



5^{ème} Rapport d'avancement des travaux
Développements associés au modèle MIMOSA :
interface d'initialisation avec les niveaux modèles
ECMWF



Développements :	M. –A. Drouin, R. Thiéblemont, N. Huret	
Rédigé par :	M. –A. Drouin, N. Huret, R. Thiéblemont	
Version :	05/2010	Création
	12/2011	Refonte totale de la partie II.6 Validation Ajout de description pour les d'autres outils développés Modification de la partie Bilan, ajout des tests avec les autres compilateurs.

Notations.....	2
Références	2
Introduction.....	3
I. Conversion du code MIMOSA en FORTRAN 95.....	3
I.1. Intérêt	3
I.2. Modifications.....	3
I.3. Validation.....	4
II. Passage de MIMOSA en niveau modèle ECMWF	5
II.1. Intérêt.....	5
II.2. Format des données en niveaux modèle.....	5
II.3. subroutine readgrib.f95	6
II.4. subroutine pvp.f95	7
II.5. programme filament.....	7
II.6. Validation.....	9
II.6.1. Validation des résultats avec différents compilateurs FORTRAN.....	9
II.6.2. Validation des simulations initialisées en niveaux modèle	11
III. Bilan de l'évolution.....	16
III.1. Performances	16
III.2. Utilisation	17
IV. Développement d'outils	18
IV.1. MakeFile.....	18
IV.2. Script ECMWF	18
IV.3. Script mimosa.sh	18
IV.3. Outils graphiques IDL.....	19
IV.3.1. Procédure de lecture des sorties binaires de MIMOSA	19
IV.3.2. Procédures de traçage	19
IV.4. Script de création d'animations	20
Bibliographie	21

Notations

ASCII	American Standard Code for Information Interchange
AVC	Advance Video Coding
ECMWF	European Centre for Medium-Range Weather Forecasts
GRIB	GRIddec Binary
grib_api	GRIddec Binary Application Programming Interface
MARS	Meteorological Archival and Retrieval System
MIMOSA	Modélisation Isentrope du transport Mésos-échelle de l'Ozone Stratosphérique par Advection
mp4	Extension officielle du format MPEG-4
MPEG	Moving Picture Experts Group
PV	Vorticité Potentielle (pvu, $1 \text{ pvu} = 1 \text{ K.kg}^{-1}.\text{m}^2.\text{s}^{-1}$)
x264	Bibliothèque libre permettant de coder des flux vidéo MPEG-4 AVC
MIMOSA	Modélisation Isentrope du transport Mésos-échelle de l'Ozone Stratosphérique par Advection
ECMWF	European Centre for Medium-Range Weather Forecasts
PV	Vorticité Potentielle (pvu, $1 \text{ pvu} = 1 \text{ K.kg}^{-1}.\text{m}^2.\text{s}^{-1}$)

Références

Bibliothèque IDL Coyote	http://www.idlcoyote.com/
ECaccess/ECtrans	http://www.ecmwf.int/services/ecaccess/guide/
ffmpeg	http://ffmpeg.org/
grib_api	http://www.ecmwf.int/publications/manuals/grib_api/index.html
imagemagick	http://www.imagemagick.org/script/index.php
jasper	http://www.ece.uvic.ca/~frodo/jasper/
Lait PV	Lait, 1993
MARS	http://www.ecmwf.int/publications/manuals/mars/guide/index.html
MIMOSA	Hauchecorne et al., 2002

Introduction

MIMOSA est un modèle haute résolution d'advection de vorticit  potentielle d velopp  en FORTRAN 77 par A. Hauchecorne (Hauchecorne et al., 2002). Il est initialis    un instant t   partir des donn es (champs de vents horizontaux U et V , temp rature et pression) de l'ECMWF sur une grille orthogonale centr e sur le p le nord. MIMOSA calcule puis advecte la vorticit  potentielle sur la gamme d'isentrope [350 K : 950 K] avec une r solution de 1/3 ou 1/6 de degr .

La version de MIMOSA utilis e jusqu'  maintenant utilisait pour son initialisation les donn es sur niveaux pressions de l'ECMWF. Ce document d crit les d veloppements r alis s pour permettre au mod le d' tre initialis  gr ce aux donn es sur les niveaux du mod le de l'ECMWF. L'int r t d'une telle modification est de permettre de couvrir une plus grande gamme de temp rature potentielle avec MIMOSA.

Cette  volution de MIMOSA a  t  l'occasion d'am liorer le code pour faciliter son initialisation sans recompilation

I. Conversion du code MIMOSA en FORTRAN 95

I.1. Int r t

Le principal int r t du passage au FORTRAN 90/95 est de pouvoir utiliser l'allocation dynamique des tableaux dans les calculs. Ainsi il ne sera plus n cessaire de recompiler le code lorsque la grille des donn es d'entr e sera modifi e. On peut aussi esp rer un gain de temps de calcul.

Au LPC₂E, MIMOSA pour pouvoir  tre utilis  n cessitait d' tre compil  avec le compilateur FORTRAN de Portland Group. La compilation retournait des erreurs lorsqu'elle  tait faite avec le compilateur gratuit GFORTRAN ou le compilateur IFORT d'Intel. On peut esp rer  tendre la vari t  des compilateurs utilisables.

I.2. Modifications

L'ensemble des modifications apport es au code est list  ci-dessous :

-  criture du code en format libre.
- passage en *implicit none*, c'est- -dire que toutes les variables doivent  tre d clar es.
- Standardisation des d clarations de variables c'est- -dire au format :
 - o *type, attributs :: variable*
- Division du programme unique en de multiples fichiers contenant chacun une subroutine
- Utilisation de modules pour les interfaces des fonctions et des sousroutines ainsi que la d claration des constantes.
- suppression des fonctions internes qui ne sont pas g n riques ou qui sont obsol tes
 - o *encode* qui a  t  remplac  par *write(unit=variable)*

- o suppression des fonctions `sind`, `cosd`, `tand` qui ne sont pas reconnues par tous les compilateurs par $\sin(\text{var} \frac{\pi}{180})$, $\cos(\text{var} \frac{\pi}{180})$, $\tan(\text{var} \frac{\pi}{180})$. Où *var* est la variable à laquelle on veut appliquer la fonction.
- o Remplacement des `.eq.`, `.gt.`, `.ge.`, `.lt.`, `.le.` qui sont maintenant obsolètes par `==`, `>`, `>=`, `<` et `<=`
- o Remplacement de la fonction interne `float` par la fonction interne `real`.
- o Remplacement des `go to` par des conditions `if` ou `select case` lorsque c'est possible.
- Allocation dynamique des différents tableaux dépendant de la taille des données d'entrée ou de choix de configuration de l'utilisateur.
- Changements des unités utilisées pour la lecture/écriture des fichiers. Par exemple, l'écriture du fichier de sortie des champs de vorticité potentielle utilisait l'unité 6 qui est normalement celle réservé aux sorties écran. Cela provoquait un blocage des sorties écrans lors de certaines simulations. On a donc configuré les sorties de lecture/écriture sur des unités supérieures à 20. Pour les sorties stations, on utilise des unités supérieures à 50.
- Ecriture de *makefile* spécifique à chaque compilateur avec le lequel MIMOSA a été testé.
- Utilisation d'une *namelist* à la norme FORTRAN 95 qui permet de configurer tous les paramètres du modèle et de les commenter

1.3. Validation

Pour valider les modifications, deux tests ont été réalisés. Le premier test était visuel, on a comparé les cartes de PV produites grâce aux sorties du programme original et celle obtenues avec la version FORTRAN 95.

On peut voir ci-dessous un exemple des résultats obtenus. A gauche une carte avec MIMOSA fortran 77 et à droite une carte avec MIMOSA

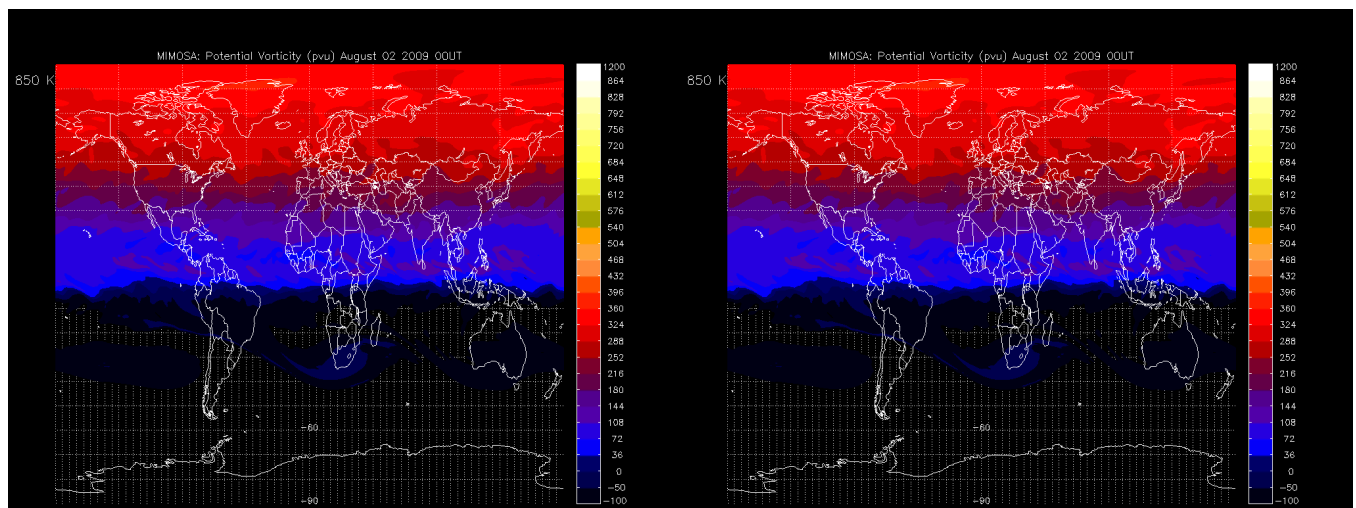


Figure 1: carte de vorticité potentielle sur la surface isentrope $\theta = 850K$, à gauche avec MIMOSA FORTRAN 77, à droite avec MIMOSA FORTRAN 95

On ne détecte pas de différence entre ces deux cartes.

Un second test a consisté à comparer les 584280 valeurs contenues dans chaque sortie du programme original et celles du programme fortran 95. Sur une simulation de dix jours, entre le 01/03/2005 et le 10/03/2005 à un niveau isentrope $\theta=850K$, on observe une différence supérieure à 0.001 et inférieure à 0.01 pour un maximum de 200 valeurs.

Pour une différence supérieure à 0.0001 on trouve environ 40000 valeurs par fichier de sortie soit environ 7%.

II. Passage de MIMOSA en niveau modèle ECMWF

II.1. Intérêt

Le passage de MIMOSA en niveau modèle présente principalement deux intérêts.

Le premier est de pouvoir effectuer des simulations sur des niveaux isentropes supérieurs à $\theta=950K$. En effet quel que soit le type de données ECMWF (opérationnelles, ERA-Interim...), le niveau pression le plus bas est 1 hPa. Les données niveaux modèle permettent de d'atteindre jusqu'à 0.01 hPa comme on peut le voir sur la figure 2. On peut voir aussi que les niveaux modèle permettent de prendre en compte l'orographie. Des simulations MIMOSA pourront donc être réalisées sur l'ensemble de la stratosphère.

Le deuxième avantage est de disposer d'une grille complète de pression. En effet avec la version actuelle de MIMOSA, on dispose d'un vecteur contenant les niveaux de pression et pour chacune de ces pressions, on dispose d'une grille pour la température T , et les composantes du vent u et v . En niveau modèle, on dispose d'une valeur de pression pour chaque point de grille.

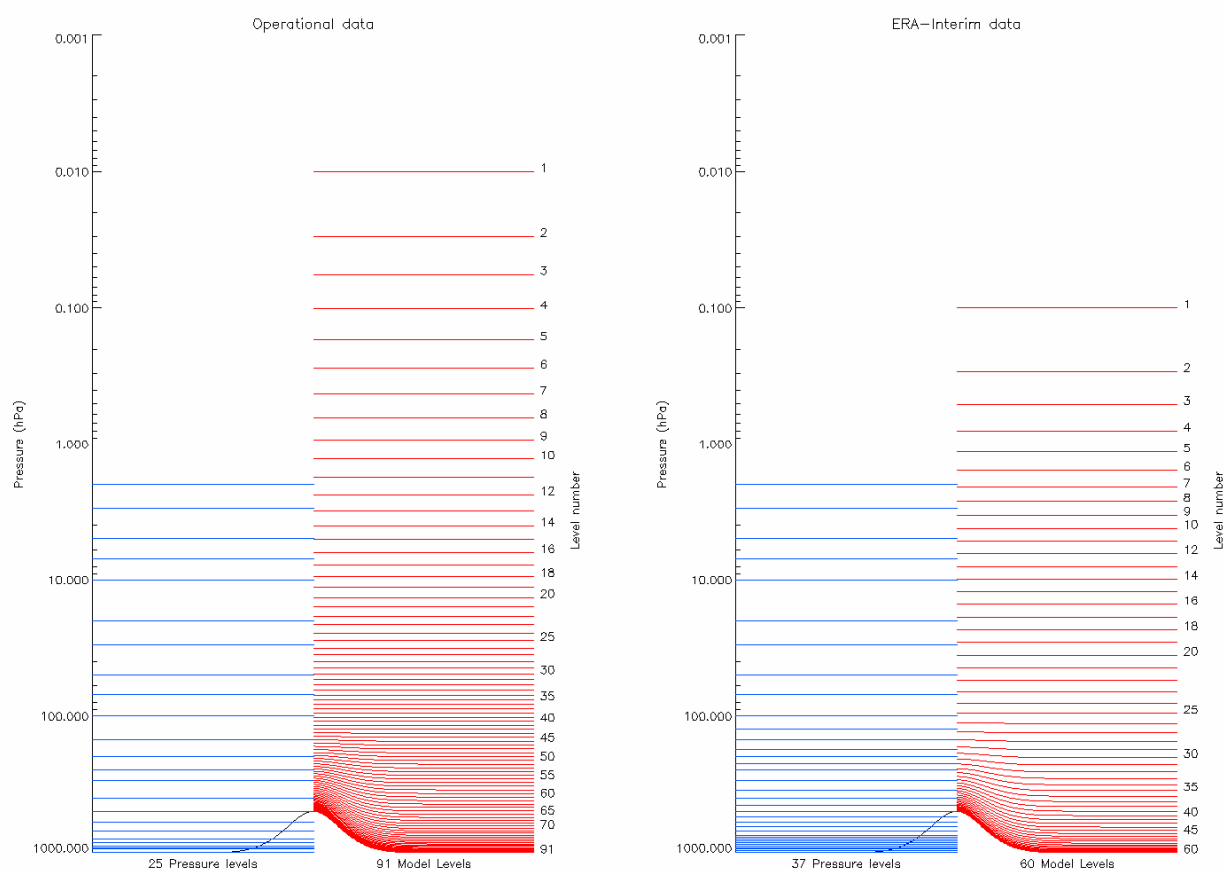


Figure 2: Comparaison entre niveaux modèle et niveaux pression pour les données ECMWF opérationnelles et ré analysées ERA-Interim

II.2. Format des données en niveaux modèle

En niveau pressions, on peut au maximum avoir des données sur 37 niveaux. En niveau modèle, si on utilise des données ECMWF ERA-Interim, les données sont définies sur 60 niveaux, si on utilise les données opérationnelles, elles le sont sur 91 niveaux. Donc utiliser le même format ASCII pour ces données créera des fichiers trop volumineux (un fichier pèserait environ 90 Mo). Il a donc été décidé d'utiliser directement le format GRIB1/2 de l'ECMWF.

En effet pour un fichier d'initialisation de MIMOSA avec une résolution de $1.125^\circ \times 1.125^\circ$ sur 37 niveaux de pression, le fichier pèse environ 38Mo. Un fichier à la même résolution sur 60 niveaux modèle va lui peser au voisinage de 17Mo.

Par contre l'utilisation de ce format va nécessiter l'utilisation de la librairie `grib_api` de l'ECMWF pour lire les données. Néanmoins l'avantage de cette librairie est qu'elle est compatible avec les compilateurs FORTRAN les plus connus, qu'elle est optimisée et maintenue par l'ECMWF. L'installation de cette librairie sera décrite dans le manuel d'utilisation de MIMOSA.

Pour le nom des fichiers, on a gardé la même nomenclature que pour les fichiers en niveaux pression en changeant seulement la l'extension :

- DYYMMDDHH.grib
- Ces fichiers seront stockés de la même manière que les fichiers ECMR. Un dossier GRIB contenant un dossier par année (YYYY). Chaque dossier année contient un dossier par mois (MM) et chaque dossier mois contient tous les fichiers *.grib du mois.

Par exemple le fichier D05030100.grib sera contenu dans le dossier /GRIB/2005/01/.

Un script permettant de récupérer les données sur les serveurs de l'ECMWF sera fourni avec MIMOSA.

II.3. *subroutine readgrib.f95*

Readgrib est la subroutine écrite pour lire les fichiers *.grib. Il a noté que pour pouvoir obtenir une grille de pression, il est nécessaire de lire la grille de pression au sol et les constantes qui permettent de calculer la grille de pression à partir de la pression au sol. L'avantage du format GRIB2 est qu'il contient ces constantes.

Pour cette routine, on assume que la grille à lire est toujours sur l'intervalle $[0^\circ, 360^\circ]$ en longitude et sur l'intervalle $[-90^\circ, 90^\circ]$ en latitude.

- Paramètres d'entrée
 - o **nx** : le nombre de points de grille selon la longitude
 - o **ny** : le nombre de points de grille selon la latitude
 - o **np** : le nombre de niveaux verticaux
 - o **Sisobaric, nomSAap** : chemin + nom du fichier à lire
- Paramètres de sortie
 - o **pl** : le tableau contenant les valeurs lues de pression
 - o **tl** : le tableau contenant les valeurs lues de température
 - o **ul** : le tableau contenant les valeurs lues de vent zonal
 - o **vl** : le tableau contenant les valeurs lues de vent méridien

Les tableaux retournés par readgrib sont directement utilisables dans la subroutine `pvp.f95`.

II.4. subroutine pvp.f95

Cette routine a pour rôle de calculer la vorticité potentielle et d'interpoler les champs u , v , t de l'ECMWF sur le niveau isentrope de la simulation.

Cette subroutine n'a pas nécessité beaucoup de modifications. La seule a été de lui ajouter un paramètre d'entrée, le tableau pl contenant les valeurs de pression et pour les calculs ou la valeur de la pression était utilisé et modifier la valeur de pression utilisée c'est-à-dire :

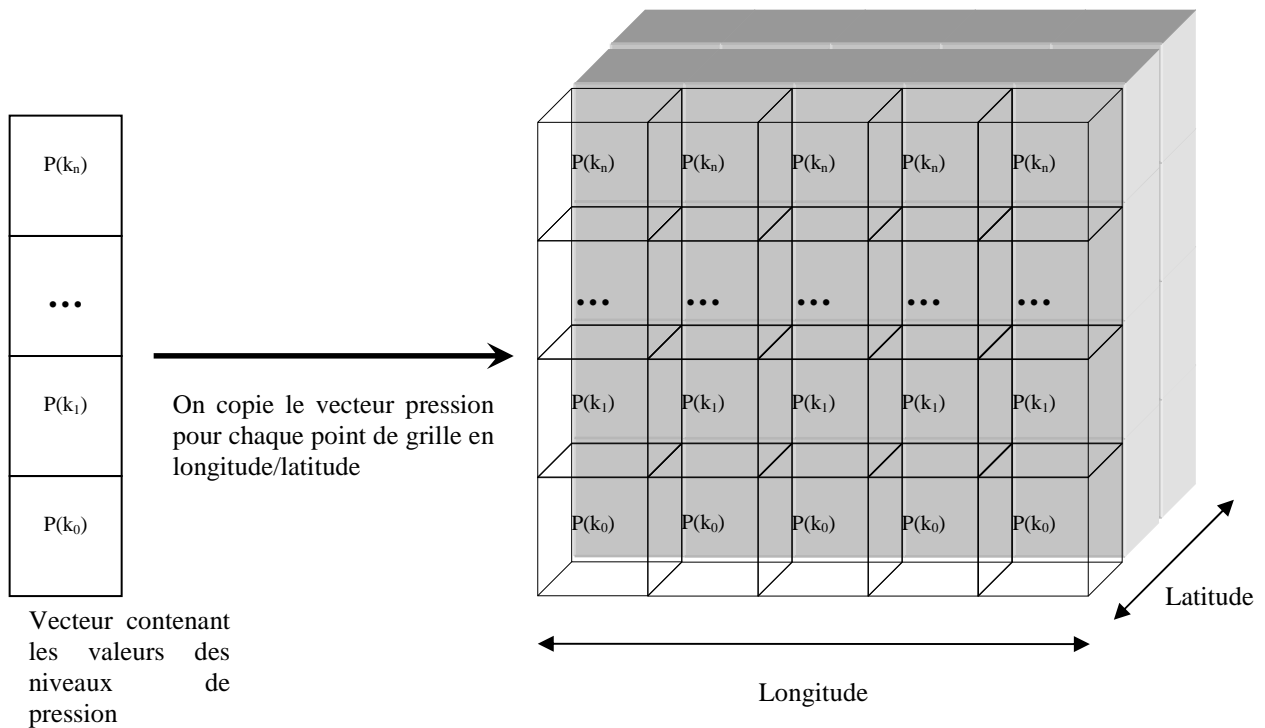
- Remplacer $Pres(k)$ ou k est le niveau de pression
Par
- $pl(i,j,k)$ ou i est l'indice du point en longitude, j l'indice du point en latitude et k le niveau vertical

II.5. programme filament

Filament est le programme principal du modèle MIMOSA. Une des contraintes était que la version compatible avec les niveaux modèle puisse toujours fonctionner avec les données en niveaux pression. En prenant en compte le besoin de ne pas recompiler MIMOSA à chaque changement de configuration ou de données d'entrée, la solution la plus simple est de faire lire à MIMOSA une *namelist* au début de chaque simulation dans laquelle sont décrit les principaux paramètres de la simulations ainsi que les caractéristiques des données d'entrée. Un exemple de *namelist* apparaît en annexe.

Pour que cette version de MIMOSA reste compatible avec les données en niveaux pression, lorsque ces données ont été lues par la subroutine *readecmr*, on crée le tableau pl en copiant le vecteur contenant les niveaux de pression à chaque point de la grille. Ainsi il n'est pas nécessaire d'utiliser plusieurs versions de *pvp* en fonction des données d'entrée. Le schéma ci-dessous décrit la transformation.

Transformation du vecteur de niveaux pression pour pouvoir être utilisé dans la procédure pvp.f95 écrite pour MIMOSA niveaux modèles



Il a aussi été ajouté la possibilité de sortir les champs interpolé de vent zonal u et méridien v sur la grille MIMOSA. Il faut dans ce cas mettre la variable `wind_out = 1`, dans la *namelist output*. Dans ce cas en plus des sorties de champs de vorticité potentielle, on obtiendra un fichier pour le vent zonal et un fichier pour le vent méridien à chaque pas de temps.

Pour l'intégration, des données d'initialisation en niveau pression il a aussi nécessaire d'adapter le nom des fichiers de sortie MIMOSA. En effet, lorsque le modèle est initialisé avec des données en niveau pression, on lançait une simulation sur une surface isentrope de 950K au maximum. Avec les données d'entrée en niveau modèle on peut lancer des simulations sur des surfaces isentropes supérieures à 1000K. Le nom des fichiers passe donc de :

pvgYYMMDDHH.TTT où TTT est la valeur de la surface isentrope

à

pvgYYMMDDHH.TTTT où TTTT est la valeur de la surface isentrope

On présente ci-dessous des exemples de cartes de vorticité potentielle, de température et des champs de vents sur une surface isentrope $\theta = 1500K$.

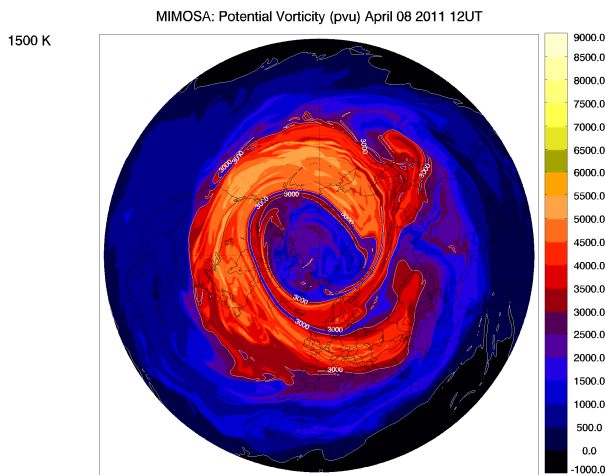


Figure 3 : carte de vorticité potentielle sur la surface
 $\theta = 1500\text{K}$ avec MIMOSA en niveaux modèle

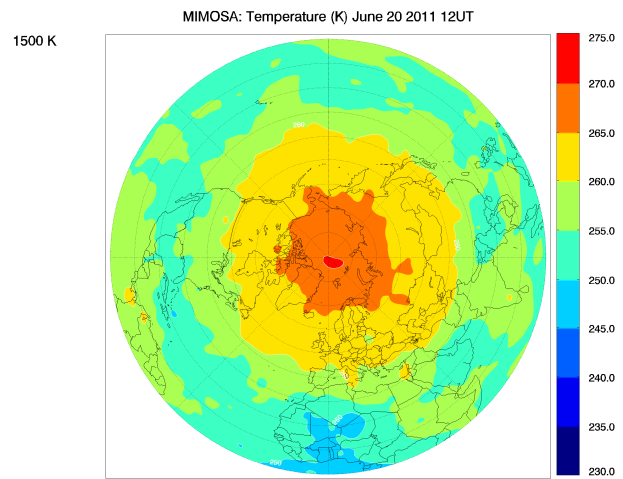


Figure 4 : carte de température sur la surface isentrope
 $\theta = 1500\text{K}$ avec MIMOSA en niveaux modèle

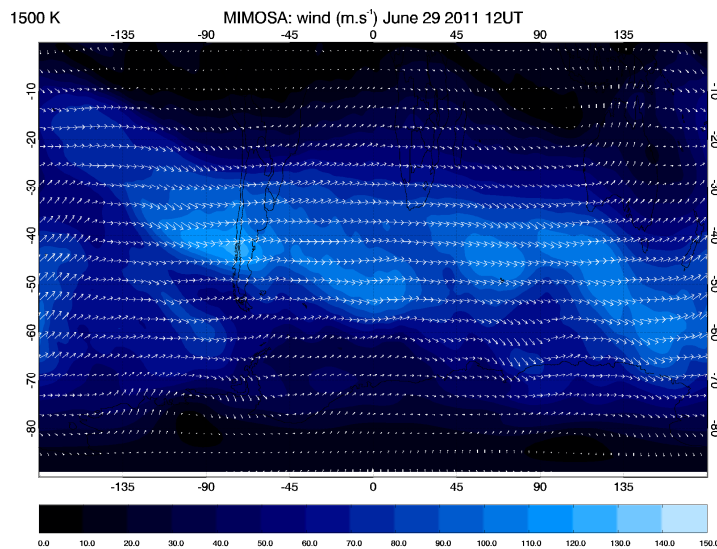


Figure 5 : Champs de vent sur la surface isentrope $\theta = 1500\text{K}$ avec MIMOSA en niveaux modèle

II.6. Validation

Pour la validation des modifications apportées à MIMOSA, nous avons procédé en plusieurs étapes.

- valider que le modèle donne les mêmes résultats quelque soit le compilateur utilisé
- valider les différences entre les résultats données par MIMOSA initialisé avec les données niveaux pression et les données niveaux modèle

II.6.1. Validation des résultats avec différents compilateurs FORTRAN

Pour cette validation, on a utilisé 3 compilateurs différents :

- pgf95 version 11.8 de Portland Group
- ifort version 12.1.0 d'intel
- gfortran le compilateur libre linux
 - o version 4.1.2
 - o version 4.4.0

Les résultats de MIMOSA sont généralement produits avec le compilateur pgf au LPC2E. Nous allons donc comparer les résultats obtenus avec les différents compilateurs à celui-ci. Pour la comparaison, nous avons fait les simulations suivantes avec chacun des compilateurs :

- simulation entre le 01/03/2011 00h et le 10/03/2011 00h
- sur les niveaux isentropes 380 et 850K
- initialisation avec les champs ECMWF ERA-Interim en niveau modèle et pression
- temps de relaxation de 240h
- sorties de champs de vorticité potentielle, de température de vent zonal et de vent méridien

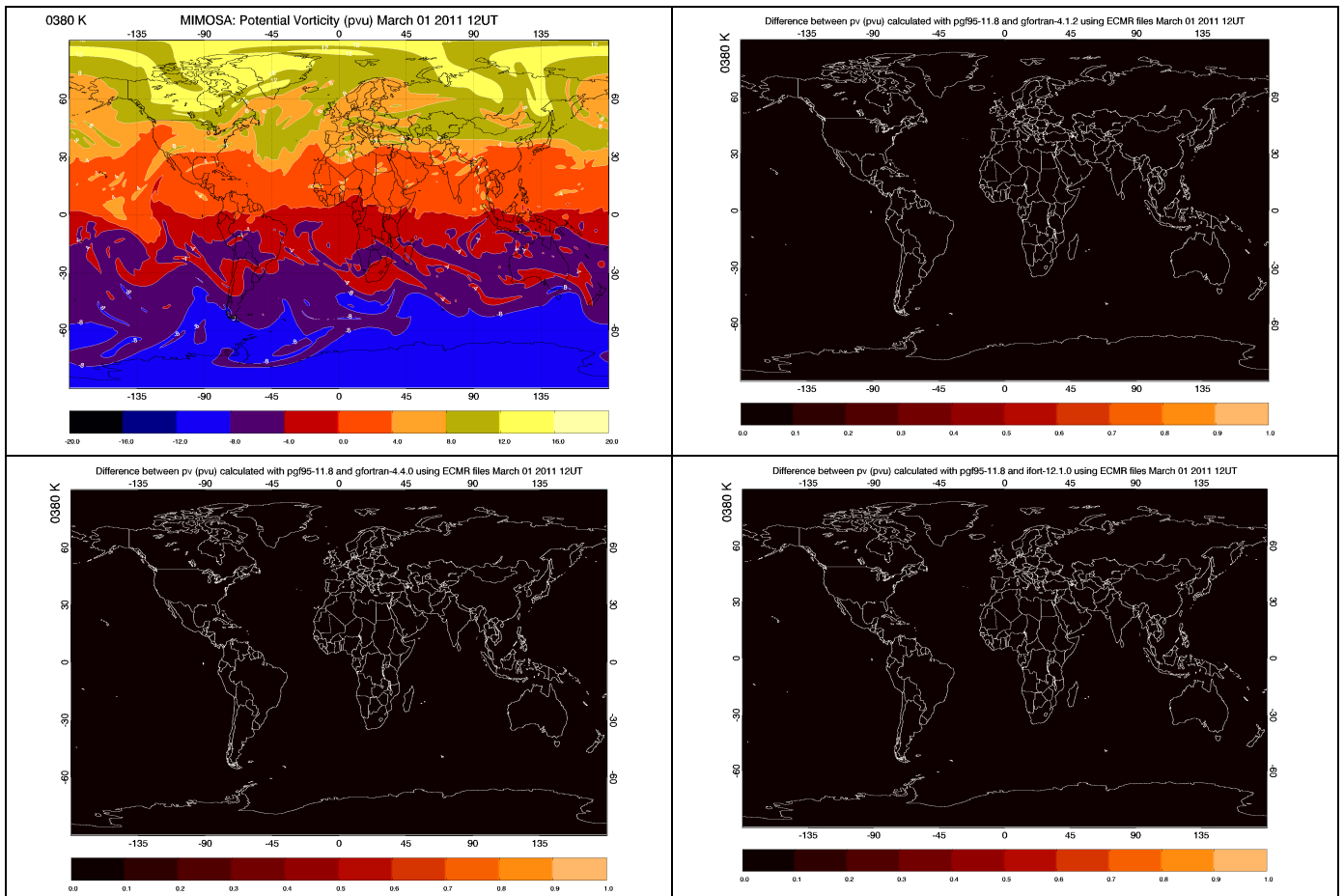


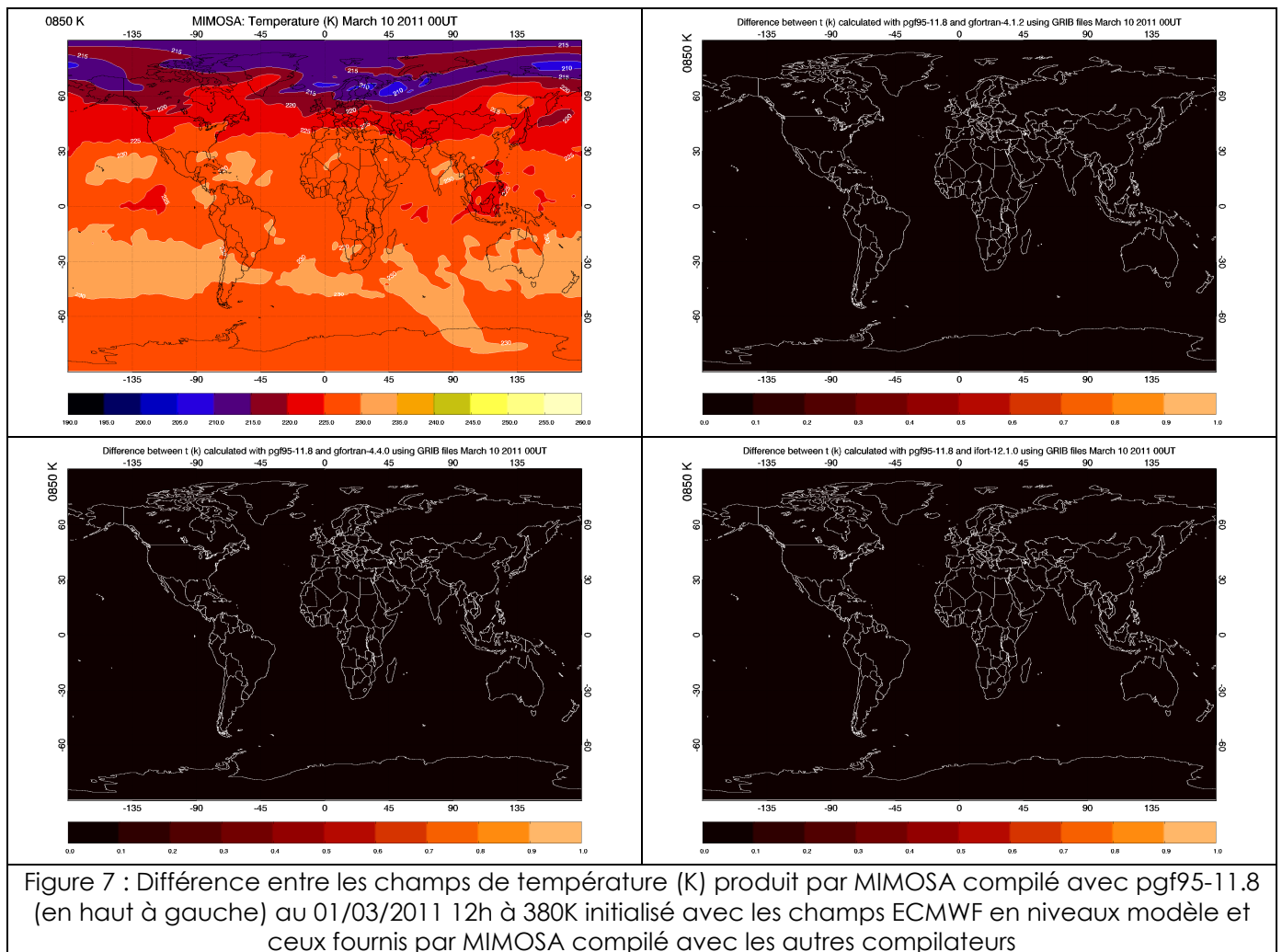
Figure 6 : Différence entre les champs de PV (pvu) produit par MIMOSA compilé avec pgf95-11.8 (en haut à gauche) au 10/03/2011 00h à 380K initialisé avec les champs ECMWF en niveaux pression et ceux fournis par MIMOSA compilé avec les autres compilateurs

On peut voir sur les figures 6 et 7, que la différence maximale entre les champs de obtenu avec MIMOSA compilé avec pgf95-11.8 et ceux obtenus avec le modèle compilé avec les autres compilateurs est au maximum de 0,1.

On observe les mêmes résultats avec les grandeurs température (t), vent zonal (u) et vent méridien (v). Les résultats ne changent pas non plus lors de l'utilisation des champs pression ou modèle pour l'initialisation de MIMOSA.

Il est à noter que l'utilisation de la commande *diff* de linux entre deux fichier binaires de sorties MIMOSA pour lesquels on n'a pas utilisé le même compilateur sont différent.

On peut donc valider l'utilisation du modèle avec ces différents compilateurs.



II.6.2. Validation des simulations initialisées en niveaux modèle

II.6.2.1. Comparaison des résultats avec l'initialisation niveaux pression

Dans cette partie nous allons d'abord présenter les différences entre les résultats obtenus avec MIMOSA lorsqu'il est initialisé avec des champs de l'ECMWF en niveau pression et ceux en niveau modèle.

On utilise la même simulation que pour le cas précédent :

- 10 jours entre le 01/03/2011 00h et le 10/03/2011 00h
- sur les surfaces isentropes 380K et 850K
- initialisation avec les champs ECMWF ERA-Interim en niveau pression et modèle
- 240 h de temps de relaxation
- compilation du modèle avec pgf95 version 11.8

Dans les figures 8 et 9, on compare on compare les champs de température et de vorticité potentielle produits par MIMOSA initialisé avec les champs ECMWF en niveaux pression et ceux en niveau modèle sur la surface isentrope 380K. Sur les figures 10 et 11 on réalise la même comparaison mais sur la surface isentrope 850K.

On peut sur les figures que l'on obtient des valeurs pour les champs de température et de vorticité potentielle qui sont cohérentes avec les résultats obtenus avec les

simulations en niveaux pression. On remarque des différences principalement dans la forme des structures.

Pour les champs de température, on remarque que les différences sont plus prononcées dans la zone située entre -30°N et 30°N de latitude. En début de simulation, la différence atteint jusqu'à 20K. La différence est moins marquée à 850K avec une différence maximale de 10K.

Les champs de vorticité potentielle présente une différence plus marquée à 380K, les différences sont réparties sur l'ensemble du globe. A 850K, les différences sont beaucoup plus marquées au niveau du pôle Nord. Ce résultats semble plutôt cohérent avec la période dans laquelle se situe la simulation qui est en fin d'hiver arctique.

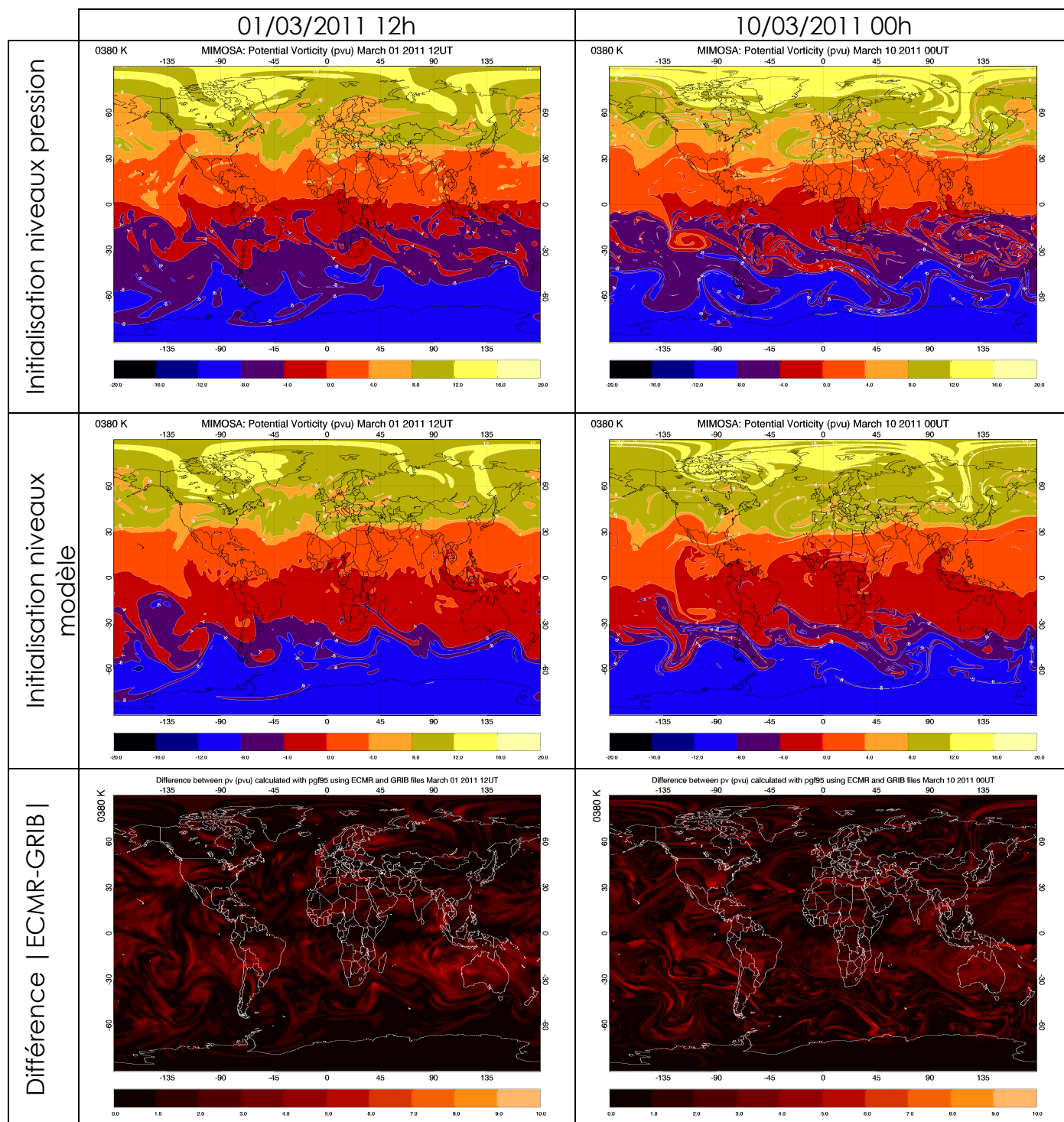
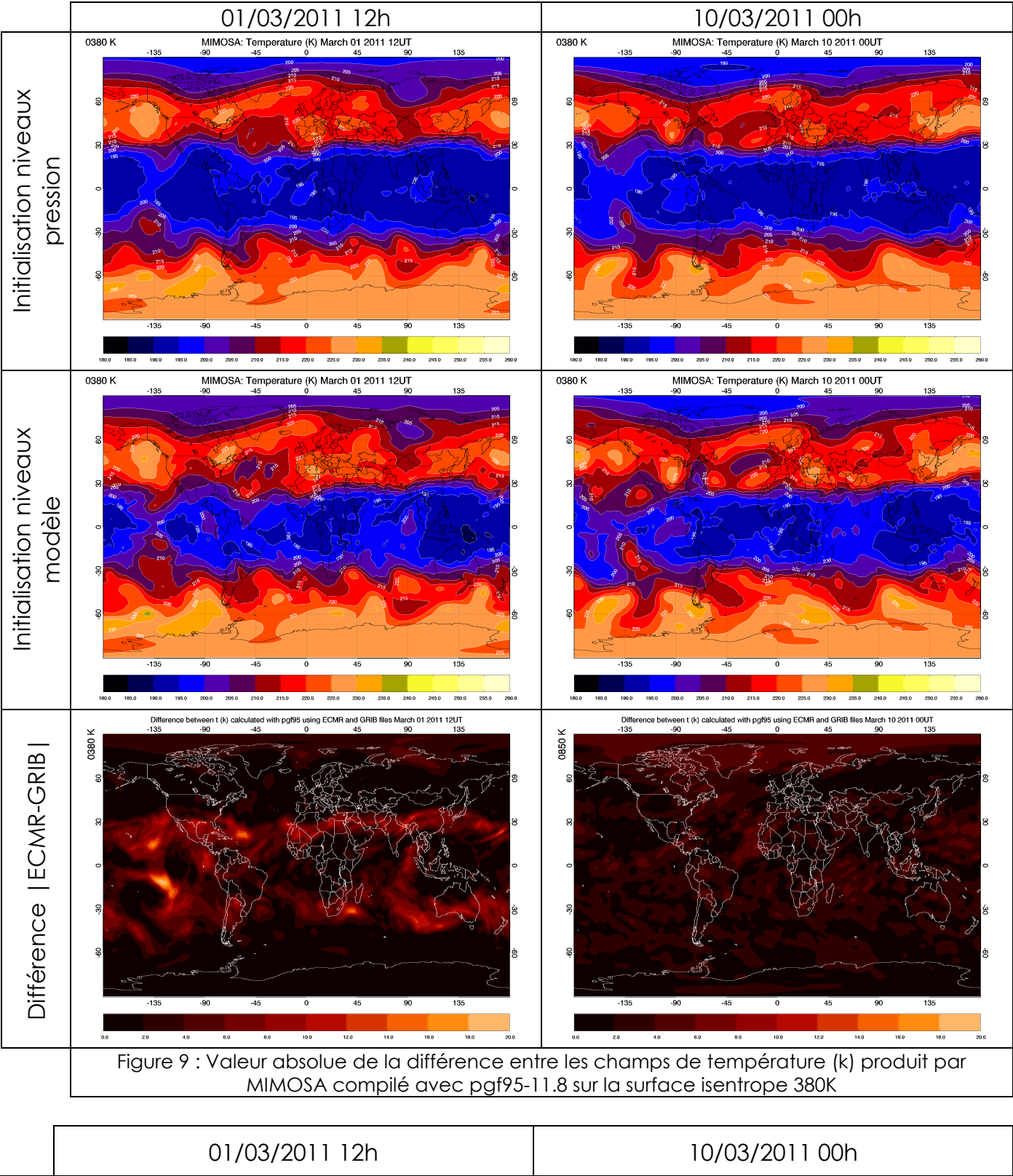


Figure 8 : Valeur absolue de la différence entre les champs de PV (pvu) produit par MIMOSA compilé avec pgf95-11.8 Sur la surface isentrope 380K



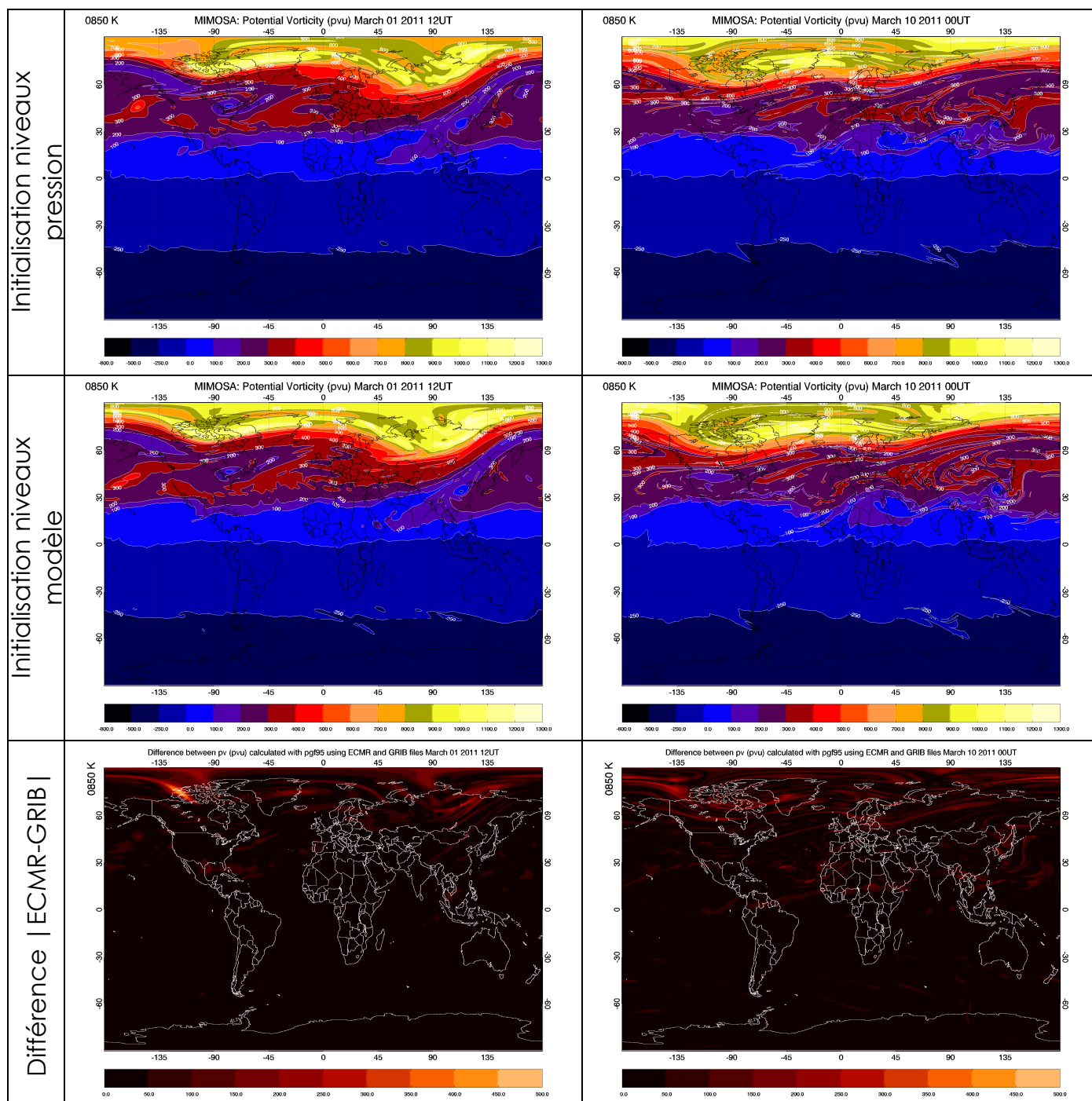
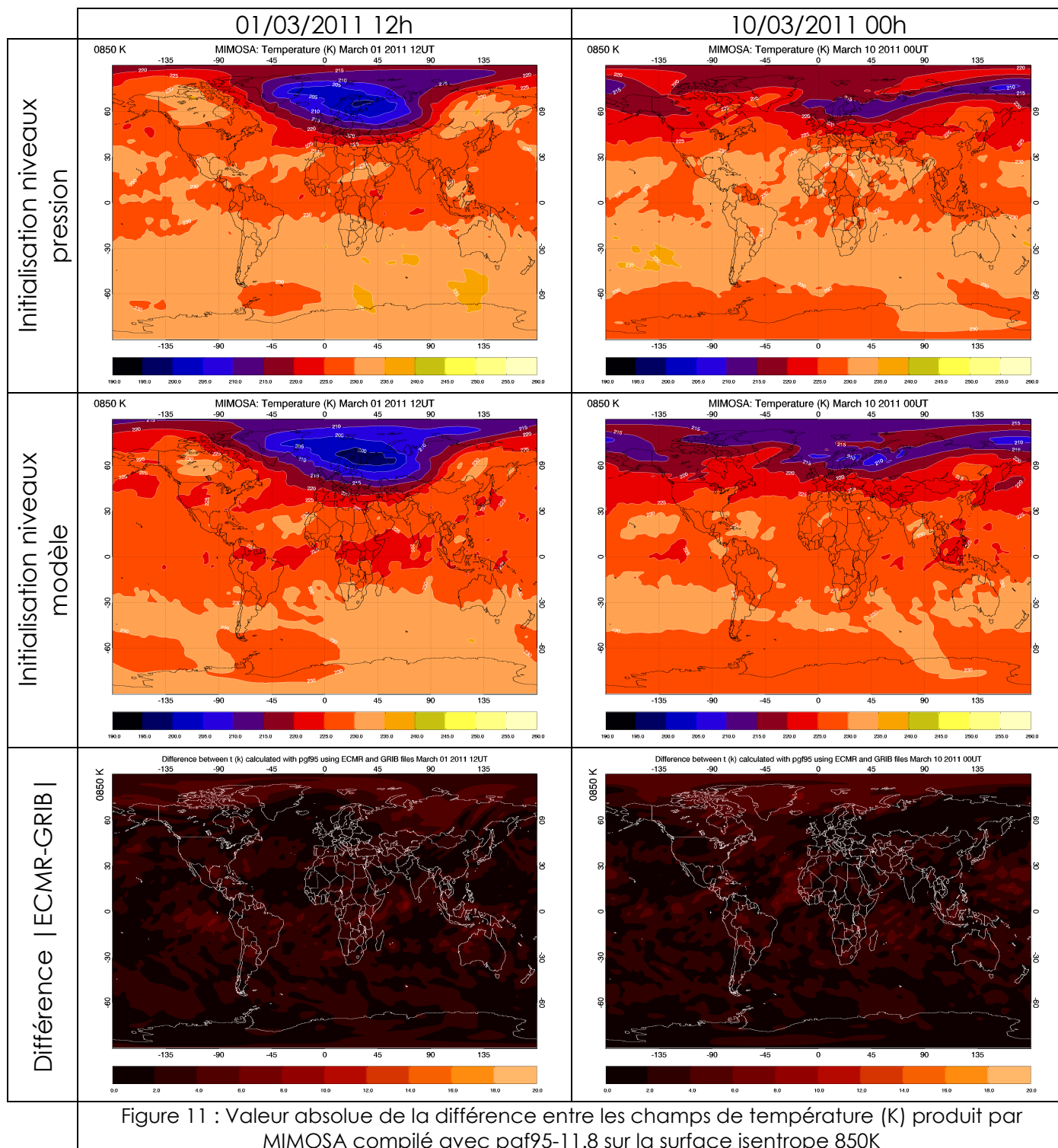


Figure 10 : Valeur absolue de la différence entre les champs de pv (pvu) produit par MIMOSA compilé avec pgf95-11.8 sur la surface isentrope 850K



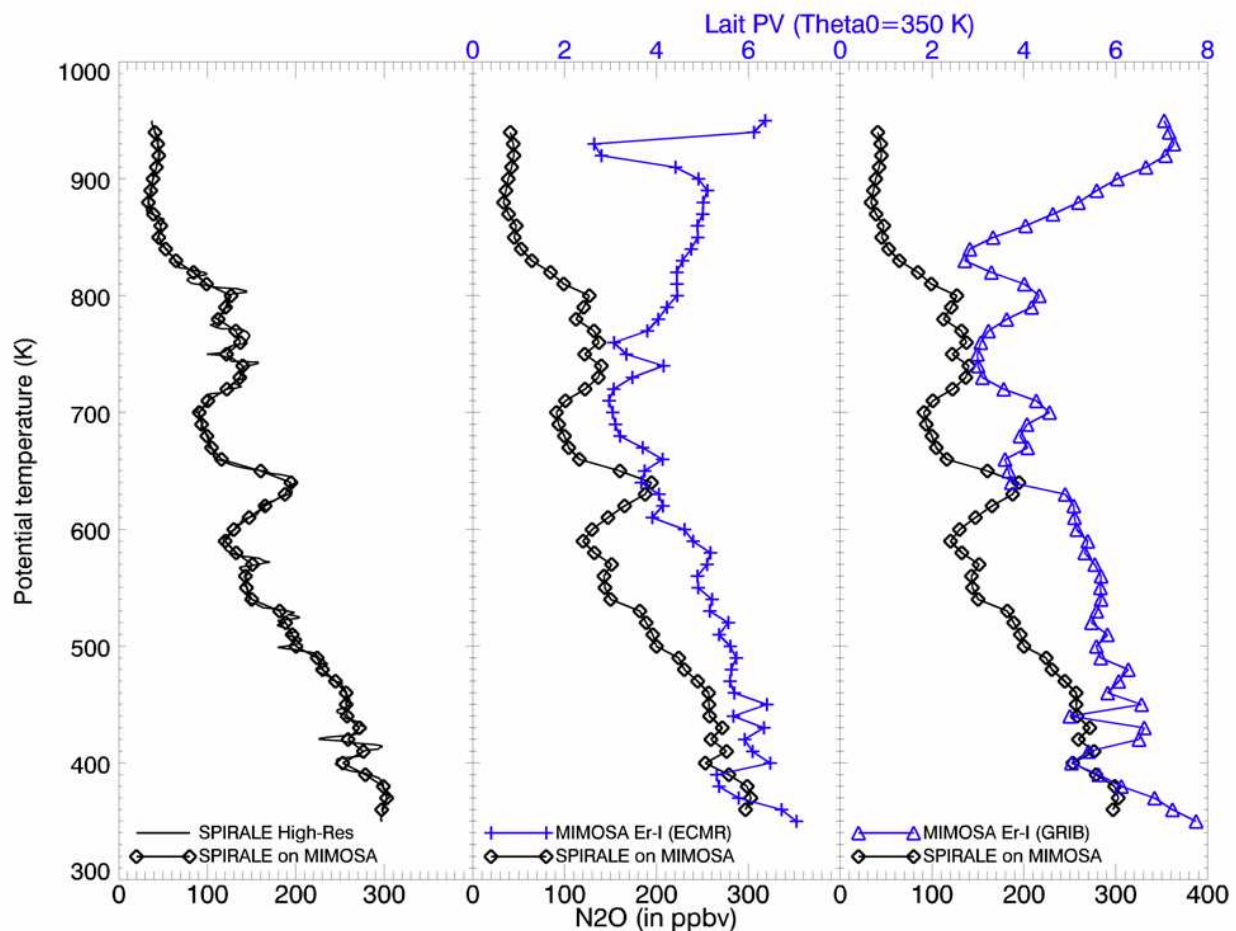
II.6.2.2. Utilisation lors de la campagne ENRICHED

Dans le but de valider les modifications apportées au modèle MIMOSA, il a été utilisé pour produire des résultats de simulations tous les jours dans le cadre du projet ENRICHED pendant la campagne de vol ballon (du 13 janvier au 23 Avril 2011). Le modèle était initialisé de la manière suivante :

- simulations de 13 jours
 - o [J-3, J] : champs ECMWF analysé
 - o [J, J+10] : champs ECMWF prévisionnels
- Simulations sur 14 niveaux isentropes
 - o $\theta = 340, 380, 400$ to 850 par $50K, 950, 1000K$
- temps de relaxation 240h
- compilé avec pgf95 version 10.8

Les résultats obtenus ont semblé tout à fait cohérents avec ceux observés durant la campagne notamment au niveau de la position du vortex polaire arctique.

II.6.2.3. Comparaison avec les mesures de l'instrument SPIRALE



III. Bilan de l'évolution

III.1. Performances

Les modifications apportées au modèle ont permis d'en améliorer les performances. Les tests ont été réalisés sur le serveur du calcul du LPC2E (serveur bi-quad cœurs AMD opteron 2,5 GHz, 32 Go de Ram).

Avant les modifications, le modèle ne pouvait être compilé qu'avec pgf77 et il fallait environ 2 min 10 s pour une simulation de 10 jours sur deux hémisphères en ne sortant que les champs de vorticité potentielle.

Les tests de performances ont été réalisés pour les conditions suivantes de simulations :

- 10 jours
- un niveau isentrope
- toutes sorties activées
 - o champs de PV
 - o champs de température
 - o champs de vent zonal et méridien

Type d'entrées	Compilateur	1 hémisphère	2 hémisphères
ECMR	gfortran 4.1.2	1 min 54 s	2 min 22 s
	gfortran 4.4.0	1 min 53 s	2 min 32 s
	pgf95 11.8	1 min 23 s	1 min 48 s
	ifort 12.1.0	0 min 42 s	1 min 06 s
GRIB	gfortran 4.1.2	1 min 02 s	1 min 38 s
	gfortran 4.4.0	1 min 01 s	1 min 38 s
	pgf 95 11.8	0 min 32 s	0 min 58 s
	ifort 12.1.0	0 min 27 s	0 min 51 s

Le tableau ci-dessus permet de voir le gain de performance. Avec les deux versions du compilateur open source, on obtient pour les simulations initialisées avec les champs niveau pression un temps équivalent pour une simulation sur deux hémisphères excepté que l'on a les sorties de vent et de température en plus.

Au vu des temps obtenus pour les simulations initialisées en niveaux modèle, le gain de temps de calcul est principalement dû à l'utilisation de la bibliothèque grib_api de l'ECMWF pour la lecture des champs.

Le cas du compilateur INTEL ifort est un peu particulier. Malgré le fait qu'il soit plutôt optimisé pour des machines à base de processeurs INTEL, il reste le plus performant notamment pour l'initialisation avec les champs niveaux pression. Il est vraisemblablement plus rapide pour les simulations avec les champs niveaux modèle car la bibliothèque grib_api a été compilée totalement avec les compilateurs INTEL (icc et ifort) ce qui n'est pas le cas pour le compilateur pgf dont nous ne disposons pas du compilateur C.

III.2. Utilisation

Ces développements ont permis de simplifier l'utilisation du modèle MIMOSA. Grâce à l'utilisation des *namelist*, il n'est plus nécessaire de recompiler le programme à chaque changement de configuration.

Le programme peut maintenant être installé sur une machine et être utilisé comme un simple exécutable.

IV. Développement d'outils

IV.1. MakeFile

Un makefile spécifique à chaque compilateur avec lequel MIMOSA a été testé a été fourni.

Avant de pouvoir utiliser ces makefile, il est nécessaire d'éditer le fichier pour fournir trois chemins vers les bibliothèques grib_api et jasper :

- **INCPATH** : cette option permet de fournir le chemin vers le module décrivant les sous-routines de la bibliothèque grib_api
- **GRIBPATH** : cette option permet de fournir le chemin vers la bibliothèque grib_api
- **JASPERPATH** : cette option permet de fournir le chemin vers la bibliothèque jasper.

Si ces bibliothèques ont été installées par l'intermédiaire du gestionnaire de paquet de votre distribution linux, ces champs peuvent être laissés vides.

Il est aussi nécessaire de configurer l'option **DEBUG** qui permet de choisir si on active ou non les options pour le débogueur du compilateur. Cette variable accepte les valeurs **yes** ou **no**.

Il est ensuite possible d'appeler le makefile avec trois options :

- **make all** permet de compiler le programme
- **make clean** permet de supprimer les objets (*.o) et les modules (*.mod) générés à la compilation
- **make veryclean** permet de supprimer l'exécutable en plus des objets et des modules.

IV.2. Script ECMWF

Deux scripts de récupération des champs ECMWF pour l'initialisation de MIMOSA ont été écrits.

- **mimosa_ecmwf_ECMR.sh** permet la préparation des champs en niveau pression.
- **mimosa_ecmwf_GRIB.sh** permet la préparation des champs en niveau modèle.

Ces deux scripts sont compatibles avec le format d'archivage GRIB2. Ils permettent la récupération de données sur plusieurs mois voire plusieurs années.

Des options de configuration sont aussi disponibles pour utiliser le système de transfert de fichier de l'ECMWF ECTTRANS.

L'utilisation de ces scripts est décrite dans le manuel de MIMOSA.

IV.3. Script mimosa.sh

Ce script développé en BASH permet pour une même période de simulation de répéter les simulations pour plusieurs surfaces isentropes. Il permet aussi de générer les listes de fichiers nécessaires aux programmes de traçage IDL.

IV.3. Outils graphiques IDL

La plupart des outils graphiques développés pour MIMOSA utilise la librairie graphique coyote. Il est donc nécessaire de la télécharger et de l'inclure dans son PATH IDL avant de pouvoir utiliser ces programmes. La procédure permettant l'ajout de la barre de couleur a été écrite par Sébastien Masson de l'IPSL.

Tous les graphes présents dans ce document ont été produits en utilisant les programmes ci-dessous. Leur utilisation est décrite dans le manuel d'utilisation de MIMOSA.

IV.3.1. Procédure de lecture des sorties binaires de MIMOSA

La procédure READ_MIMOSA_BIN, permet la lecture des fichiers de sorties MIMOSA. Cette routine permet de lire toutes les sorties binaires MIMOSA, que ce soit les champs de vorticité potentielle, les champs de température ou les champs de vent zonal ou méridien.

Cette procédure prend en entrée le nom complet du fichier à lire et accepte les options suivantes :

- HEADER : pour retourner l'entête complet du fichier
- DATA : pour retourner les données contenues dans le fichier
- DIMX : pour retourner le nombre de points de grille en longitude
- DIMY : pour retourner le nombre de points de grille en latitude

Cette procédure est utilisée par toutes les routines de traçage qui vont être décrite dans la partie suivante.

IV.3.2. Procédures de traçage

Trois procédures de traçage ont été créées. Elles fonctionnent toutes sur le même principe. On fournit en entrée le dossier contenant les fichiers à tracer, le fichier contenant le nombre et la liste des fichiers à lire et le nom du dossier de sortie. Les procédures retournent des cartes sur lesquelles sont tracées les contours de la grandeur accompagnés d'une barre de couleur pour chaque pas de temps.

- TRACE_MIMOSA_PV
 - o Traçage des champs de vorticité potentielle
- TRACE_MIMOSA_TEMP
 - o Traçage des champs de température
- TRACE_MIMOSA_VENT
 - o Traçage des champs $\sqrt{u^2 + v^2}$
 - o Traçage des champs de vecteurs indiquant la direction du champ de vent
- TRACE_3D_VORTEX
 - o Permet à partir de simulations MIMOSA sur plusieurs surfaces isentropes de représenter une valeur de vorticité potentielle sous forme de volume. Pour cela la vorticité potentielle est normalisée grâce à la formule de Lait (Lait, 1993)

IV.4. Script de création d'animations

Ce script développé en bash permet de générer des animations à partir des graphes produits grâce aux programmes IDL décrits ci-dessus.

Il utilise les programmes imageMagick pour convertir les images au format PNG en JPG et ffmpeg pour convertir les images JPG générées en vidéos encodées en x264.

Son utilisation est décrite dans le manuel de MIMOSA. On peut voir des exemples des vidéos produites à l'adresse suivante :

<http://lpc2e.cnrs-orleans.fr/~enriched/news.html#news111130>

Bibliographie

Hauchecorne, A., Gaudin, S., Marchand, M., Heese, B., and Souprayen, C. (2002), Quantification of the transport of chemical constituents from the polar vortex to midlatitudes in the lower stratosphere using the high-resolution advection model MIMOSA and effective diffusivity, *J. Geophys. Res.*, Vol. 107, No. D20, 8289, doi : 10.1029/2001JD000491.

Lait, Leslie R., (1994), An alternative form for potential vorticity, *Journal of atmospheric sciences*, vol. 51, Issue 12, pp. 1754-1759, doi: 10.1175/1520-0469(1994)051<1754:AAFFPV>2.0.CO;2