *zlle* (voir ANNEXE 21)
- **12,1** chiffres significatifs exacts (soit 0,2 chiffre en moins) pour *zno3phy* (voir ANNEXE 22)

**Conclusion** Les variables dynamiques du modèle ont une bonne précision qui semble se stabiliser (excepté pour la densité) à au moins 9 chiffres significatifs exacts. On ne peut conclure de manière générale sur la comparaison entre la précision de ces variables dynamiques et de celles calculées avec la configuration ORCA car pour certaines variables la précision est supérieure et pour d’autres inférieure et cela, sans règle apparente (du moins à première vue).

La concentration des traceurs passifs semble converger vers une très bonne précision, plus de dix chiffres significatifs exacts, et cela plus rapidement avec Smolarkiewicz qu’avec Arakawa. Cependant cette précision “limite” est d’environ un chiffre significatif exact de moins avec le schéma de Smolarkiewicz (par rapport aux résultats obtenus avec le schéma d’Arakawa).

Quant à la précision des variables biologiques intermédiaires, elle diminue encore au 480<sup>ème</sup> pas de temps donc il faudrait plus de pas de temps pour éventuellement déterminer leur valeur limite. On peut d’ores et déjà dire qu’elles ont le même nombre de chiffres significatifs exacts avec le schéma d’Arakawa et celui de Smolarkiewicz à deux dixièmes près en moyenne pour certaines variables.

Si nous avons disposé de plus de temps il aurait été intéressant de regarder les résultats sur 10 ou 15 jours de simulation soit 2400 ou 3600 pas de temps car c’est la durée habituellement utilisée avec cette configuration, on aurait, alors, mieux pu entrevoir la précision vers laquelle chaque variable semble tendre.

## 5.5 Conservation des traceurs

La somme des concentrations des variables biologiques doit théoriquement se conserver dans le temps car, comme le disait Lavoisier, “Rien ne se perd, rien ne se crée, tout se transforme”. On peut se reporter au schéma de l’ANNEXE 1 pour les échanges entre ces variables à la surface.

Avec le schéma d’Arakawa (avec de la diffusion horizontale), la somme des concentrations des traceurs passifs (pour les six variables biologiques) ne se conserve pas au cours des itérations. Elle diminue significativement (voir le premier schéma en ANNEXE 25).

Pour tenter de voir à partir de quelle subroutine la somme des traceurs n’est pas conservée, nous avons commencé à afficher la somme des tendances (elles sont utilisées pour le calcul de la concentration des traceurs passifs) avant et après la première subroutine qui intervient dans le calcul de ces tendances (*trcbio*). Avant cette subroutine, la somme est nulle avec 14 chiffres significatifs exacts ce qui correspond à sa valeur d’initialisation et après cette subroutine la somme n’est pas significative (ce qui ne veut pas dire que les tendances, elles, ne sont pas significatives). On ne peut donc continuer plus loin dans cette direction de recherche.

Quand on utilise le schéma de Smolarkiewicz, avec de la diffusion horizontale, la somme des concentrations des variables biologiques diminue, jusqu'à ce que le schéma explose, à partir de ce moment là, la somme des traceurs explose aussi, elle oscille de plus en plus au cours des itérations (voir le deuxième schéma ANNEXE 25). Le schéma a été pris à la même échelle que le précédent pour pouvoir mieux les comparer, c'est pourquoi les valeurs 'sortent' du schéma.

Enfin dans le cas du schéma de Smolarkiewicz, sans diffusion horizontale, la somme des traceurs passifs augmente de façon significative (voir ANNEXE 26). Son augmentation est plus faible que la diminution dans le cas d'Arakawa donc **le schéma de Smolarkiewicz sans diffusion horizontale est plus conservatif que le schéma d'Arakawa avec diffusion horizontale, et cela semble être dû à la diffusion horizontale.**

## 5.6 Planning

Le temps imparti pour ce projet de fin d'étude était d'environ 55 jours répartis sur deux jours par semaine de janvier à mai et cinq jours par semaine au mois de juin (2000).

On peut découper ces 55 jours de la manière suivante :

- Environ 6 jours pour trouver des moyens pour accélérer l'exécution du code (les fonctions de CADNA ont été 'inlinés')
- Environ 17 jours pour cadnatiser le code OPA-TRC-NNPZDDOM à partir du code ORCA
- Environ 24 jours pour avoir les résultats (débogage) et les analyser
- Environ 8 jours pour faire le rapport

# Conclusion

Ce projet de fin d'étude m'a permis de mieux comprendre le code OPA, que j'avais déjà entrevu lors de l'étude sur la stabilité numérique d'ORCA. Il m'a également permis d'approfondir la connaissance de la bibliothèque CADNA, qui je l'espère sera de plus en plus utilisé, tant par les industriels que par les chercheurs.

Il était nécessaire de passer en revue l'étude mathématique du code pour souligner que l'algorithme du gradient conjugué préconditionné converge théoriquement et que les schémas numériques utilisés respectent les critères de stabilité, car c'est seulement à ces conditions que l'étude de la stabilité numérique à un sens.

D'un point de vue informatique, l'algorithme du gradient conjugué 'calcule bien' (mise à part lors du calcul de la fonction de courant barotropique de l'île) même si la solution obtenue n'atteint pas toujours la précision qu'elle pourrait atteindre.

Les variables dynamiques (ou traceurs actifs) possèdent une bonne précision, ce qui ne nous a pas surpris car c'était déjà le cas avec la configuration ORCA (les schémas numériques utilisés n'ont pas changés). La nouveauté de cette étude résidait, mise à part le domaine et les conditions initiales, dans le couplage avec un code de biologie marine NNPZDDOM.

Les variables de biologie marine (ou traceurs passifs), calculées avec le schéma numérique d'Arakawa ont une bonne précision, ce qui ne nous a pas non plus vraiment surpris car c'est le schéma numérique utilisé pour les variables dynamiques.

Par contre nous n'avions aucun a priori sur la précision des variables biologiques avec le schéma de Smolarkiewicz (sans diffusion horizontale), si ce n'est que les océanographes avaient des résultats cohérents par rapport à la réalité. La précision des variables biologiques avec ce schéma s'est révélée quasiment aussi élevée qu'avec le schéma d'Arakawa (au maximum 1,2 chiffre significatif exact de moins).

Par manque de temps, les instabilités numériques n'ont pu être recherchées, mise à part dans l'algorithme du gradient conjugué préconditionné qui a été 'décortiqué' séparément. Cependant, vu que dans l'étude concernant la configuration ORCA, aucune instabilité n'ayant des conséquences n'avaient été repéré et que dans notre configuration les variables ont une très bonne précision, même si on découvrait des instabilités, il y aurait peu de chances qu'elles aient des conséquences telles que les variables n'auraient plus assez de chiffres significatifs exacts (seuls deux ou trois chiffres sont vraiment nécessaires).

Par manque de temps également, nous n'avons pu envisager de perturber les données (en fixant leur précision initiale), cependant cela aurait été moins utile que dans l'étude précédente car les données ne sont pas lues dans un fichier *restart*

donc nous n'aurions pas pu extrapoler les résultats à un nombre de pas de temps supérieur.

Ce projet de fin d'étude a permis de mettre en évidence la bonne qualité numérique du code OPA-TRC-NNPZDDOM dans son ensemble, que ce soit avec l'utilisation du schéma d'Arakawa ou de Smolarkiewicz pour la biologie marine. Il a également permis de mettre en évidence que les deux schémas numériques utilisés pour la biologie marine ne sont pas conservatifs.

Un autre schéma numérique pour la biologie marine est actuellement à l'étude, il serait donc intéressant, dans le futur, d'étudier sa stabilité numérique.



# Bibliographie

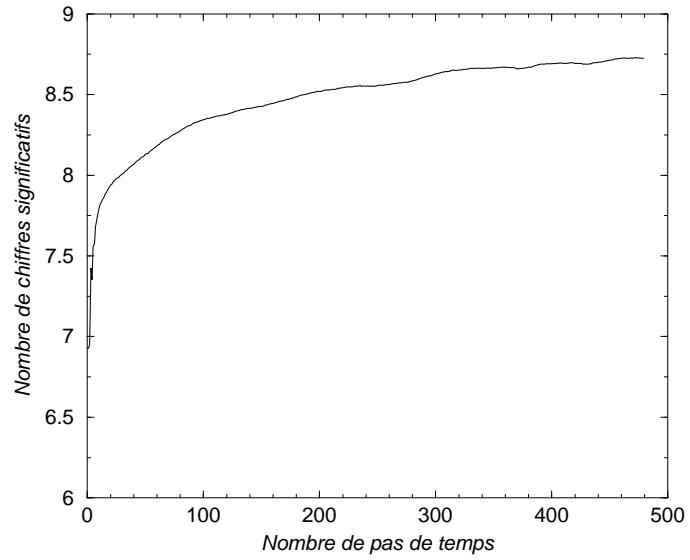
- [1] Site de l'IPSL : <http://www.ipsl.jussieu.fr>
- [2] Etude précédente : <http://www.ipsl.jussieu.fr/modelisation/liste-notes-tech.html>
- [3] Site du LODYC : <http://www.lodyc.jussieu.fr>
- [4] Site du LIP6 : <http://www.lip6.fr>
- [5] Documentation de CADNA :  
<http://www-anp.lip6.fr/chpv/francais/cadna/index.html>
- [6] Site du CCR : <http://www.ccr.jussieu.fr>
- [7] Notes du Pôle de Modélisation n°11 de décembre 1998 sur OPA 8.1
- [8] Fabienne JEZEQUEL et Jean-Marie CHESNEAUX, Etude de la stabilité numérique du code OPA-version 8.0, voir aussi [2]
- [9] Jean-Marie CHESNEAUX, Stéphane GUILAIN et Jean VIGNES, La bibliothèque CADNA Présentation et utilisation, voir aussi [6]
- [10] Gurvan MADEC, Pascale DELECLUSE, Maurice IMBARD et Claire LEVY, Notes du Pôle de Modélisation de l'IPSL n°11 de décembre 1998 sur OPA 8.1, Laboratoire d'Océanologie DYnamique et de Climatologie
- [11] LEVITUS S., 1982, Climatological Atlas of the World Ocean, NOAA professional paper 13, december 1982
- [12] Jean-Marie CHESNEAUX, L'arithmétique stochastique et le logiciel CADNA, Habilitation à diriger des recherches, Université de Pierre et Marie Curie, Paris (1995)
- [13] Fanny BONNET-TERRAS, Etude de la stabilité numérique du code ORCA, voir aussi [2]
- [14] F.MESINGER et A.ARAKAWA, Numerical methods used in atmospheric models, GARP (Global Atmospheric Research Programme)publications series N°17 August 1976
- [15] Marina LEVY, Modélisation des processus biogéochimiques en méditerranée Nord-Occidentale Cycle Saisonnier et Variabilité Mésoéchelle, thèse au sein du LODYC, 1996
- [16] IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985. Institute of Electrical and Engineers, Août 1985.
- [17] Jean-Marie CHESNEAUX, Study of the computing accuracy by using probabilistic approach, Contribution to Computer Arithmetic and Self-Validating Numerical Methods, ed. C. Ulrich, (J.C. DBltzer), 1990, P.19-30.

- [18] J.VIGNES et M. LA PORTE, Error analysis en computing in : Information Processing 74, North-Holland, 1974
- [19] J.VIGNES, Review on stochastic approach to round-off error analysis and its applications, Math. Comp. Simul., 30, 198, P.481-491.
- [20] M. PICCHAT et J. VIGNES, Ingénierie du contrôle de la précision des calculs sur ordinateur, Editions Technip, 1993.
- [21] Jean VIGNES, Zéro mathématique et zéro informatique, C.R. Acad. Sci. Paris Sér. I Math. 303 (1986) 997-1000 ; also : La Vie des Sciences 4 (1) (1987) 1-13
- [22] Jean-Marie CHESNEAUX, Modélisation théorique et conditions de validité de la méthode CESTAC, C.R.A.S., Paris, Série 1, tome 307, 1988, p.417-422
- [23] J.-M. CHESNEAUX, Y. SALHI, Theoretical study of C.E.S.T.A.C. method and ADA implmentation, Numerical and applied Mathematics, ed. C. Brezinski, (J.C. Baltzer), 1989, P.617-621.
- [24] J.-M. CHESNEAUX, Etude théorique et implémentation en ADA de la méthode CESTAC, Doctorat d'Université P. et M. Curie, 1988
- [25] J.-M. CHESNEAUX et J. VIGNES, Sur la robustesse de la méthode CESTAC, C.R.A.S., Paris, t.307, série 1, 1988, P.855-860.
- [26] J.-M.MULLER, Arithmétique des ordinateurs, Ed. Masson, 1989
- [27] William H. PRESS, William T. VETTERLING, Saul A. TEUKOLSKY, Brian P. FLANNERY, Numerical Recipes in Fortran, The Art of Scientific Computing, Second Edition
- [28] A. DRAUX, cours d'Analyse Numérique, Génie Mathématique 3ème année (GM3), INSA de Rouen.
- [29] B. GLEYSE, cours d'Algorithme et d'Arithmétique, Génie Mathématique 3ème année (GM3), INSA de Rouen.
- [30] J. DUMONTET, J. VIGNES, Détermination du pas optimal dans les calculs des dérivées sur ordinateur, R.A.I.R.O, Vol 11, n°1, 1977, p. 13-25

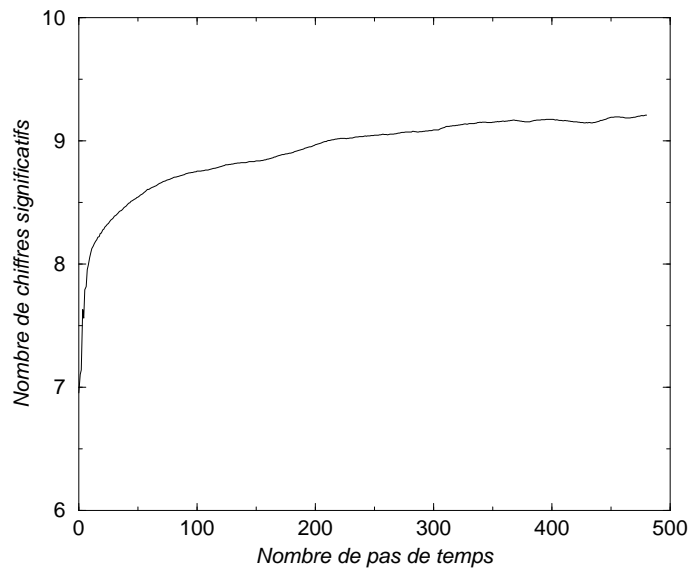
# ANNEXE 1

# ANNEXE 2 : variables dynamiques

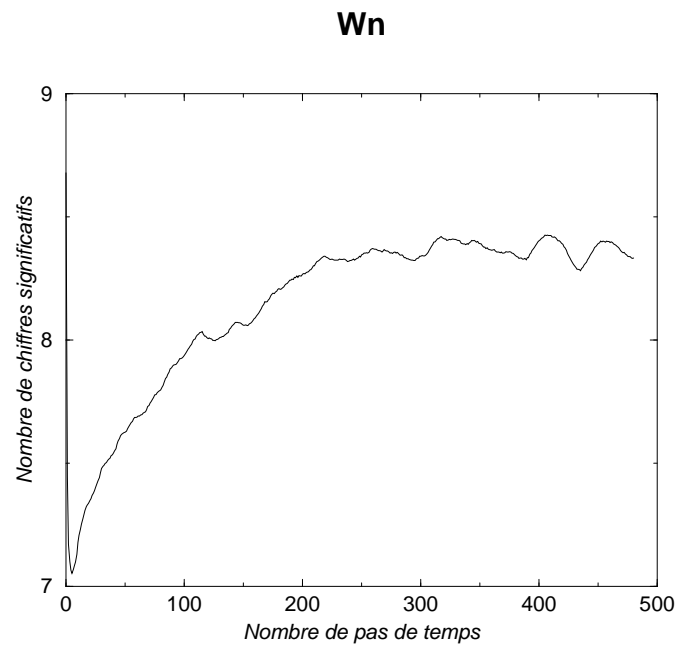
**un**



**Vn**

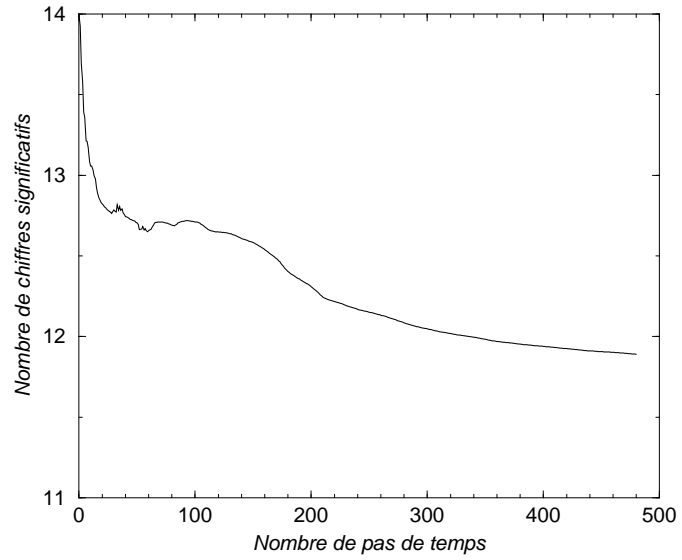


# ANNEXE 3 : variables dynamiques

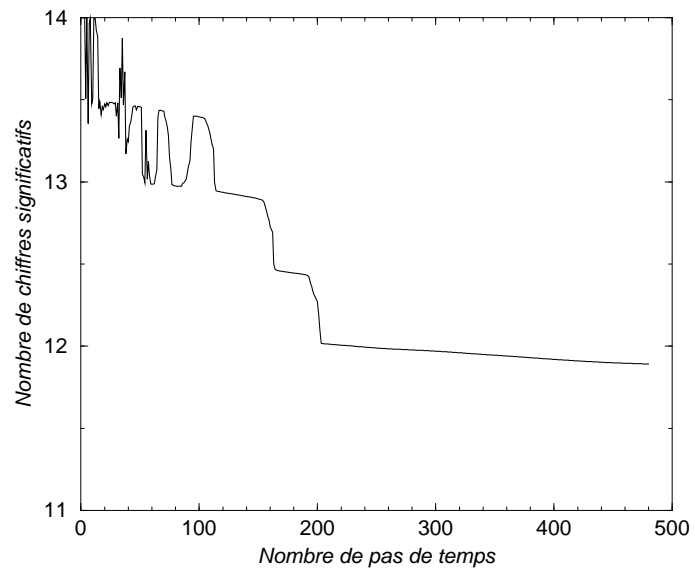


# ANNEXE 4 : variables dynamiques

## Temperature : tn

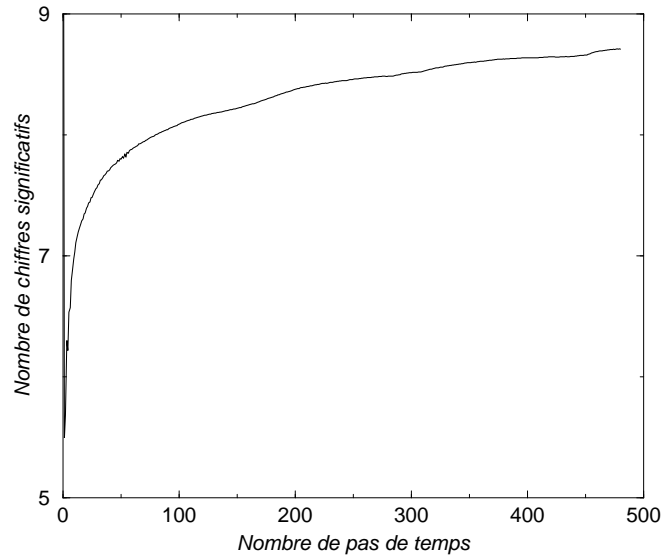


## Salinite : sn

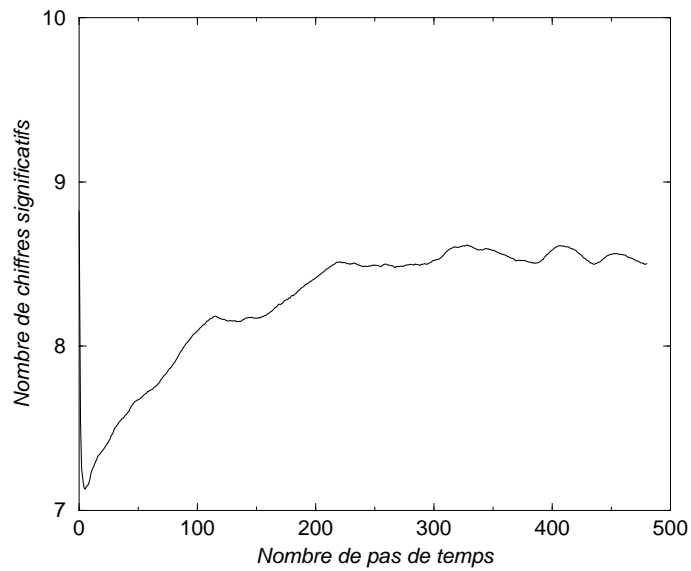


# ANNEXE 5 : variables dynamiques

## Vorticité relative : rotn

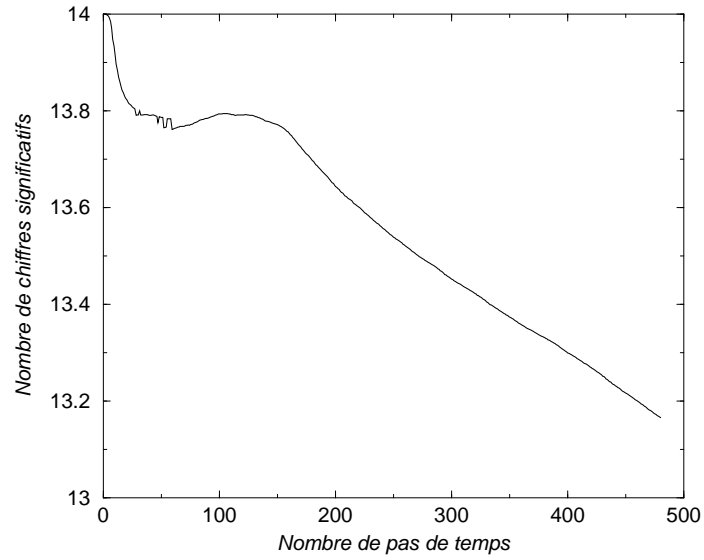


## Divergence horizontale : hdivn

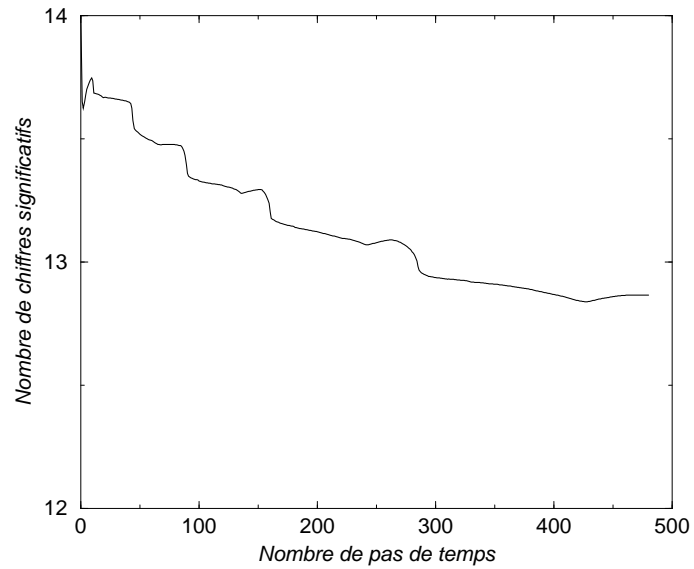


# ANNEXE 6 : variables dynamiques

## Densite : rdn



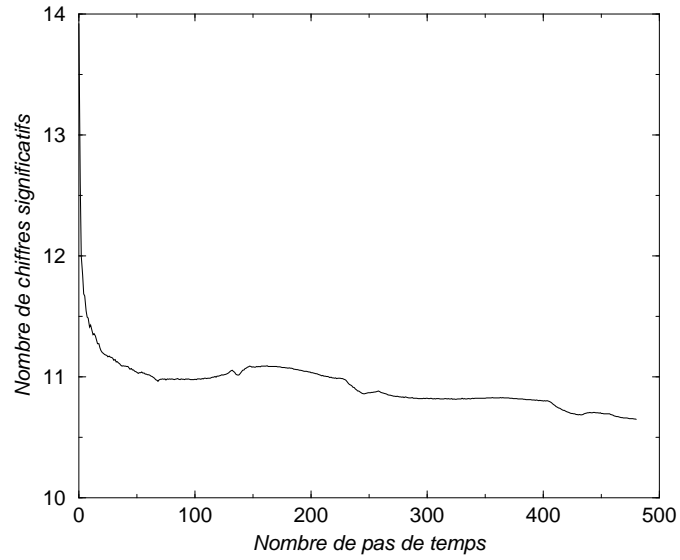
## Energie : en



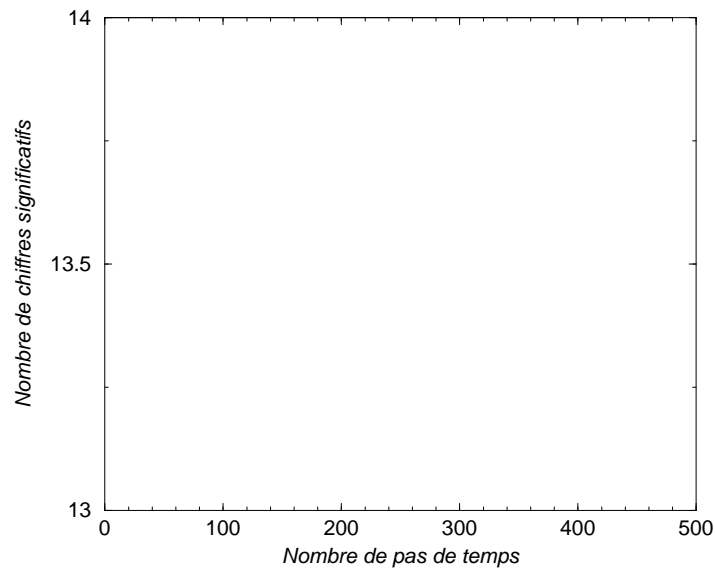


# ANNEXE 7 : variables dynamiques

## Frequence de Brunt–Vaisala : bn2n

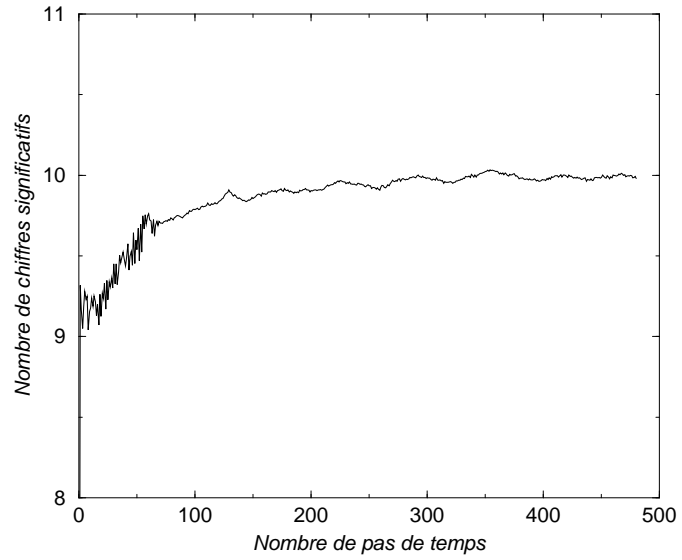


## q et qsr (chaleur a la surface)

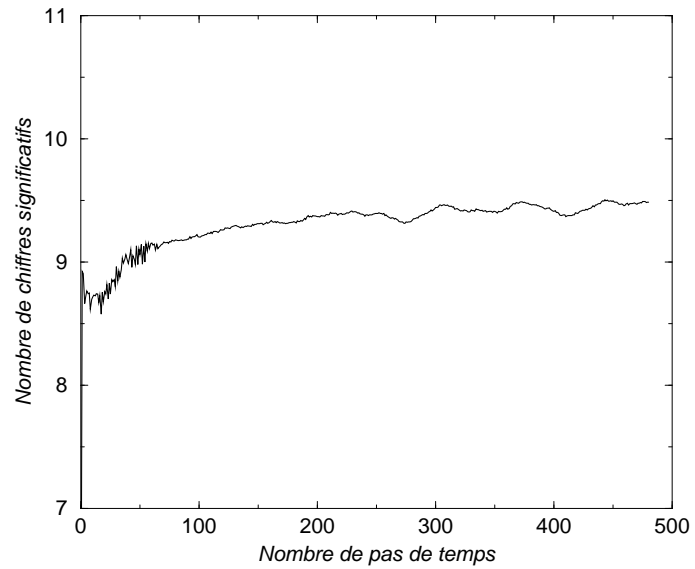


# ANNEXE 8 : variables dynamiques

## Gradient de pression de surface : spgu

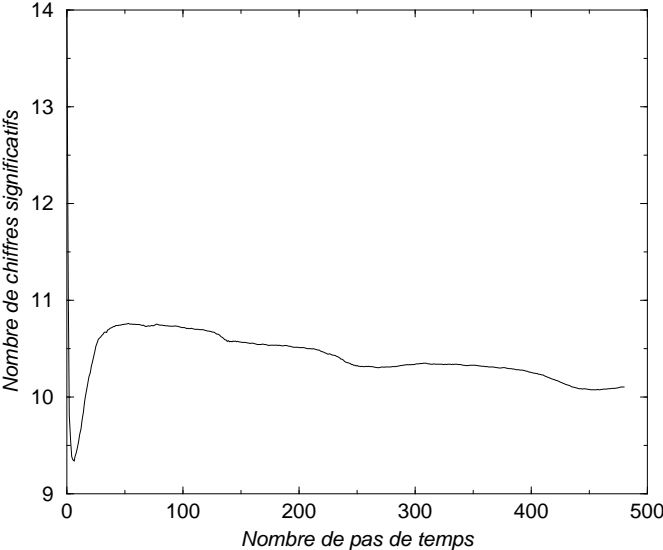


## Gradient de pression de surface : spgv

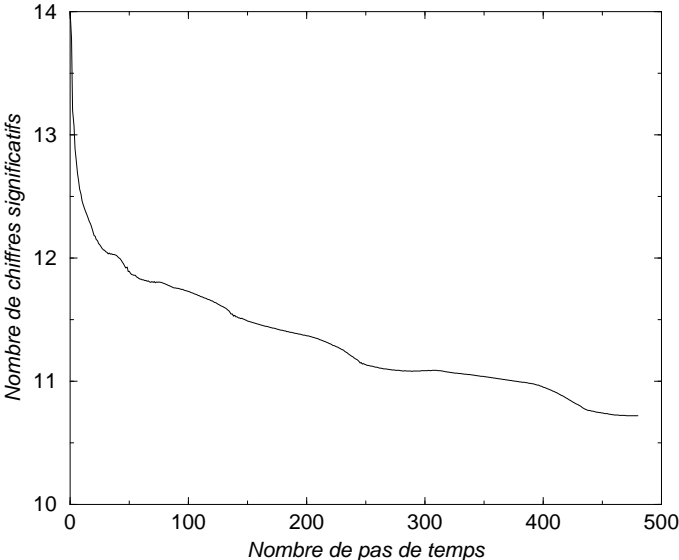


# ANNEXE 9 : Variables biologiques avec Arakawa

## Amonium

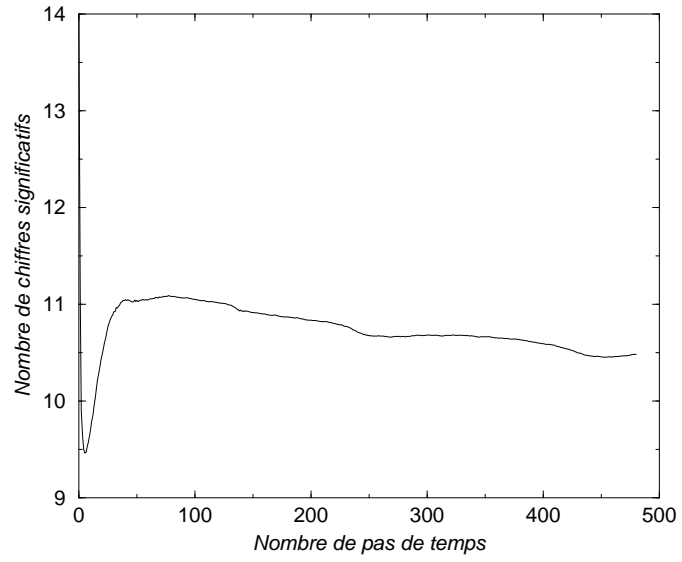


## Nitrate

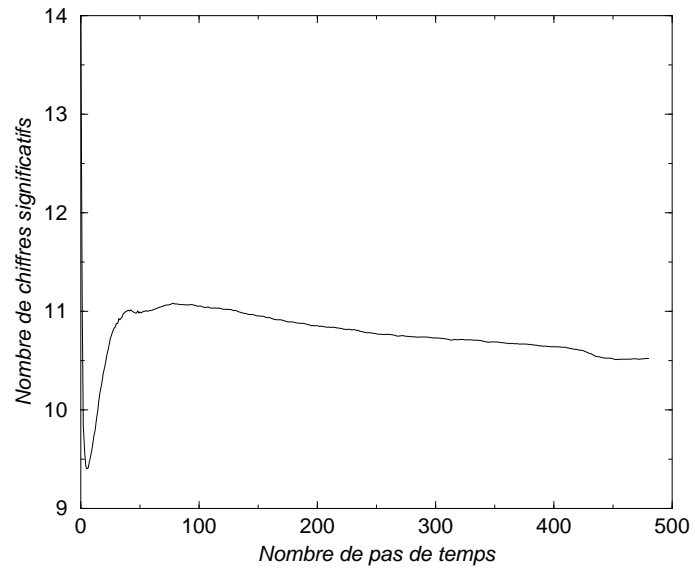


# ANNEXE 10 : Avec Arakawa

## Phytoplancton

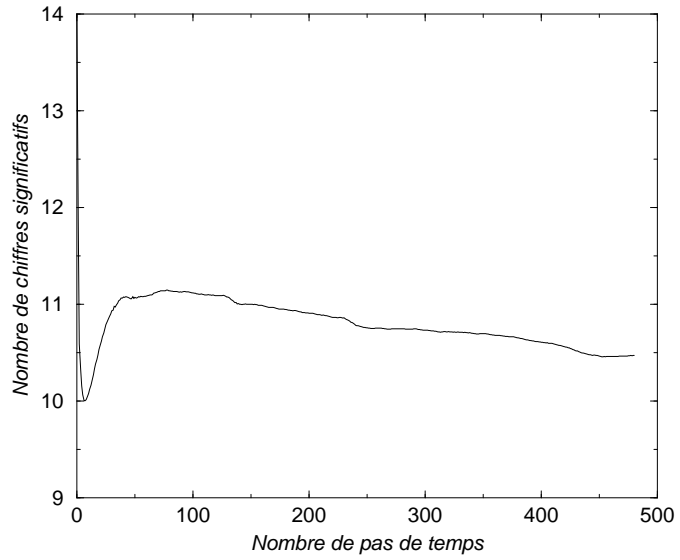


## Zooplancton

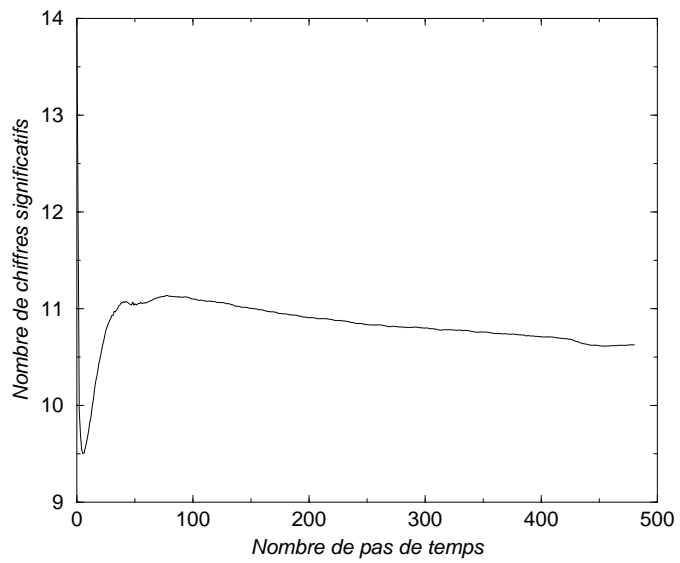


# ANNEXE 11 : Avec Arakawa

## Detritus

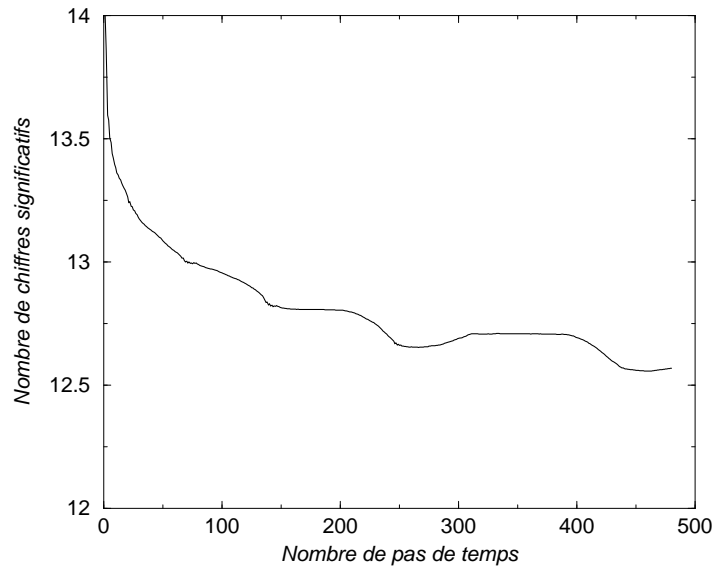


## Matiere organique dissoute

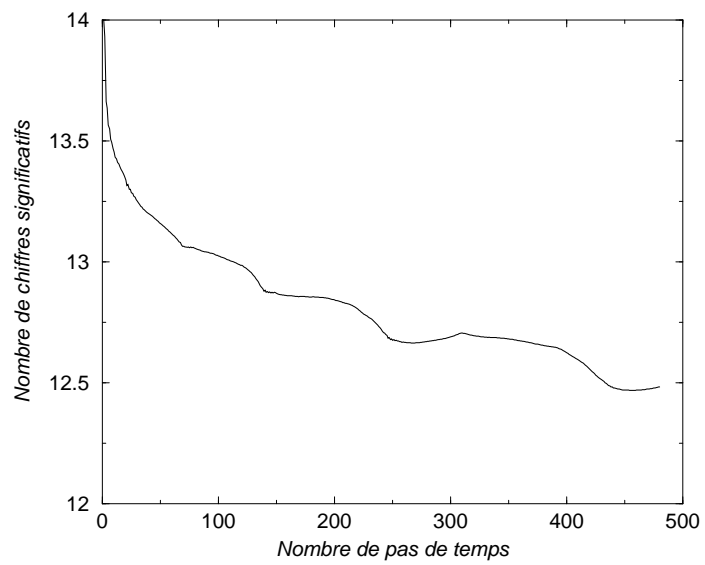


# ANNEXE 12 : Avec Arakawa

## zln03

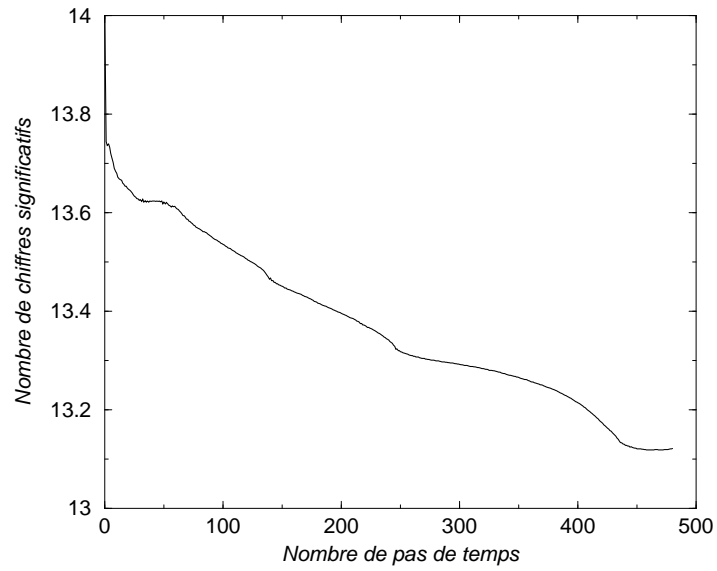


## zlnh4

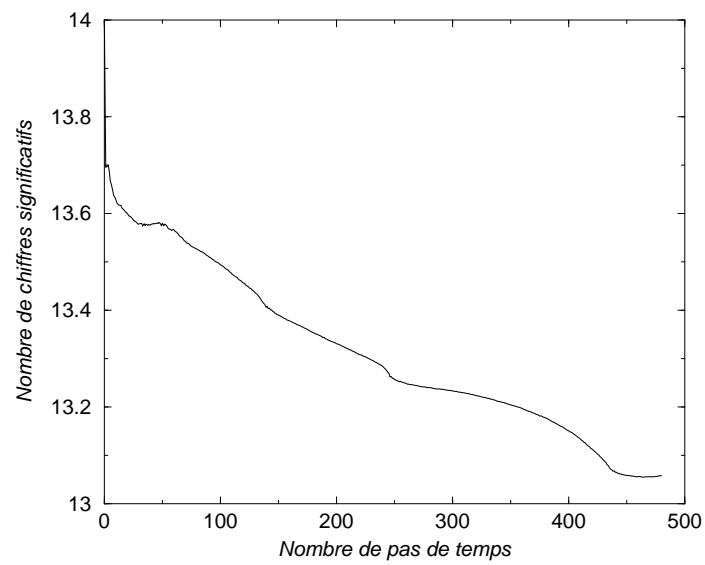


# ANNEXE 13 : Avec Arakawa

**zlle**

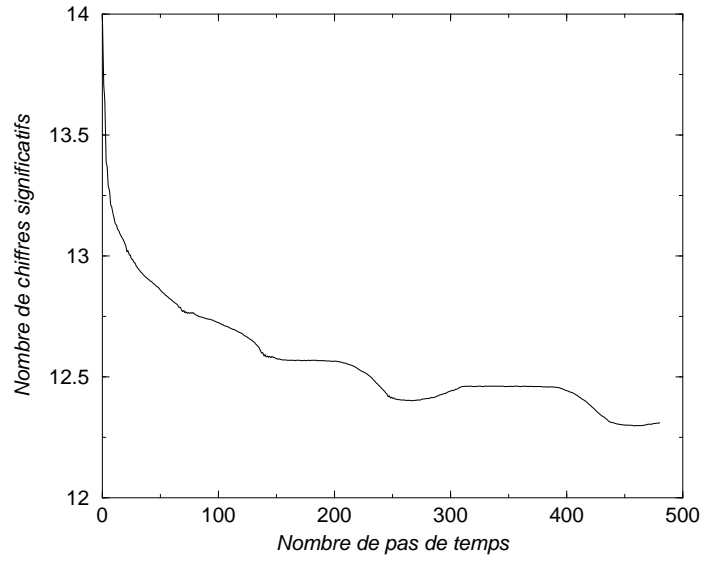


**xpar**

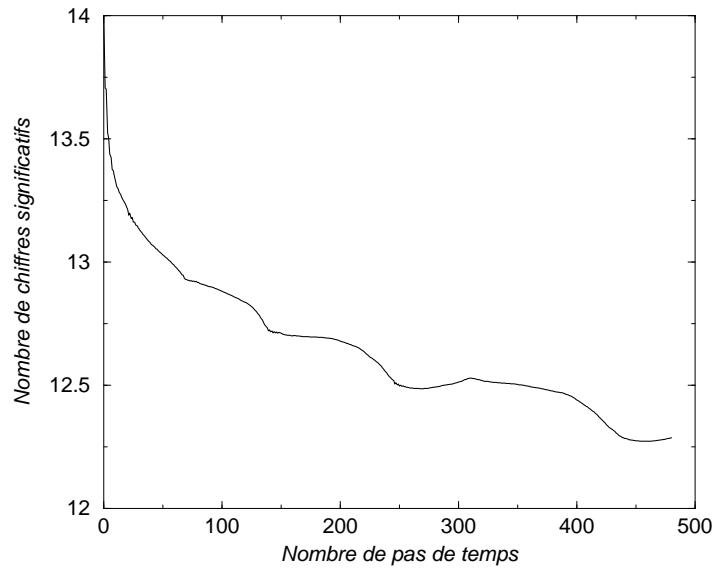


# ANNEXE 14 : Avec Arakawa

## zno3phy



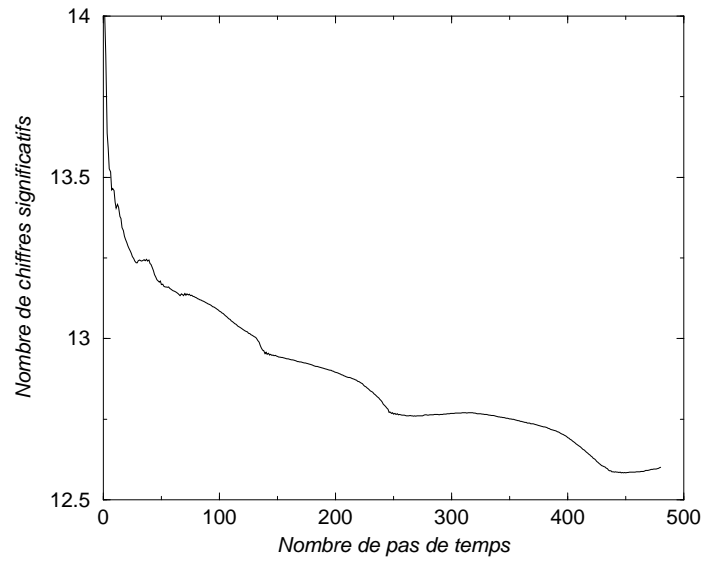
## znh4phy





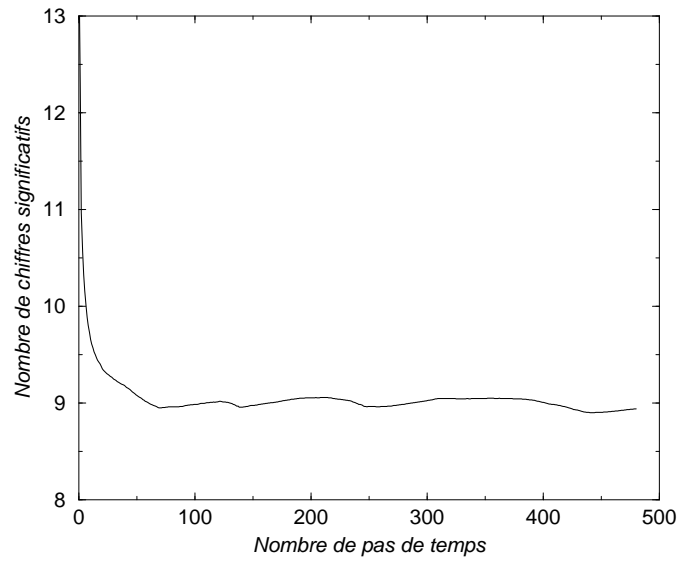
# ANNEXE 15 : Avec Arakawa

## zphysoo

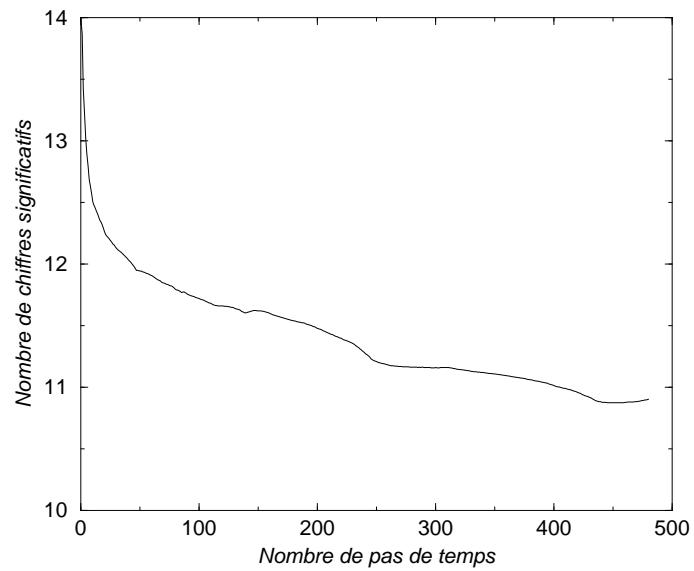


# ANNEXE 16 : Variables biologiques avec Smolarkiewicz

## Amonium

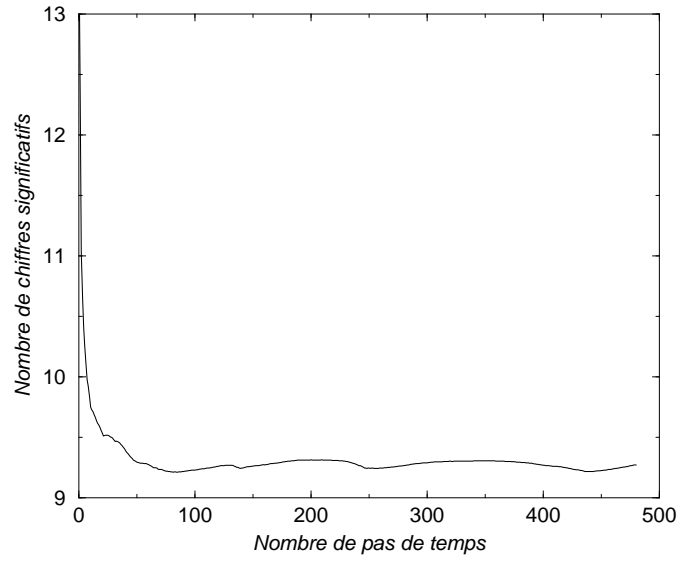


## Nitrate

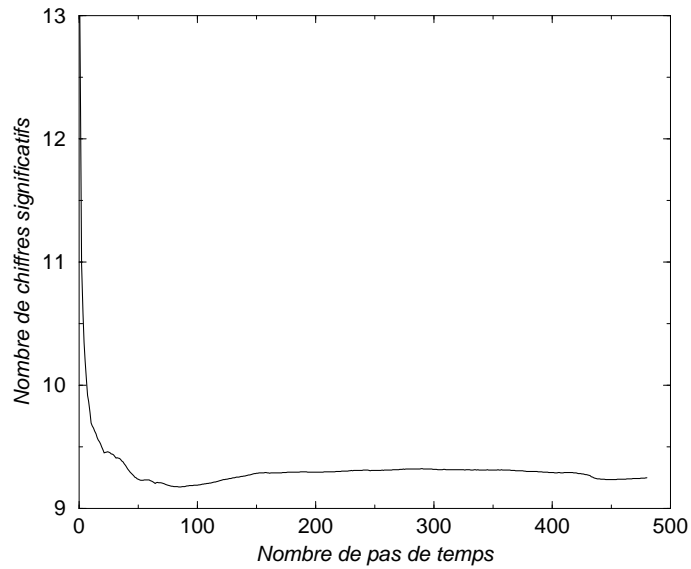


# ANNEXE 17 : Avec Smolarkiewicz

## Phytoplancton

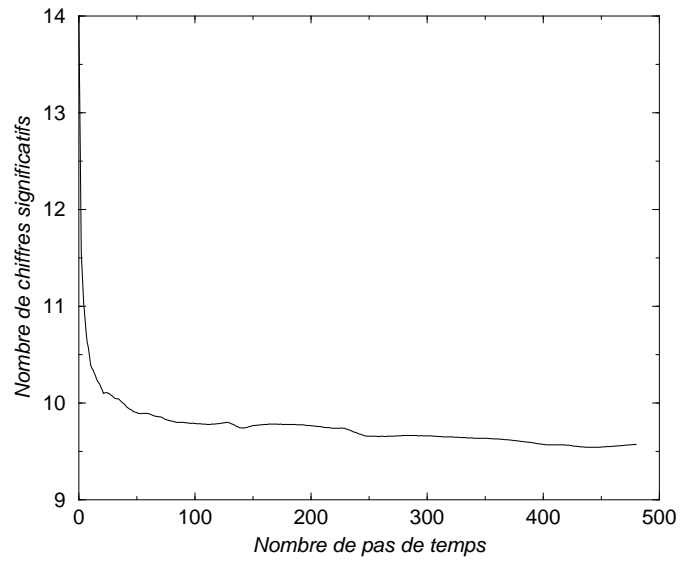


## Zooplancton

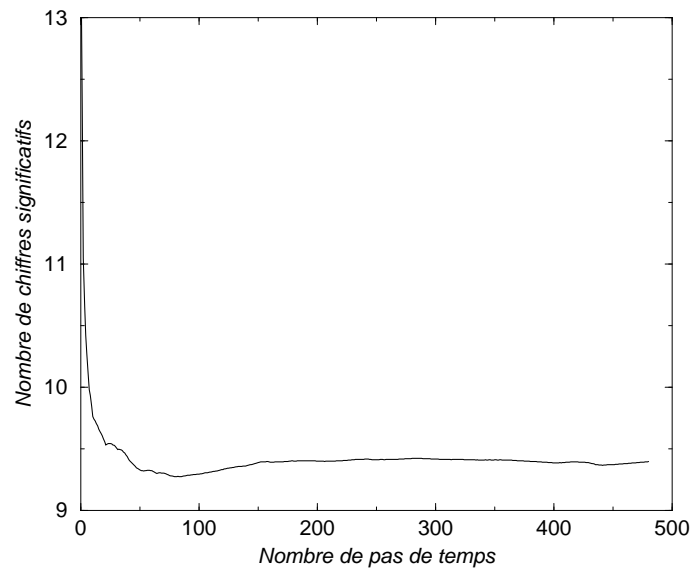


# ANNEXE 18 : Avec Smolarkiewicz

## Detritus

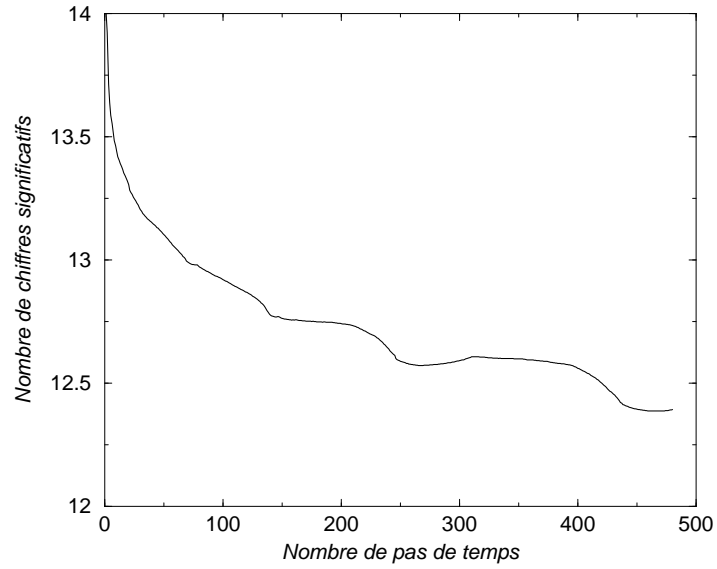


## Matiere organique dissoute

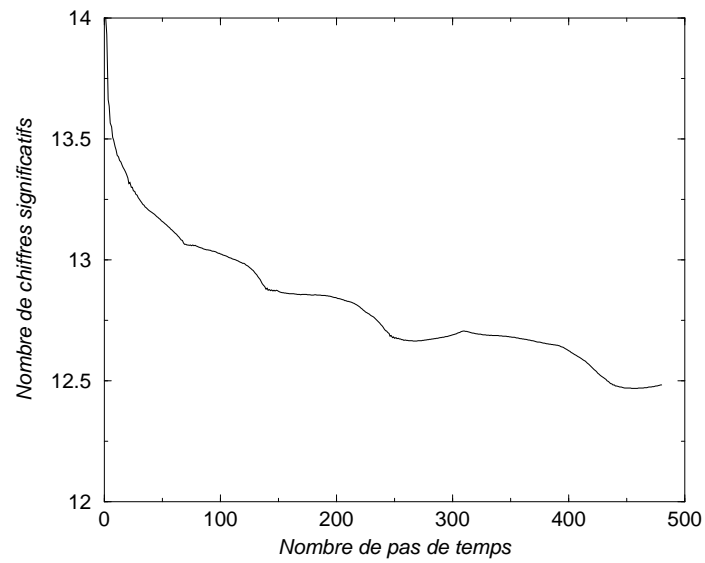


# ANNEXE 20 : Avec Smolarkiewicz

## zIno3

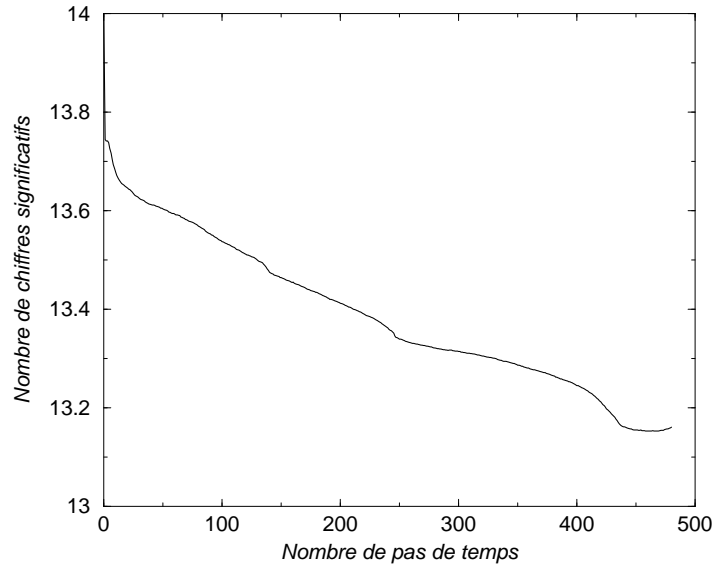


## zInh4

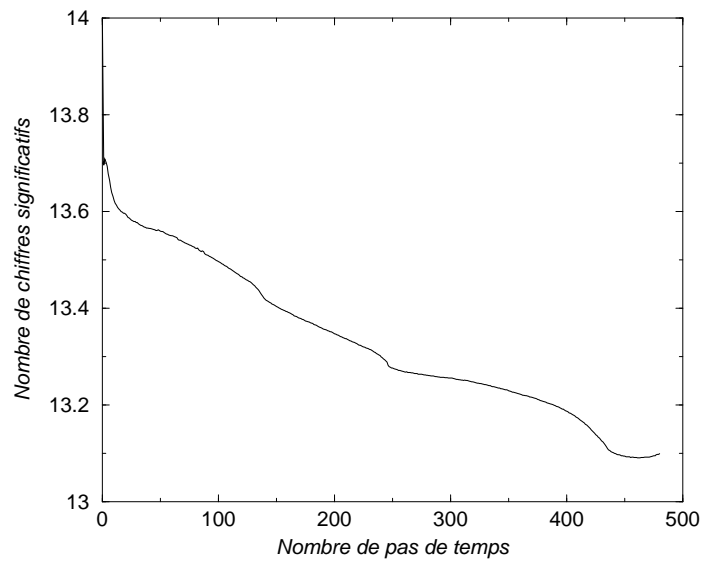


# ANNEXE 21 : Avec Smolarkiewicz

**zlle**

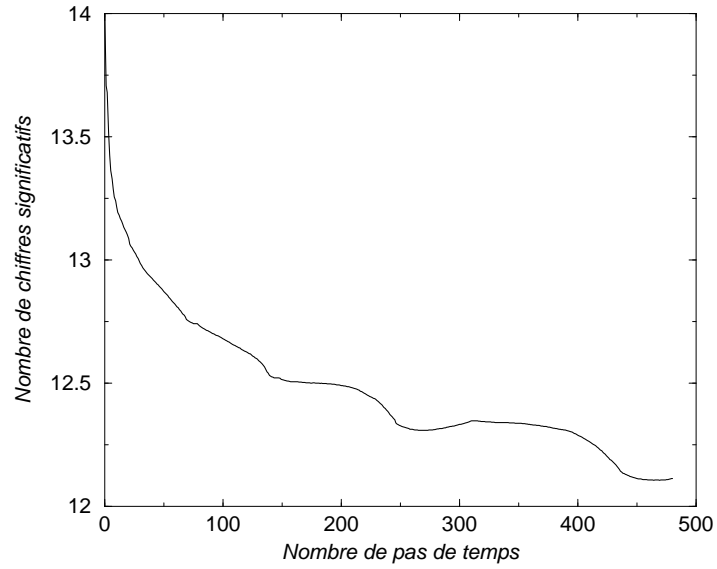


**xpar**

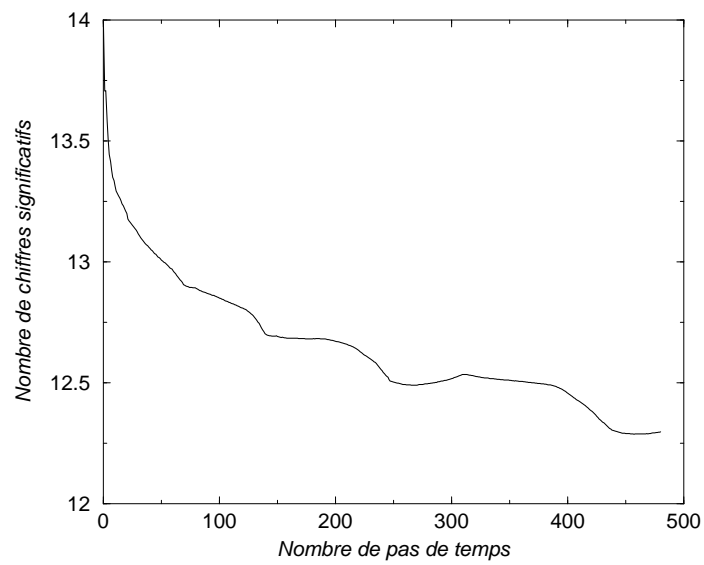


# ANNEXE 22 : Avec Smolarkiewicz

## zno3phy

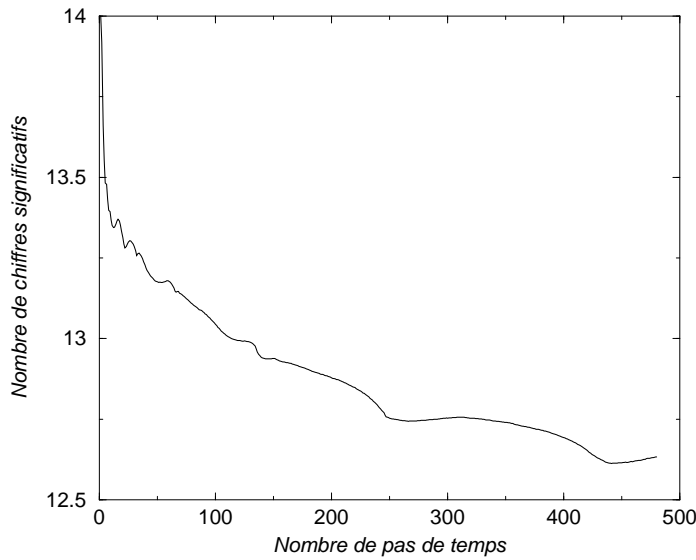


## znh4phy



ANNEXE 23 : Avec Smolarkiewicz

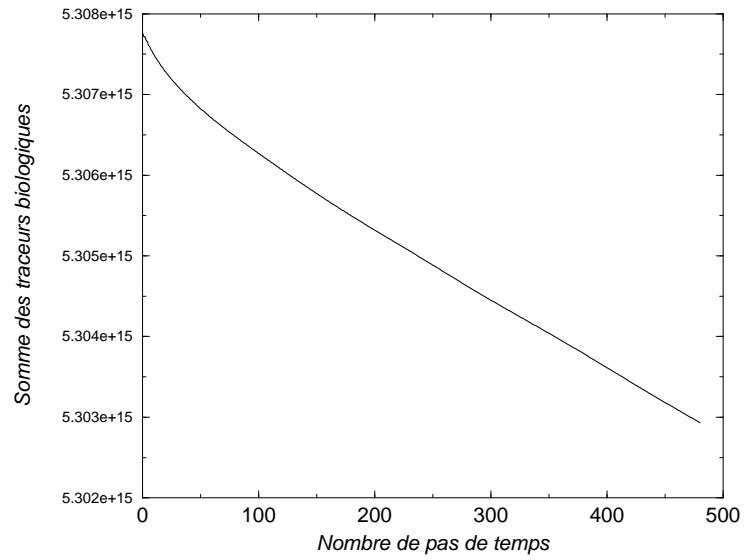
**zphyzoo**



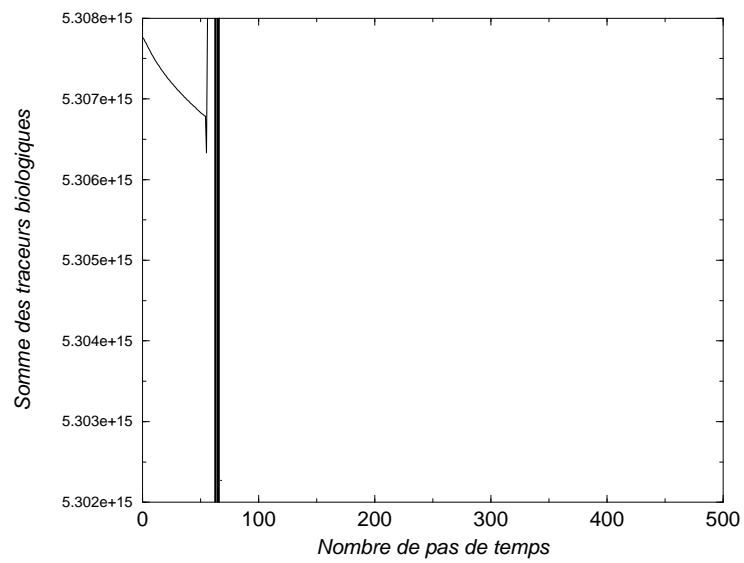


# ANNEXE 24 : Conservation des traceurs biologiques

## Avec Arakawa



## Avec Somlarkiewicz, avec diffusion



# ANNEXE 25 : Conservation des traceurs biologiques

**Avec Smolarkiewicz, sans diffusion**

