ESA contract
Martian Environmental Model

Ref : ESA 11369/95/NL/JG(SC)

WP III deliverable :

User Manual for the Martian Atmospheric General Circulation Model

Christophe HOURDIN and Frédéric HOURDIN
(LMD)

Approved by Olivier Talagrand January 21, 1999

Draft

Contents

1 introduction **4**

I Formulations physiques et numériques **5**

2 Discrétisation des équations de la dynamique **6**

3 Les paramétrisation physique du modèle Martien **10**
3.1 Radiation 10
3.2 Vertical turbulent mixing 11
3.3 Convective adjustment 12
3.4 Surface processes 13
3.5 CO2 condensation-sublimation 13

4 Interface Physique/dynamique **15**

5 Entrées/Sorties **18**
5.1 format DRS 18
5.1.1 L'éditeur de fichiers DRS: drsed 18
5.1.2 Visualisation graphique des fichiers DRS sous GrAds 19
5.1.2.1 Le fichier de données (.dat) 20
5.1.2.2 Le fichier de contrôle (.ctl ou .head) 20
5.2 Entrées 21
5.2.1 run.def 21
5.2.2 callphys.def 23
5.2.3 esasig.def 24
5.2.4 les fichiers d'initialisation: start et startfi 24
5.3 Sorties 30
5.3.1 fichiers DRS de redemarrage restart et restartfi 30
5.3.2 fichiers DRS histmoy 30
5.3.3 fichiers DRS diagfi 31
5.3.4 fichiers "anglais" cumul/stats 32

6 Organisation informatique et compilation **33**
6.1 Orgnisation des fichiers sources du modèle 33
6.2 Organigramme du modèle 34
6.2.1 La partie dynamique: gcm.F 34
6.2.2 La partie physique: physiq.F 38
6.2.3 etats initiaux: newstart.F 39
6.2.4 banque d'etats initiaux: start2archive.F 41
6.3 Compilation du modèle 41

II Faire tourner le modèle 44

7 première simulation 46
7.1 Installer le modèle 46
7.2 Compiler le modèle 48
7.3 Les fichiers d'entrée (état initial) 48
7.4 Faire tourner le modèle 48
7.5 Visualisation des fichiers de sortie 48
7.5.1 Visualisation des fichiers histmoy, diagfi, histins... 48
7.5.2 Visualisation des fichiers cumul/stats 49
7.6 relancer la simulation 50

8 Changer les conditions initiales 51

Chapter 1

introduction

Ce document est à la fois une documentation scientifique et un manuel d'utilisation du modèle de circulation générale de l'atmosphère Martienne développé conjointement par le Laboratoire de Météorologie Dynamique du CNRS à Paris et le Atmospheric and Oceanic Planetary Physics à Oxford.

Il contient tout d'abord une description du modèle séparée en deux parties principales. Ces parties correspondent à deux parties indépendantes du modèle:

Le code hydrodynamique, qui intègre dans le temps et sur la sphère les équations de la mécanique des fluides.

Un jeu de paramétirsations physiques, avec notamment le calcul du transfert radiatif dans l'atmosphère et du mélange turbulent dans la couche limite.

Chaque paramétrisation est décrite à la fois sur le plan physique et sur le plan numérique.

Le document contient ensuite la description de l'interface entre les deux parties indépendantes du modèle (interface physique/dynamique), la description des Entrées-sorties du modèle à savoir notamment: en entrée les fichiers nécessaires à l'initisalisation du modèle (état de l'atmosphère à l'instant $t0$ plus un jeu de conditions aux limites) et en sortie, les "fichiers hitoires", archives de l'hitoire de l'écoulement atmosphérique telle qu'elle est simulée par le modèle, les "fichiers diagfi", ..., les "fichiers stats" , moyennes diurnes ... Egalement les moyens d'éditer ou de visualiser ces fichiers (l'éditeur "drsed" ou le logiciel graphique "grads").

La partie suivante contient un manuel d'utilisation informatique pour la compilation et pour l'execution du modèle.

Enfin, pour un **premier contact avec le modèle** (voir section 7), une dernière partie permet de guider l'utilisateur dans une première simulation (choix des conditions initiales et des paramètres, et visualisation des fichiers de sorties).

Part I

Formulations physiques et numériques

Chapter 2

Discrétisation des équations de la dynamique

Robert Sadourny, Phu Le Van, Frédéric Hourdin
Laboratoire de Météorologie Dynamique du CNRS
Ecole Normale Supérieure
24 rue Lhomond
75231 PARIS cedex 05
FRANCE

Le modèle climatique du LMD est bâti, comme tous les modèles de circulation générale atmosphérique, sur la résolution numérique des équations primitives de la météorologie décrites dans de nombreux ouvrages (2). L'analyse présentée ici a été menée sur la nouvelle version de la dynamique du LMD écrite par Phu Le Van (7) sur une formulation de Robert Sadourny. Cette formulation diffère de l'ancienne essentiellement par deux points: dans la nouvelle formulation, la répartition des points en longitude et en latitude peut être changée arbitrairement. L'autre modification porte sur la répartition des points aux pôles[1].

La coordonnée verticale du modèle est la pression normalisée par sa valeur à la surface: $\sigma = p/p_s$. On utilise en fait σ aux niveaux inter-couches et $s = \sigma^\kappa$ au milieu des couches. On note X et Y les coordonnées horizontales:

X (resp. Y) est une fonction biunivoque de la longitude λ (resp. de la latitude ϕ). Ces deux fonctions peuvent être choisies de façon arbitraire dans le modèle LMDZ ce qui permet d'effectuer un zoom sur une région du globe particulière. Une grille de ce type est montrée sur la Figure 2.1. Les variables scalaires (température potentielle $\theta = c_p T/{p_s}^\kappa$, géopotentiel Φ et pression de surface p_s) sont évaluées aux points correspondant à des couples de valeurs entières $(X, Y) = (i, j)$. Les variables dynamiques sont décalées par rapport aux variables scalaires en utilisant une grille C dans la définition de Arakawa (?): le vent zonal est calculé aux points $(X, Y) = (i + 1/2, j)$ et le vent méridien aux points $(X, Y) = (i, j + 1/2)$. La disposition des variables sur la grille est illustrée sur la Figure 2.2.

On utilise en fait les composantes covariantes ($\tilde{u}$ et $\tilde{v}$) et contravariantes ($\tilde{\tilde{u}}$ et $\tilde{\tilde{v}}$) du vent définies par

$$\begin{array}{lllll} \tilde{u} = c_u u & \text{et} & \tilde{\tilde{u}} = u/c_u & \text{avec} & c_u = a\cos\phi\,(d\lambda/dX) \\ \tilde{v} = c_v v & \text{et} & \tilde{\tilde{v}} = v/c_v & \text{avec} & c_v = a\,(d\phi/dY) \end{array} \tag{2.1}$$

où u et v sont les composantes physiques du vecteur vent horizontal. On introduit également:

[1] Aux pôles sont calculés: le vent méridien dans l'ancienne formulation et les variables scalaires dans la nouvelle.

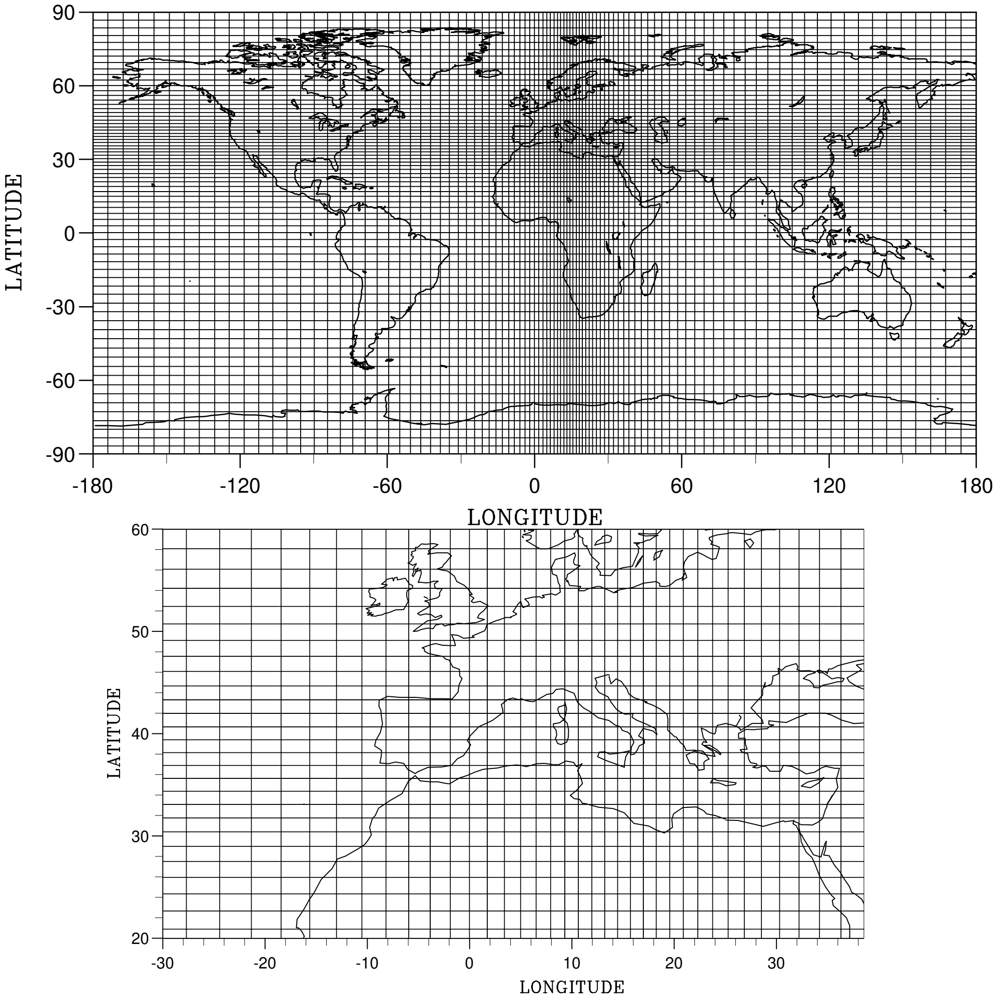

Figure 2.1: Grille obtenue avec 96 points en longitude et 73 en latitude et un zoom d'un facteur 3 centré sur la méditérannée (grille utilisée au laboratoire par Ali Harzallah)

.

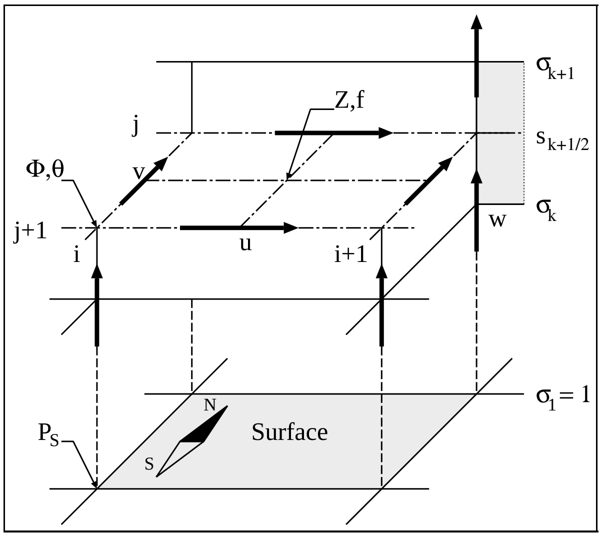

Figure 2.2: Disposition des variables dans la grille du LMD

la pression extensive:

$\tilde{p}_s$ (pression au sol multipliée par l'aire de la maille).

les trois composantes du flux de masse:

$$U = \overline{\tilde{p}_s}^X \tilde{\tilde{u}},\ V = \overline{\tilde{p}_s}^Y \tilde{\tilde{v}} \text{ et } W = \tilde{p}_s \dot{\sigma} \text{ avec } \dot{\sigma} = \frac{d\sigma}{dt} \tag{2.2}$$

le facteur de Coriolis multiplié par l'aire de la maille:

$f = 2\Omega \sin\phi c_u c_v$
où Ω est la vitesse de rotation de la planète.

la vorticité potentielle absolue:

$$Z = \frac{\mathcal{F}\left(\delta_X \tilde{v} - \delta_Y \tilde{u}\right) + f}{\overline{\tilde{p}_s}^{X,Y}} \tag{2.3}$$

l'énergie cinétique

$$K = \frac{1}{2}\left(\overline{\tilde{u}\tilde{\tilde{u}}}^X + \overline{\tilde{v}\tilde{\tilde{v}}}^Y\right) \tag{2.4}$$

La notation δX signifie simplement qu'on effectue la différence entre deux points consécutifs suivant la direction X. La notation $\overline{a}^X$ signifie qu'on prend la moyenne arithmétique de la quantité a suivant la direction X. $\mathcal{F}$ est un filtre longitudinale appliqué dans les régions polaires. Les équations discrétisées sont écrites sous la forme suivante:

équations du mouvement:

$$\frac{\partial \tilde{u}}{\partial t} - \overline{Z}^Y \overline{V}^{X,Y} + \delta_X \mathcal{F}\left(\Phi + K\right) + s\overline{\theta}^X \delta_X \mathcal{F}\left({p_s}^\kappa\right) - \frac{\overline{\tilde{u}_a}^{Y,Y} \delta_Z \overline{W}^X}{\overline{\tilde{p}_s}^X \delta_Z \sigma} + \frac{\delta_Z \left(\overline{W}^X \overline{\tilde{u}_a}^Z\right)}{\overline{\tilde{p}_s}^X \delta_Z \sigma} = S_{\tilde{u}} \tag{2.5}$$

oú $\tilde{u}_a$ est la composante zonale covariante du vecteur vent absolu: $\tilde{u}_a = \tilde{u} + c_u a\Omega\cos\phi$ et

$$\frac{\partial\tilde{v}}{\partial t} + \overline{Z}^X\overline{U}^{X,Y} + \delta_Y\mathcal{F}\left(\Phi + K\right) + s\overline{\theta}^Y\delta_Y\mathcal{F}\left({p_s}^\kappa\right) - \frac{\overline{\tilde{v}}^{X,X}\delta_Z\overline{W}^Y}{\overline{\tilde{p}_s}^Y\delta_Z\sigma} + \frac{\delta_Z\left(\overline{W}^Y\overline{\tilde{v}}^Z\right)}{\overline{\tilde{p}_s}^X\delta_Z\sigma} = S_{\tilde{v}} \tag{2.6}$$

équation thermodynamique:

$$\frac{\partial\left(\tilde{p}_s\theta\right)}{\partial t} + \mathcal{F}\left[\delta_X\left(\overline{\theta}^X U\right) + \delta_Y\left(\overline{\theta}^Y V\right)\right] + \frac{\delta_Z\left(\overline{\theta}^Z W\right)}{\delta_Z\sigma} = S_\theta \tag{2.7}$$

équation hydrostatique:

$$\delta_Z\Phi = -p_s^\kappa\overline{\theta}^z\delta_Z s \tag{2.8}$$

équations de continuité:

$$\frac{\partial p_s}{\partial t} = \mathcal{F}\left[\sum_z\delta_Z\sigma\left(\delta_X U + \delta_Y V\right)\right] \tag{2.9}$$

$$\delta_Z W = -\delta_Z\sigma\left[\mathcal{F}\left(\delta_X U + \delta_Y V\right) + \frac{\partial p_s}{\partial t}\right] \tag{2.10}$$

On a noté S les termes sources dans les différentes équations. Dans ces termes sources, on distingue 1) d'une part les paramétrisations physiques mentionnées plus haut et qui font intervenir pour une maille donnée du modèle, tous les points situés sur une même verticale mais ceux-là seulement; 2) les opérateurs de dissipation horizontales, scencés rendre compte des échanges entre échelles explicitement représentées dans le modèle et échelles sous-mailles. Ces opérateurs ont la structure de Laplaciens agissant sur des plans horizontaux c'est à dire qu'il font intervenir un voisin de chaque côté dans les deux directions horizontales. Cet opérateur est généralement itéré pour le rendre plus selectif en échelle (plus on itère un laplacien et plus sont effet sur les petites échelles devient important relativement).

Chapter 3

Les paramétrisation physique du modèle Martien

3.1 Radiation

The radiative transfer code is adapted from the one developed in the context of terrestrial modeling by Fouquart and Bonnel (1980) for solar radiation and Morcrette *et al.* (1986) for thermal radiation. This code, originally developed for the climate GCM of Laboratoire de Météorologie Dynamique, has since been included by Morcrette in the operational model of the European Centre for Medium-Range Weather Forecasts (ECMWF).

For thermal radiation,

effects of both atmospheric carbon dioxide and dust are included. Particular attention was given to the parametrization of absorption by the CO_2 15μm band with inclusion of Doppler effect (3). Cooling rates computations were carefully validated by comparison to line-by-line integrations. The model, in which Doppler effects are introduced, is very accurate up to 70 km. The thermal spectrum is divided into three parts, one for the core of the CO_2 15μm band, one for the wings and the third one for the remaining part of the spectrum. For the three parts, the transmission by dust is computed using grey absorption approximation. For the two intervals of the CO_2 15μm band, the total transmissivity is evaluated as the product between transmissivity of dust and that of carbon dioxide. Strictly speaking, it can be shown that this evaluation of the combined transmissivity is valid when there is no correlation between the spectral variations of the two absorbers. This is generally assumed for dust and carbon dioxide in the CO_2 15μm band. Scattering is not taken into account because of the strong isotropy of the thermal radiation.

Solar radiation:

in the original code developed by Fouquart and Bonel (1980), the upward and downward fluxs are obtained from the reflectances and transmittances of the layers. The interaction between gaseous absorption and scattering (by dust, molecules or clouds) is introduced using the *photon path distribution method.* At this stage, only absorption and scattering by dust (already present in the version of the code used at the ECMWF) is included in the Martian version although absorption by the near infra-red bands of carbon dioxide may become non negligible for very non-dusty conditions. The transmittances and reflectances of the layers are computed using the Delta-approximation to account for the strong asymmetry of the aerosol phase function.

The atmospheric dust content

is specified as a mixing ratio constant in both time and space, except for the vertical distribution which is taken according to Pollack *et al.* (1990). Beyond simplicity, the reason for this choice is that dust transport has not been included in the GCM until now. This strong approximation may affect the global CO_2condensation-sublimation cycle which primarily depends on the local dust-content at low latitudes (13). The optical parameters of dust (such as single scattering albedo and asymmetry factor) are taken from Pollack *et al.* (1979) .

3.2 Vertical turbulent mixing

The formulation of the vertical turbulent mixing is taken almost directly from the very simple scheme used in the LMD climate model. The effect of turbulent mixing on momentum and potential temperature is evaluated by the mean of a classical diffusion equation

$$\frac{\partial}{\partial t} = \frac{1}{\rho}\frac{\partial}{\partial z}\left(\rho K_z \frac{\partial}{\partial z}\right) \tag{3.1}$$

where the turbulent mixing coefficient K_z is computed as

$$K_z = l e^{1/2} \tag{3.2}$$

in term of mixing length l and a diagnostic estimate of the turbulent energy

$$e = Max\left\{ l^2 \left[\left(\frac{\partial \vec{V}}{\partial z}\right)^2 - 2.5\frac{g}{\theta}\frac{\partial \theta}{\partial z}\right], e_{min} \right\} \tag{3.3}$$

(the minimum value of the kinetic energy was set to $e_{min} = 1 \times 10^{-6}$ m^2 s^{-1}). In the case where $e > e_{min}$, the vertical mixing coefficient can be expressed as a function of the Richardson number

$$Ri = \frac{g\partial\theta/\partial z}{\theta\left(\partial\vec{V}/\partial z\right)^2} \tag{3.4}$$

as

$$K = l^2 \left\| \frac{\partial \vec{V}}{\partial z} \right\| \sqrt{1 - 2.5 Ri} \tag{3.5}$$

For both momentum and potential temperature, the turbulent surface flux is computed as the product between the vertical gradient of the quantity (estimated between the surface value and that in the first atmospheric layer) and a drag coefficient C_D given by:

$$C_D = C_{D\,0}\left[V_0 + ||\vec{V_1}||\right] \tag{3.6}$$

where $\vec{V_1}$ is the wind in the first atmospheric layer and $V_0 = 1$ m s^{-1}. The diffusion equation is integrated with an implicit time scheme. In the simulations presented below, l was fixed to 35 m and $C_{D\,0}$ to $2{\times}10^{-3}$, typical values used for terrestrial surfaces in the climate GCM.

we use the formulation developed by (8) for the terrestrial planetary boundary layer, in the form which has been implemented at ECMWF.

The drag coefficient is given by

$$C = \left(\frac{k}{\ln\frac{z}{z_0}}\right)^2 f\left(R_i, \frac{z}{z_0}\right) \tag{3.7}$$

where z_0 is the roughness length, z is the height of the middle of the first atmospheric layer, $k = 0.4$ is the von Karman constant and

$$R_i = \frac{gz\,(\theta - T_s)}{\theta ||V||^2} \tag{3.8}$$

is the Richardson number where $\theta = T\,(p_s/p)^\kappa$ and $||V||$ are the potential temperature and wind velocity in the first atmospheric layer, and T_s the surface temperature. Two different functions f are used for momentum (f_m) and potential temperature (f_θ). In either case, a different formulation is used depending on whether the atmosphere is stable ($R_i > 0$)

$$f_m = \frac{1}{1 + 2bR_i/\sqrt{1 + dR_i}} \tag{3.9}$$

$$f_\theta = \frac{1}{1 + 3bR_i\sqrt{1 + dR_i}} \tag{3.10}$$

or unstable ($R_i < 0$)

$$f_m = 1 - \frac{3bR_i}{1 + 3bC\,[k/\ln{(z/z_0)}]^2\,\sqrt{-R_i\,(1 + z/z_0)}} \tag{3.11}$$

$$f_\theta = 1 - \frac{3bR_i}{1 + 3bC\,[k/\ln{(z/z_0)}]^2\,\sqrt{-R_i z/z_0}} \tag{3.12}$$

As in the original terrestrial version, we use $C = b = d = 5$.

3.3 Convective adjustment

Whereas much more sophisticated schemes for vertical turbulent diffusion may spontaneously simulate some kind of vertical convection (see for example, Mellor and Yamada (1974)), our simple scheme (as many others) is not able to prevent subadiabatic vertical temperature gradients

$$\frac{\partial \theta}{\partial z} < 0 \tag{3.13}$$

If such an unstable profile is produced by the model, an adiabatic profile is immediately restored with a simple energy conserving scheme. If the resulting temperature profile is unstable at its upper or lower limit this mechanism is instantaneously extended in such a way that the final profile is entirely stable.

This convective adjustment is in fact achieved in a real atmosphere by parcel exchange through vertical convective motions. These motions not only transport energy but also momentum. The intensity of the momentum exchange is linked to the mass fluxs involved in the convection, which cannot be estimated in such a simple model. The best one can do is to compute an estimate of the instability of the atmosphere from the relative enthalpy exchange necessary to restore the adiabatic profile $\bar{\theta}$ from the original profile θ

$$\alpha = \int \left|\theta - \bar{\theta}\right| \rho dz \Big/ \int \theta \rho dz \tag{3.14}$$

When $\alpha < 1$ (this condition is always verified in the simulations) the angular momentum is entirely mixed on a fraction α of the mesh. This rather arbitrary choice is qualitatively acceptable in the sense that a larger instability will produce a larger momentum mixing.

3.4 Surface processes

Surface temperature evolution is governed by the balance between incoming fluxs (direct solar insolation, thermal radiation from the atmosphere and the surface itself and turbulent heat fluxs) and thermal conduction in the soil. The parametrization of this last process is often rather crude in terrestrial GCMs where a great part of the surface temperature are either imposed or computed in oceanic models. For a dry planet like Mars, an accurate parametrization of heat conduction is crucial to determine correctly surface temperatures and their response to diurnal, synoptic and seasonal forcing. A new parametrization was therefore developed for the Martian version of the LMD GCM.

The time evolution of the temperature under the surface is given by a classical conduction equation

$$\frac{\partial T}{\partial t} = -\frac{1}{C}\frac{\partial F_c}{\partial z} \tag{3.15}$$

where the conductive flux F_c is given by

$$F_c = -\lambda \frac{\partial T}{\partial z} \tag{3.16}$$

and where λ and C are the soil conductivity and specific heat per unit volume, respectively. In the simple case of a vertically homogeneous soil (which is assumed here), it can easily be shown that the model, as far as the time evolution of the surface temperature is concerned, is only dependent on the soil thermal inertia $I = \sqrt{\lambda C}$.

Although atmospheric GCMs often use force-restore schemes with one or two layers to simulate the time evolution of the surface temperature, it is much more accurate and straightforward to perform a direct temporal integration of these equations using a multi-layer difference scheme in the ground (4; 16). This was found to be numerically cheap enough even for a large number of layers in the soil: with 11 levels, this parametrization only represents 0.1% of the CPU-time of the Martian GCM. The soil model is similar to that presented by Warrilow *et al.* (1981). The accuracy of the model was checked by computing the phase and intensity of the surface temperature oscillation forced by a sin varying surface flux. For periods in the range from 0.3 to 2000 sols, the model produces errors of less than 1% on the intensity and phase shifts lower than $.02\pi$.

3.5 CO2 condensation-sublimation

A condensation temperature is introduced, following Pollack *et al.* (1981), as

$$T_{CO_2} = 149.2 + 6.48 \ln p \tag{3.17}$$

with T_{CO_2} in K and pressure p in hPa. Both atmosphere and surface temperatures are prevented from falling below T_{CO_2} by precipitating atmospheric CO_2onto the surface.

In the atmosphere,

if the temperature of a given layer falls below T_{CO_2} (as an effect of dynamics or other physical processes), condensation occurs, in an amount appropriate to restore, by latent heat release, the condensation temperature corresponding to the local pressure. All condensed carbon dioxide instantly precipitates to the ground without sublimation. Surface pressure is modified in agreement with the total amount of precipitation.

At the surface,

the temperature of the frost is kept at the condensation value either by condensing atmospheric CO_2or by sublimating CO_2ice.

The sublimation-condensation scheme exactly conserves both energy and mass. The energy balance on the caps is mainly controlled by albedo and emissivity of ice, which are unfortunately poorly known. The impact of those parameters on the annual pressure cycle has been carefully analysed in a recent study by Pollack *et al.* (1993). Direct measurements (5; 11), and model studies (15) yield large ranges of values: about 0.7-1. and 0.4-0.8 for emissivity and albedo, respectively. For this study, these quantities were somewhat arbitrarily fixed to 0.8 and 0.6 respectively (they were not tuned to fit Viking data). The approximation for the pressure-vapor curve used in paper 1 is replaced by a more accurate relationship based on the Clausius-Clapeyron relation for perfect gas. Assuming that the latent heat of sublimation L is independent of temperature, the vapor pressure curve reduces to

$$\ln\left(p/p_0\right) = \frac{L}{R}\left(\frac{1}{T_0} - \frac{1}{T_{\mathrm{CO_2}}}\right) \tag{3.18}$$

(where R is the gas constant and T_0 is the condensation temperature corresponding to the pressure chosen as a reference, here $p_0 = 1$ mbar). $L = 5.9 \times 10^5$ J kg^{-1} and $T_0 = 136.3$ K are fixed in the range of experimental values (6, e. g.). The change of the pressure-vapor curve was found to have a minor effect on the atmospheric mass budget.

partAspects informatiques

Chapter 4

Interface Physique/dynamique

Le modèle de circulation générale (MCG) effectue des calculs en chaque point d'un "maillage" 3D qui couvre l'ensemble de l'atmosphère. Des valeurs typiques de ce maillage sont par exemple 64x48x25 ou 96x72x25. Il s'agit du nombre de points en longitude, latitude et altitude. On obtient alors horizontalement des mailles de l'ordre de 200x200 kilomètres vers l'equateur. La répartition des couches verticales est irrégulière, afin de permettre notamment une plus grande précision au niveau du sol. On décrit généralement à l'aide de 25 couches l'atmosphère jusqu'à une hauteur de l'ordre de 80 km. La première couche décrit alors les quelques premiers mètres au dessus du sol, tandis que les dernières décrivent plusieurs kilomètres.

Le modèle fonctionne en deux parties:
- d'une part la **partie dynamique** contenant la résolution numérique des équations générales de la circulation atmosphérique. Cette partie est commune au modèle terrestre et martien, et plus généralement à toutes les atmosphères de type terrestre.
- D'autre part la **partie physique** propre à la planète considérée.

Les calculs concernant la partie dynamique, sont réellement effectués sur une grille 3D avec des échanges horizontaux entre mailles, alors que la partie physique peut être vue comme une juxtaposition de "colonnes" d'atmosphère n'interagissant pas entre elles (shéma 4.1).

On utilise pour chaque partie une grille différente. On a représenté figure 4.2 la numérotation des grilles physiques et dynamiques ainsi que les différentes positions possibles des variables sur ces grilles. Afin de repérer les coordonnées d'une variable (sur un point de grille, au dessus, en dessous, à droite, ou à gauche), on utilise les coordonnées **rlonu, rlatu, rlonv et rlatv**.

Pour la grille dynamique, on répète en i=IM+1 les valeurs de i=1 (périodicité en longitude). Les valeurs aux pôles sont quant à elles dupliquées IM+1 fois. En revanche, pour la grille physique, il n'y a qu'une valeur aux pôles et pas de périodicité en longitude.

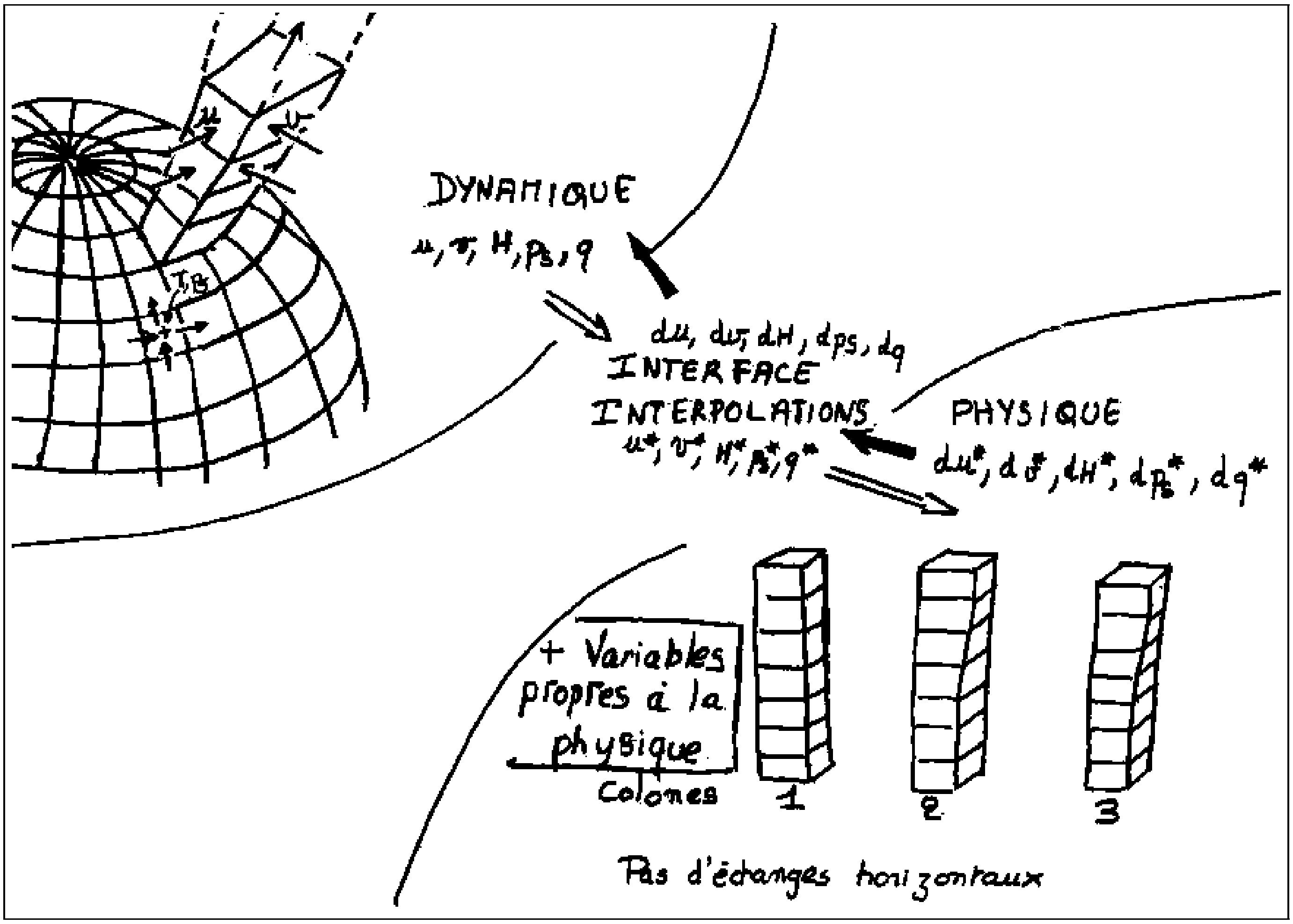

Figure 4.1: Interface physique/dynamique

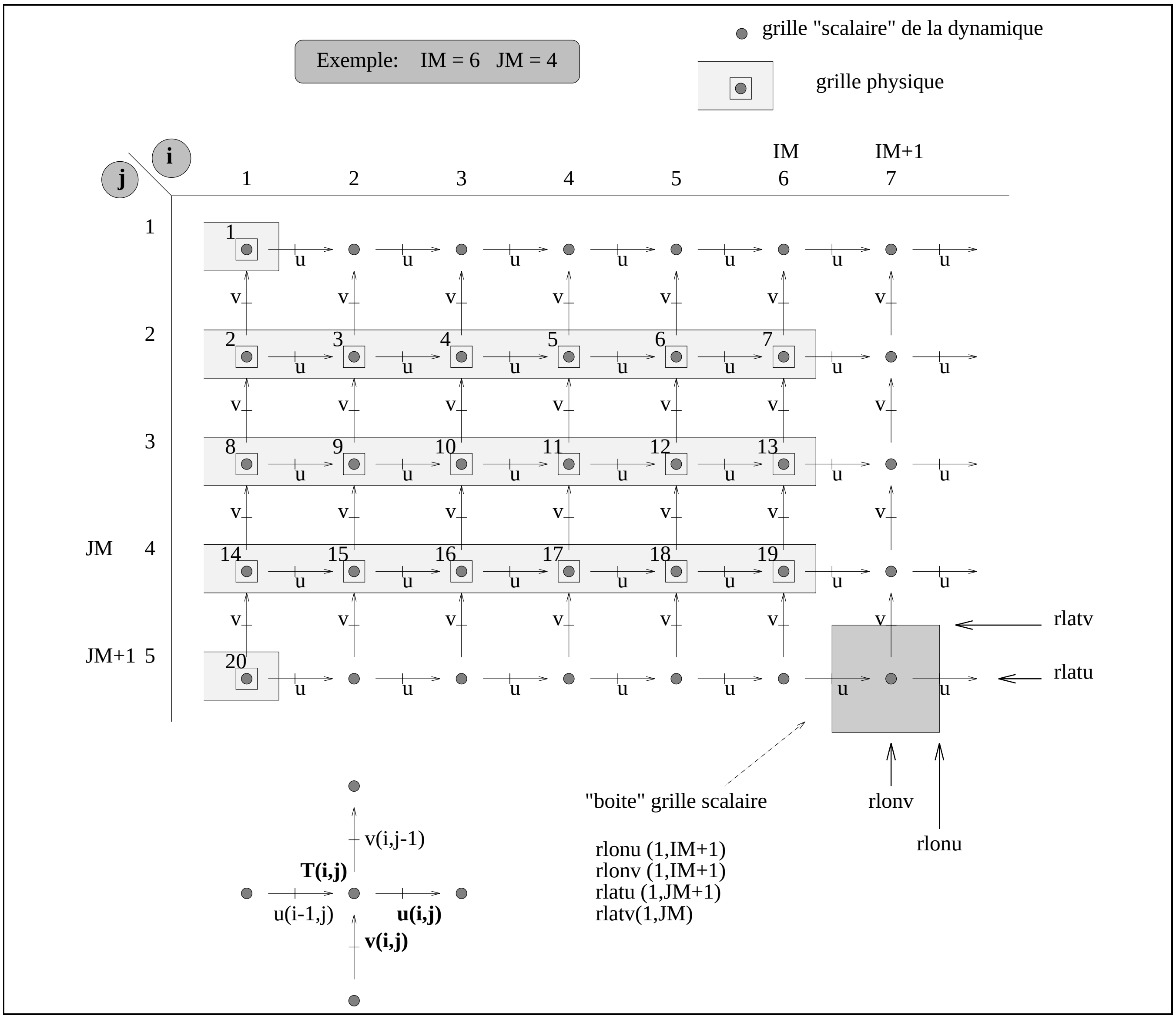

Figure 4.2: Grille physique et grille scalaire

Chapter 5

Entrées/Sorties

5.1 format DRS

Les entrées sorties du MCG sont au format DRS (data Retreavle Systeme). DRS est un jeu de routines FORTRAN qui permet de stocker et d'extraire des ensembles de données dans un fichier DRS. DRS est conçu pour générer des fichiers histoire pour les modèles de circulation atmosphérique, mais ses caractéristiques étendent son utilisation à d'autres systèmes. Une des caractéristique notable de DRS est sa capacité à stocker un jeu de données par morceaux, et ensuite à pouvoir extraire tout ou partie de cet ensemble par appel à une seule fonction. La librairie DRS a été conçue par le PCMDI (Program for Climate Model Diagnosis and Intercomparison) et est accessible sur le WWW.

`http://www-pcmdi.llnl.gov/drach/DRS.html`

Un jeu de données sous format DRS se compose en fait de 2 fichiers: un fichier qui contient les données (**.dat**), et un fichier dictionnaire (**.dic**) qui contient la description du fichier de données. Ces 2 fichiers sont des fichiers binaires.

5.1.1 L'éditeur de fichiers DRS: drsed

L'éditeur drsed est inclu dans la librairie DRS.

Principales commandes sous drsed

drsed diagfi

édite le fichier DRS "diagfi" (diagfi.dat diagfi.dic). drsed donne la main sous le prompt '>'

\> *list*

permet de visualiser les différents champs stockés dans "diagfi".

\> *print pression(i,j,k,l)*

affiche à l'écran la valeur du champ "pression" au point i, j, k, l où i, j, k et l sont exprimées en coordonnées réelles (longitude, latitude, altitude, temps).

\> *printi pression(i,j,k,l)*

affiche à l'écran la valeur du champ "pression" au point i, j, k, l, où i, j, k et l sont exprimées en coordonnées de grille.

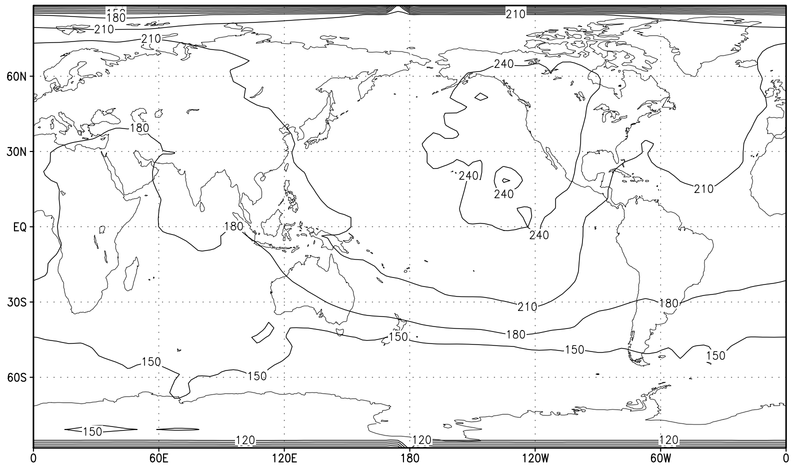

Figure 5.1: Exemple de visualisation sous GrADS des données de temperature à un instant donné

> *printi pression(i,j,k,*)*

affiche à l'écran la valeur du champ "pression" au point i, j, k, pour toutes les couches l, où i, j, k sont exprimées en coordonnées de grille.

> *print pression*

affiche à l'écran l'ensemble des valeurs du champ "pression"

> *help*

pour plus de détails.

5.1.2 Visualisation graphique des fichiers DRS sous GrAds

GrAdS (The Grid Analysis and Display System) est un logiciel graphique du domaine publique qui a été développé par Brian Doty au "Center for Ocean-Land-Atmosphere (COLA)".

Il permet entre autre de visualiser directement des données stockées sous le format DRS. On peut voir par exemple figure 5.1 la visualisation sous GrADS des données de temperature à un instant donné. Cependant, contrairement à DRS, GrADS ne reconnait que des fichiers où toutes les variables sont stockées sur la même grille horizontale. Ces variables peuvent être de 1, 2, 3 ou 4 dimensions (X,Y,Z et t).

GrADS est également accessible par le WWW.

`http://grads.iges.org/grads/`

GrADS a besoin comme DRS de deux fichiers: un fichier des données (**.dat**), et un fichier de contrôle (**.ctl**). Pour adapter les fichiers DRS au format GrAdS, l'utilitaire **dic2ctl** génère à partir du fichier .dic le fichier de contrôle .ctl. GrADS ne pouvant pas lire à partir du même fichier de contrôle les variables temporelles et non temporelles, le programme **dic2ctl** construit en fait deux fichiers de contrôle. Un fichier (**.ctl**), pour les variables qui évoluent dans le temps, et un fichier (**.head**) pour les variables de dimension non temporelle.

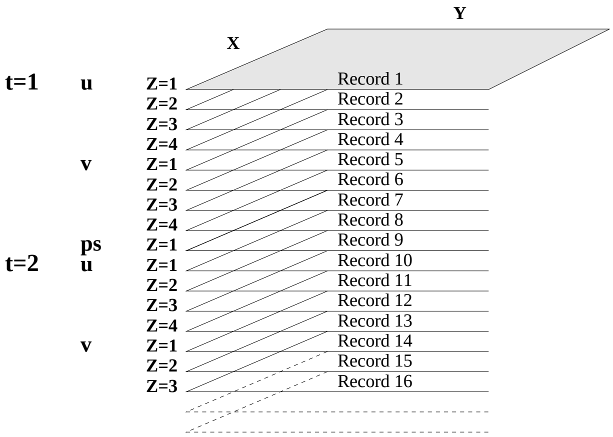

Figure 5.2: Exemple de stockage des vents u et v (champs 4D: X,Y,Z et t), et de la pression de surface ps (champ 3D: X,Y et t)

5.1.2.1 Le fichier de données (.dat)

Le fichier de données est un fichier binaire à accès direct, où un enregistrement constitue une grille horizontale X,Y. La figure 5.2 représente le stockage dans le fichier .dat des vents u et v (champs 4D: X,Y,Z et t), et de la pression de surface ps (champ 3D: X,Y et t).

5.1.2.2 Le fichier de contrôle (.ctl ou .head)

Le fichier de contrôle est un fichier ascii. Il indique (voir exemple ci-dessous) le nom du fichier contenant les données (dset), les valeurs des différentes dimensions (xdef, ydef, zdef et tdef) en chaque point de grille; chacune des valeurs si la dimension est non linéaire (zdef), la valeur de départ et l'incrément pour les dimensions linéaires (xdef, ydef et tdef). Il indique également le nombre de variables (ou de champs) stockées (vars) ainsi que leurs noms, un commentaire et l'unité.

```
--------------------------------------------------------
dset statsa2m4.dat
title Run    0.000, day     0 to     0
undef -9.99E33
options yrev
xdef    64 linear -180.0000   5.6250
ydef    49 linear  -88.1633   3.6735
zdef    25 levels
 0.9996389 0.9983609 0.9960319 0.9918210 0.9842223
 0.9706507 0.9467649 0.9058902 0.8391817 0.7385885
 0.6042400 0.4522929 0.3100970 0.1979461 0.1201197
 0.0704957 0.0404525 0.0228180 0.0126525 0.0068702
 0.0036093 0.0018078 0.0008212 0.0003282 0.0000653
tdef    12 linear          0z1jan0001  2hr
vars         18
co2ice         0 99  mean CO2 ice cover (kg/m2)
sdco2ice       0 99  stdv CO2 ice cover (kg/m2)
emis           0 99  mean emissivity
sdemis         0 99  stdv emissivity
tsurf          0 99  mean surface temperature (K)
sdtsurf        0 99  stdv surface temperature (K)
```

```
ps              0 99  mean surface pressure (Pa)
sdps            0 99  stdv surface pressure (Pa)
t              25 99  mean temperature (K)
sdt            25 99  stdv temperature (K)
u              25 99  mean zonal velocity (m/s)
sdu            25 99  stdv zonal velocity (m/s)
v              25 99  mean meridional velocity (m/s)
sdv            25 99  stdv meridional velocity (m/s)
rho            25 99  mean density (kg/m3)
sdrho          25 99  stdv density (kg/m3)
q2             25 99  mean b/layer eddy ke (m2/s2)
sdq2           25 99  stdv b/layer eddy ke (m2/s2)
endvars
-------------------------------------------------------
```

5.2 Entrées

Le MCG a besoin en entrée de deux fichiers (DRS) d'initialisation:
-**start** contenant les états initiaux des variables de la dynamique.
-**startfi** contenant les états initiaux des variables de la physique.
et de trois fichiers (ascii) de paramètres:
-**run.def** les paramètres de la partie dynamique du programme, et de l'intégration temporelle du modèle.
-**callphys.def** les paramètres pour l'appel de la physique.
-**esasig.def** les paramètres pour la distribution verticale des couches.

5.2.1 run.def

```
-------------------------------------------------------
Parametres de controle du run:
------------------------------

- Nombre de jours d'integration
    nday
  47

- nombre de pas par jour (multiple de iperiod)
day_step
  960.

- periode pour le pas Matsuno (en pas)
 iperiod
  5

- periode de sortie des variables de controle (en pas)
 iconser
  120

- periode d'ecriture du fichier histoire (en jour)
   iecri
  10

- periode de stockage fichier histmoy (en jour)
```

```
 periodav
  1

- periode de la dissipation (en pas)
 idissip
  5

- choix de l'operateur de dissipation (star ou  non star )
  lstardis
  F

- nombre d'iterations de l'operateur de dissipation   gradiv
  nitergdiv
  1

- nombre d'iterations de l'operateur de dissipation  nxgradrot
  nitergrot
  2

- nombre d'iterations de l'operateur de dissipation  divgrad
  niterh
  2

- temps de dissipation des plus petites long.d ondes pour u,v (gradiv)
  tetagdiv
  5500.

- temps de dissipation des plus petites long.d ondes pour u,v (nxgradrot)
  tetagrot
  3600.

- temps de dissipation des plus petites long.d ondes pour   h
  tetatemp
  3600.

- coefficient pour gamdissip
 coefdis
  0.

- periode de la physique (en pas)
  iphysiq
  20

- choix du shema d'integration temporelle (Matsuno ou Matsuno-leapfrog)
 purmats
  F

- choix d'une grille reguliere
  grireg
  T

- avec ou sans physique
  physic
  T
```

```
- frequence (en  pas !! ) de l'ecriture du fichier diagfi
  ecritphy
  960.

- longitude en degres du centre du zoom
  clon
  0.

- latitude en degres du centre du zoom
  clat
  0.

- facteur de grossissement du zoom,selon longitude
  alphax
  0.

- facteur de grossissement du zoom ,selon latitude
  alphay
  0.

-  Fonction  f(y)  hyperbolique ( si .true. )  ou   sinusoidale
 fxyhypb
 F

.
-----------------------------------------------------
```

5.2.2 callphys.def

```
-----------------------------------------------------
callrad (T => calcul du transfert radiatif)
T
calldifv (T => calcul de la diffusion verticale)
T
mellor (T => diffusion verticale Mellor et Yamada)
T
calladj (T => calcul de l'ajustement convectif)
T
callcond (T => calcul de Condensation-sublimation du CO2)
T
callsoil (T => Utilisation d'un modele de sol multi-couches)
T
calllott (T => Prise en compte de l'effet du relief sous-maille)
T
season (T => Avec cycle saisonnier)
T
diurnal (T => Avec cycle diurne)
T
lwrite (T => sortie debug)
F
laervar (T => aerosols =f(Ls) (scenar viking si tauvis=1)
T
laervarlat (T => topdust = f(Ls,lat) (scenar viking))
```

```
T
topdustref (sommet couche poussiere (km) - utilise si laervarlat=F et esadust=T)
55.
esadust (normalement T - F:Ancienne distribution (Pollack 90))
T
iradia ( On appelle le rayonnement tous les iradia pas physiques)
2
callstats ( T=> sortie des fichiers ESA "cumul")
T
callco2v (T => Calcul du rayonnement proche IR par le CO2)
T
callnlte (T => prise en compte des effets NON LTE dans le proche IR)
T
-------------------------------------------------------
```

5.2.3 esasig.def

```
-------------------------------------------------------
10  atmospheric scale height (km) (used as a reference only)
80  maximum altitude (km)
.01 thikness of the first layer (km)
2.  number of layer for "sponge layer" at the top
-------------------------------------------------------
```

5.2.4 les fichiers d'initialisation: start et startfi

Les fichiers start et startfi, comme tout les fichier DRS du GCM, sont bâtis sur le même modèle (voir la composition d'un fichier DRS figure 5.3). Ils contiennent:
- un entête constitué d'une variable "contrôle" suivie d'une série de variables définissant la grille (physique ou dynamique).
- Une série de variables non temporelles donnant des informations sur les conditions de surface de la planète.
- Une variable "temps" donnant les valeurs des différent instants auxquels les variables temporelles sont stockées (une seule valeur de temps (t=0) pour start et startfi, puisqu'il s'agit des conditions intiales).
- les variables d'état du modèle sur la grille correspondante aux différents instants de la simulation sélectionnés (réglé par un pas de temps) pour les fichiers de sortie en cours de simulation (diagfi, histmoy, etc...).

Visualisons à l'aide de l'éditeur drsed le contenu d'un fichier start:
(Attention, la commande "list" de drsed donne les variables dans l'ordre inverse du stockage; la 1ère variable stockée est donc la dernière dans le listing à savoir la variable "contrôle", puis la variable "rlonu"...)

drsed start
> list

```
-------------------------------------------------------
phis: geopotentiel au sol
pext: pression extensive
q01: traceur
h: temperature potentielle
vcov: vents v covariants
ucov: vents u covariants
```

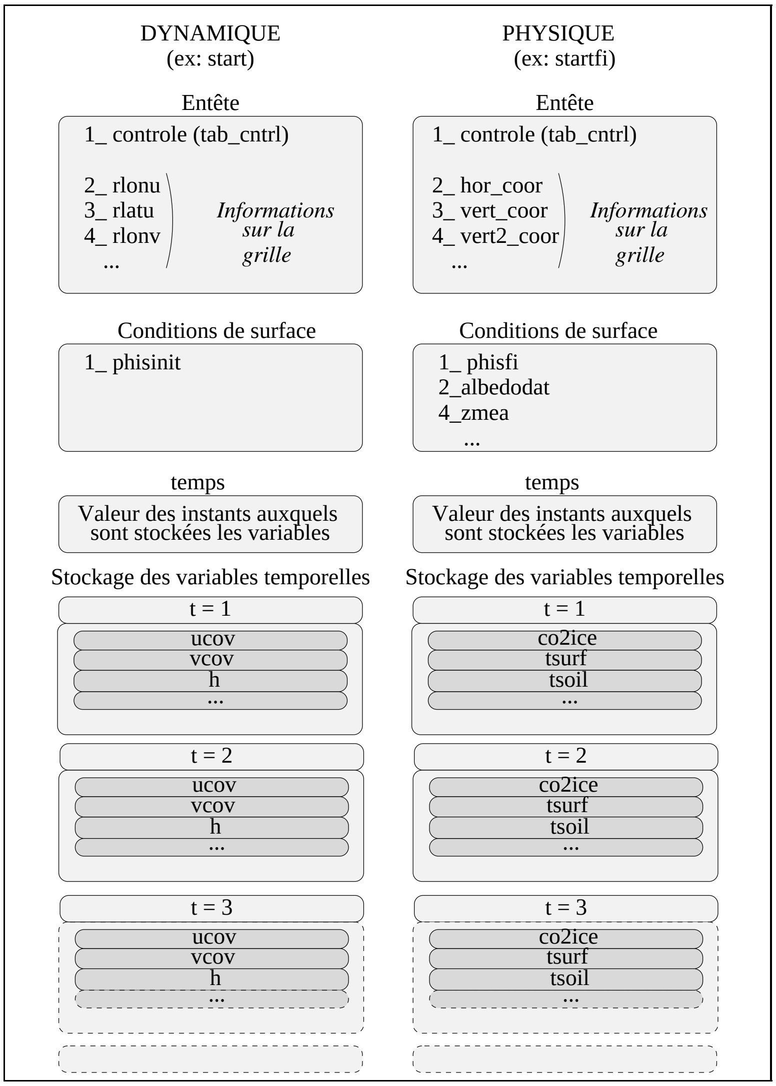

Figure 5.3: Organisation des fichiers DRS

```
temps: temps simule
phisinit: Geopotentiel au sol init
aire: Aires des mailles
cv: Coefficients de passage cov <--> naturel
cu: Coefficients de passage cov <--> naturel
sig_s: sigmas aux niveaux s
s: Niveaux s
sig: Niveaux sigmas
rlatv: Latitudes aux points v
rlonv: Longitudes aux points v
rlatu: Latitudes aux points u
rlonu: Longitudes aux points u
controle: Tableau des parametres du run
-------------------------------------------------------
```

Listing du contenu d'un fichier startfi:

drsed startfi
> list

```
-------------------------------------------------------
q2: pbl wind variance
emis: surface emissivity
tsoil: Soil temperature
tsurf: Surface temperature
co2ice: CO2 ice cover
temps: temps simule
zthe: Relief: parametre theta
zgam: Relief: parametre gamma
zsig: Relief: parametre sigma
zstd: ecartype du relief
zmea: Relief moyen
inertiedat: Inertie thermique du sol
albedodat: Albedo du sol nu
phisfi: Geopotentiel au sol init
area: Aires des mailles
lati: Latitudes
long: Longitudes
surf_coor: Indice vertical sol
vert2_coor: Indice vertical (bords)
vert_coor: Indice vertical
hor_coor: Indice horizontal
controle: Tableau des parametres du run
-------------------------------------------------------
```

Entêtes physiques et dynamiques

Il y a deux types d'entêtes: un pour les fichiers physiques et un pour les fichiers dynamiques. Cet entête commence toujours par une variable "contrôle" (décrite un peu plus loin), que l'on affecte différemment pour la physique et pour la dynamique. Les autres variables de l'entête concernent les grilles (physique et dynamique). On retrouve donc:

les coordonnées horizontales
- **rlonu**, **rlatu**, **rlonv**, **rlatv** pour la dynamique,
- **lati**, **long** pour la physique,

les coordonnées verticales
- **sig** ,**s** ,**sig_s** pour la dynamique,
- **surf_coor**, **vert2_coor**, **vert_coor**, **hor_coor** pour la physique,

des coefficients de passages de la grille physique à la grille dynamique
- **cu**,**cv** dans l'entête dynamique uniquement

et pour finir l'aire des mailles
- **aire** pour la dynamique,
- **area** pour la physique.

Conditions de surface

Des conditions de surface sont données essentiellement dans les fichiers DRS physiques par les variables:

- **phisfi** pour le géopotentiel au sol initial,
- **albedodat** pour l'albedo du sol nu,
- **inertiedat** pour l'inertie thermique du sol,
- **zmea**, **zstd**, **zsig**, **zgam** et **zthe** pour le relief.

Pour la dynamique:
- **physinit** pour le géopotentiel au sol initial

Remarque: les variables **phisfi** et **physinit** contiennent les mêmes informations (géopotentiel au sol), mais physfi donne les valeurs du géopotentiel sur la grille physique tandis que physinit les donne sur la grille dynamique.

Variables d'état physiques et dynamiques

Pour des raisons de commodité, on stocke dans les fichiers d'initialisation, non pas les variables "naturelles", mais directement les variables utilisées par le modèle.

Pour la dynamique:

- **ucov** et **vcov** les vents covariants
  Ces variables sont reliées aux vents "naturels" par
  `ucov = cu * u` et `vcov = cv * v`

- **h** la température potentielle,
  ou plus précisément l'enthalpie potentielle reliée à **T** la temperature par $h = C_p T \left(\frac{P_s}{P}\right)^K$

- **q01** les traceurs,

- **pext** la pression multipliée par l'aire de la maille
  reliée à **ps** la pression par ...

ucov et **vcov**, variables "vectorielles", sont stockés respectivement sur les grilles u et v "décalée" de la dynamique (voir section 4.2).
h, **q01**, **pext**, variables scalaires, sont stockées sur la grille dite "scalaire" de la dynamique.

Pour la physique:

- **co2ice** la glace carbonique en surface,

- **tsurf** la température de surface,

- **tsoil** la température dans différentes couches sous la surface,

- **emis** l'emissivité de la surface,

- **q2** la variance du vent,
  ou plus précisement la racine carrée de l'énergie cinétique turbulente.

Toutes ces variables sont stockées sur la grille "physique" (voir section 4.2).

La variable "contrôle"

Qu'ils soient physiques ou dynamiques, les entêtes des fichiers DRS du GCM commencent tous par une variable **contrôle**. Cette variables est un tableau de 100 réels (tableau appellé "tab_cntrl" dans le programme), qui contient les paramètres de contrôle du programme. Ces paramètres, différents pour la physique et la dynamique, sont listés ci-dessous. On peut également visualiser le contenu du tableau tab_cntrl en tapant la commande "> *print controle*" sous drsed.

La variable "contrôle" de l'entête d'un fichier DRS dynamique:

```
-------------------------------------------------------
tab_cntrl(1) = float(iim)       ! nombre de points en longitude
tab_cntrl(2) = float(jjm)       ! nombre de points en latitude
tab_cntrl(3) = float(llm)       ! nombre de couches
tab_cntrl(4) = float(idayref)   ! jour 0
tab_cntrl(5) = rad     ! rayon de mars(m) ~3397200
tab_cntrl(6) = omeg    ! vitesse de rotation (rad.s-1)
tab_cntrl(7) = g       ! gravite (m.s-2) ~3.72
tab_cntrl(8) = mugaz   ! Masse molaire de l'atm (g.mol-1) ~43.49
tab_cntrl(9) = kappa   ! = r/cp  ~0.256793 (=rcp dans physique)
tab_cntrl(10) = daysec ! duree du sol (s)  ~88775
tab_cntrl(11) = dtvr   ! pas de temps de la dynamique (s)

tab_cntrl(12) = etot0  ! energie totale    !
tab_cntrl(13) = ptot0  ! pression totalei   !    variables
tab_cntrl(14) = ztot0  ! enstrophie totale   !  de controle
tab_cntrl(15) = stot0  ! enthalpie totale   !    globales
tab_cntrl(16) = ang0   ! moment cinetique  !
tab_cntrl(17) = clon   ! longitude en degres du centre du zoom
tab_cntrl(18) = clat   ! latitude en degres du centre du zoom
tab_cntrl(19) = alphax ! facteur de grossissement du zoom,selon longitude
tab_cntrl(20) = alphay ! facteur de grossissement du zoom ,selon latitude
-------------------------------------------------------
```

La variable "contrôle" de l'entête d'un fichier DRS physique:

```
-------------------------------------------------------
c Info sur la grille physique
      tab_cntrl(1) = float(ngridmx)  ! nombre de points de la grille physique
      tab_cntrl(2) = float(nlayermx) ! nombre de couches
      tab_cntrl(3) = float(idayref)  ! jour 0

c Info sur la Planete Mars pour la dynamique et la physique
      tab_cntrl(5) = rad      ! rayon de mars(m) ~3397200
      tab_cntrl(6) = omeg     ! vitesse de rotation (rad.s-1)
      tab_cntrl(7) = g        ! gravite (m.s-2) ~3.72
      tab_cntrl(8) = mugaz    ! Masse molaire de l'atm (g.mol-1) ~43.49
      tab_cntrl(9) = rcp      !  = r/cp  ~0.256793 (=kappa dans dynamique)
      tab_cntrl(10) = daysec  ! duree du sol (s)  ~88775

      tab_cntrl(11) = dtphys  ! pas de temps de la physique
      tab_cntrl(12) = 0.
      tab_cntrl(13) = 0.

c Info sur la Planete Mars pour la physique uniquement
```

```
      tab_cntrl(14) = year_day  ! duree de l'annee (sols) ~668.6
      tab_cntrl(15) = periheli  ! dist.min. soleil-mars (Mkm) ~206.66
      tab_cntrl(16) = aphelie   ! dist.max. soleil-mars (Mkm) ~249.22
      tab_cntrl(17) = peri_day  ! date du perihelie (sols depuis printemps)
      tab_cntrl(18) = obliquit  ! Obliquite de la planete (deg) ~23.98

c Couche limite et Turbulence
      tab_cntrl(19) = z0        ! surface roughness (m) ~0.01
      tab_cntrl(20) = lmixmin   ! longueur de melange ~100
      tab_cntrl(21) = emin_turb ! energie minimale ~1.e-8

c propriete optiques des calottes et emissivite du sol
      tab_cntrl(22) = albedice(1)  ! Albedo calotte nord ~0.5
      tab_cntrl(23) = albedice(2)  ! Albedo calotte sud ~0.5
      tab_cntrl(24) = emisice(1)   ! Emissivite calotte nord ~0.95
      tab_cntrl(25) = emisice(2)   ! Emissivite calotte sud ~0.95
      tab_cntrl(26) = emissiv      ! Emissivite du sol martien ~.95
      tab_cntrl(31) = iceradius(1) ! mean scat radius of CO2 snow (north)
      tab_cntrl(32) = iceradius(2) ! mean scat radius of CO2 snow (south)
      tab_cntrl(33) = dtemisice(1) ! time scale for snow metamorphism (north)
      tab_cntrl(34) = dtemisice(2) ! time scale for snow metamorphism (south)

c Proprietes des poussiere aerosol
      tab_cntrl(27) = tauvis      ! profondeur optique visible moyenne

      tab_cntrl(28) = 0.
      tab_cntrl(29) = 0.
      tab_cntrl(30) = 0.
-------------------------------------------------------
```

5.3 Sorties

5.3.1 fichiers DRS de redemarrage restart et restartfi

5.3.2 fichiers DRS histmoy

On stocke dans le fichier DRS histmoy (**histmoy.dat histmoy.dic**), tout au long de la simulation et à intervalle de temps régulier, les variables dynamiques moyennées.

Visualisation sous drsed du contenu d'un fichier histmoy:

drsed histmoy
> list

```
-------------------------------------------------------
ps: pression extensive moyennee
q1:
T: temperature moyennees
v: vents v naturels moyennes
u: vents u naturels moyennees
dttot: periode moyennee
temps: temps simule
phisinit: Geopotentiel au sol init
aire: Aires des mailles
cv: Coefficients de passage cov <--> naturel
cu: Coefficients de passage cov <--> naturel
sig_s: sigmas aux niveaux s
s: Niveaux s
sig: Niveaux sigmas
rlatv: Latitudes aux points v
rlonv: Longitudes aux points v
rlatu: Latitudes aux points u
rlonu: Longitudes aux points u
controle: Tableau des parametres du run
-------------------------------------------------------
```

On retrouve (de bas en haut du listing) de la variable "controle" à la variable "aire", l'entête dynamique, puis "phisinit" pour les conditions de surface dynamique, suivi de la variable "temps" qui contient les valeurs des différents instants auxquels on a stocké les variables d'état dynamique moyennées.

- le vent zonal **u**
- le vent meridien **v**
- la température **T**
- les traceurs **q1**
- la pression extensive **ps**

Une variable supplémentaire **dttot** donne l'intervalle de temps sur lequel on a moyenné (temps écoulé entre deux instants de stockage).

On règle la fréquence de stockage des variables dans le fichier "histmoy" par le paramètre **periodav** du fichier "run.def" (voir 5.2.1)

5.3.3 fichiers DRS diagfi

On stocke dans le fichier DRS diagfi (**diagfi.dat diagfi.dic**), tout au long de la simulation et à intervalle de temps régulier, les variables physique instantannée.

Visualisation sous drsed du contenu d'un fichier diagfi:

drsed diagfi
> list

```
-------------------------------------------------------
ray: bla
cl: bla
cond: bla
gw: bla
q2: wind variance
v: Vent merid
u: Vent zonal
dtrad: dT/dt
dtsw: dT/dt SW
dtswc: dT/dt SWc
dtlwc: dT/dt LWc
dtlw: dT/dt LW
T: Temperature
Tsurf: Surf T
ps: Psurf
emis: grd emis
co2ice: couche de glace co2
temps: temps simule
phisinit: Geopotentiel au sol init
aire: Aires des mailles
cv: Coefficients de passage cov <--> naturel
cu: Coefficients de passage cov <--> naturel
sig_s: sigmas aux niveaux s
s: Niveaux s
sig: Niveaux sigmas
rlatv: Latitudes aux points v
rlonv: Longitudes aux points v
rlatu: Latitudes aux points u
rlonu: Longitudes aux points u
controle: Tableau des parametres du run
-------------------------------------------------------
```

Une petite particularité du fichier "diagfi" est de contenir un entête dynamique alors qu'il stocke des variables physiques. Ceci par commodité, afin de pouvoir utiliser le même programme de conversion au format GrAdS (dic2ctl) que pour les fichiers histmoy.

On retrouve donc (de bas en haut du listing) de la variable "controle" à la variable "aire", l'entête dynamique, puis "phisinit" pour les conditions de surface dynamique, suivi de la variable "temps" qui contient les valeurs des différents instants auxquels on a stocké les variables d'état physiques instantannées.

On règle la fréquence de stockage des variables dans le fichier "diagfi" par le paramètre **ecritphy** du fichier "run.def" (voir 5.2.1)

5.3.4 fichiers "anglais" cumul/stats

Chapter 6

Organisation informatique et compilation

Le modèle du LMD est organisé sur un répertoire de base. Ce répertoire est associé à la variable d'environnement **LMDGCM**. L'exemple ci-dessous montre l'affectation de cette variable en csh sous UNIX pour la version de référence du modèle au LMD.

```
setenv LMDGCM /users/lmdz/SOURCES/LMDZ.2.2
```

Voici très sommairement le contenu de ce répertoire.

```
lmdz.2/

   cray/  Fichiers necessaires pour faire tourner le modele sur le cray.

   libf/  Ensemble des Sources FORTRAN du modele (.F pour cpp) et
            Fichiers include (.h) organises en sous repertoires
                      (physique, dynamique, filtres ...).

   create_make_gcm   Executable servant a la creation du makefile.
                      Cette commande est executee automatiquement
                            par "makegcm" (voir plus loin).
```

6.1 Orgnisation des fichiers sources du modèle

Les fichiers sources du modèle sont stockés dans différents sous-répertoires au sein du répertoire **libf**. Ces sous-répertoires correspondent à différentes parties du modèle:

grid: Constitué essentiellement du fichier "dimensions.h", qui contient les paramètres définissant la grille du modèle, à savoir le nombre de points en longitude (IIM), en latitude (JJM) et sur la verticale (LLM), ainsi que le nombre de traceurs (NQMX) advectés dans la dynamique (par exemple 2 pour l'eau et la vapeur d'eau pour la version de base du modèle terrestre). Le second fichier contient les fonction $\lambda(I)$, $\phi(J)$ reliant la longitude et la latitude aux indices du modèle, ainsi que leur dérivées.

dyn3d: contient les sous-programmes de la dynamique.

phylmd: contient les sous-programmes correspondant au paramétrisations physique du modèle dévelopées au LMD.

phyec: les sous-programmes de la physique du Centre Européen pour les Prévisions Météorologiques à Moyen Terme.

phymars: les sources de la physiques martienne (mêmme chose pour phytitan, phyparam...).

filtrez: contient les sources du filtre en longitude apppliqués dans les hautes latitude, la où le critère de stabilité de Courant-Friedrich-Levy est violé.

adjnt: contient les sources de l'adjoint du code dynamique.

Le modèle est entièrement écrit en **FORTRAN-77**. Les sources des programmes sont écrit dans des fichiers "fichier.F". Le suffixe .F est le suffixe standard pour un fichier FORTRAN qui doit être passé par le **pré-processeur-C (cpp)** avant compilation. La plupart des compilateur FORTRAN reconnaissent automatiquement ce suffixe. Un fichier source correspond généralement à un programme ou sous-programme FORTRAN ou à un petit ensemble cohérent de sous-programmes (par exemple, le code radiatif qui est souvent décomposé en une dizaine de sous-programme est stoké dans certaines versions du modèle dans un seul fichier .F).

Si l'on souhaite faire tourner le modèle sur un cray, le directory **cray** contient les sources des fonctions intrinsèques disponibles sur le CRAY et utilisées par le modèle.

6.2 Organigramme du modèle

6.2.1 La partie dynamique: gcm.F

```
gcm
   |- lectba
   |- defrun_new
   |- iniconst
   |        |- -- disvert
   |- inigeom
   |        |- -- fxyhyp
   |- inifilr
   |        |- -- inifgn
   |                   |- -- jacobi
   |                   |- -- acc
   |                   |- -- eigen_sort
   |                   |- -- jacobi
   |                   |- -- acc
   |                   |- -- eigen_sort
   |- inidissip
   |        |- -- filtreg
   |        |- -- divgrad2
   |                   |- -- grad
   |                   |- -- divergst
   |                   |- -- filtreg
   |                   |- -- grad
   |                   |- -- covcont
   |                   |- -- diverg
   |                   |- -- filtreg
   |        |- -- divgrad
```

```
|                  |--- filtreg
|                  |--- grad
|                  |--- covcont
|                  |--- diverg
|                  |--- filtreg
|        |--- filtreg
|        |--- covcont
|        |--- gradiv2
|                  |--- diverg
|                  |--- filtreg
|                  |--- grad
|                  |--- divergst
|                  |--- filtreg
|                  |--- grad
|        |--- gradiv
|                  |--- diverg
|                  |--- filtreg
|                  |--- grad
|                  |--- covcont
|        |--- nxgraro2
|                  |--- rotat
|                  |--- filtreg
|                  |--- nxgradst
|                  |--- rotatst
|                  |--- filtreg
|                  |--- nxgrad
|        |--- nxgrarot
|                  |--- rotat
|                  |--- filtreg
|                  |--- nxgrad
|- exner_new
|        |--- filtreg
|- iniwrite
|- iniav
|        |--- iniwrite
|- test_period
|- filtreg
|- geopot
|- caldyn
|        |--- covcont
|        |--- pbar
|        |--- flumass
|        |--- dh1
|                  |--- convflu
|                  |--- filtreg
|        |--- convmas
|                  |--- convflu
|                  |--- filtreg
|        |--- vitvert
|        |--- tourpot
|                  |--- filtreg
|        |--- dudv1
|        |--- enercin
|        |--- bernoui
```

```
|                     |−−− filtreg
|         |−−− dudv2
|         |−−− advect
|− integrd
|         |−−− filtreg
|− exner_new
|         |−−− filtreg
|− calfis
|         |−−− gr_dyn_fi
|         |−−− inifis
|         |−−− multipl
|         |−−− gr_dyn_fi
|         |−−− initial0

|         |−−− physiq

|         |−−− gr_fi_dyn
|         |−−− multipl
|− addfi
|− exner_new
|         |−−− filtreg
|− sponge
|− dissip
|         |−−− initial0
|         |−−− covcont
|         |−−− gradiv2
|                     |−−− diverg
|                     |−−− filtreg
|                     |−−− grad
|                     |−−− divergst
|                     |−−− filtreg
|                     |−−− grad
|         |−−− gradiv
|                     |−−− diverg
|                     |−−− filtreg
|                     |−−− grad
|                     |−−− covcont
|         |−−− nxgraro2
|                     |−−− rotat
|                     |−−− filtreg
|                     |−−− nxgradst
|                     |−−− rotatst
|                     |−−− filtreg
|                     |−−− nxgrad
|         |−−− nxgrarot
|                     |−−− rotat
|                     |−−− filtreg
|                     |−−− nxgrad
|         |−−− divgrad2
|                     |−−− grad
|                     |−−− divergst
|                     |−−− filtreg
|                     |−−− grad
|                     |−−− covcont
```

```
|                   |−−− diverg
|                   |−−− filtreg
|       |−−− divgrad
|                   |−−− filtreg
|                   |−−− grad
|                   |−−− covcont
|                   |−−− diverg
|                   |−−− filtreg
|− addit
|− writeavs
|       |−−− initial0
|       |−−− covnat
|       |−−− exner_new
|                   |−−− filtreg
|       |−−− multscal
|       |−−− multipl
|       |−−− multscal
|       |−−− multipl
|       |−−− gr_u_scal
|       |−−− gr_v_scal
|       |−−− covnat
|       |−−− exner_new
|                   |−−− filtreg
|       |−−− multscal
|       |−−− multipl
|       |−−− multscal
|       |−−− multipl
|       |−−− gr_u_scal
|       |−−− gr_v_scal
|− geopot
|− writehis
|− writedem
|− writeavs
|       |−−− initial0
|       |−−− covnat
|       |−−− exner_new
|                   |−−− filtreg
|       |−−− multscal
|       |−−− multipl
|       |−−− multscal
|       |−−− multipl
|       |−−− gr_u_scal
|       |−−− gr_v_scal
|       |−−− covnat
|       |−−− exner_new
|                   |−−− filtreg
|       |−−− multscal
|       |−−− multipl
|       |−−− multscal
|       |−−− multipl
|       |−−− gr_u_scal
|       |−−− gr_v_scal
|− geopot
|− writehis
```

|– **writedem**

6.2.2 La partie physique: physiq.F

physiq

```
|– readfi
|      |– –– tabfi
|– surfini
|– iniorbit
|– sucst
|– surad
|      |– –– suaer
|               |– –– aerave
|                        |– –– blackl
|      |– –– sulw
|      |– –– susw
|– soil
|– iniwritefi
|– sugwd
|– instat
|– solarlong
|      |– –– iniorbit
|– zerophys
|– stats
|      |– –– wstats
|– orbite
|– solang
|– mucorr
|– radite
|      |– –– lwmain
|               |– –– lwu
|               |– –– lwb
|               |– –– lwv
|                        |– –– flusv
|                                 |– –– sys3v
|                        |– –– lwvn
|                                 |– –– lwtt
|                        |– –– lwvd
|                                 |– –– lwtt
|                        |– –– lwvb
|                                 |– –– lwtt
|      |– –– writeg1d
|      |– –– sw
|               |– –– swu
|                        |– –– swtt
|               |– –– sw1s
|                        |– –– swr
|                                 |– –– dedd
|      |– –– writeg1d
|– drag_noro
|      |– –– orodrag
|               |– –– orosetup
|               |– –– gwstress
```

```
|                       |− −− gwprofil
|− zerophys
|− vdifc
|         |− −− vdif_cd
|         |− −− vdif_kc
|         |− −− multipl
|− zerophys
|− convadj
|− newcondens
|         |− −− co2snow
|− soil
|− writefi
|− iniwritefi
|− writefi
|− writedrsfi
|         |− −− gr_fi_dyn
|         |− −− iniwrite
```

6.2.3 etats initiaux: newstart.F

newstart

```
|− tabfi
|− defrun_new
|− iniconst
|         |− −− disvert
|− inigeom
|         |− −− fxyhyp
|− gr_dyn_fi
|− lect_start_archive
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− gr_dyn_fi
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− gr_dyn_fi
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− gr_dyn_fi
|         |− −− interp_vert
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− gr_dyn_fi
|         |− −− interp_vert
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− interp_vert
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− interp_vert
|         |− −− interp_horiz
|                       |− −− iniinterp_h
|         |− −− scal_wind
|         |− −− interp_horiz
```

```
|                       |−−− iniinterp_h
|           |−−− gr_dyn_fi
|− lectba
|− readfi
|           |−−− tabfi
|− datareaddrs
|           |−−− initial0
|           |−−− dump2d
|           |−−− interp_horiz
|                       |−−− iniinterp_h
|           |−−− dump2d
|           |−−− multscal
|− gr_dyn_fi
|− initial0
|− multscal
|− gr_dyn_fi
|− ini_fi
|           |−−− iniwritefi
|           |−−− writefi
|− inifilr
|           |−−− inifgn
|                       |−−− jacobi
|                       |−−− acc
|                       |−−− eigen_sort
|                       |−−− jacobi
|                       |−−− acc
|                       |−−− eigen_sort
|− exner
|           |−−− filtreg
|− geopot
|− caldyn
|           |−−− covcont
|           |−−− pbar
|           |−−− flumass
|           |−−− dh1
|                       |−−− convflu
|                       |−−− filtreg
|           |−−− convmas
|                       |−−− convflu
|                       |−−− filtreg
|           |−−− vitvert
|           |−−− tourpot
|                       |−−− filtreg
|           |−−− dudv1
|           |−−− enercin
|           |−−− bernoui
|                       |−−− filtreg
|           |−−− dudv2
|           |−−− advect
|− iniwrite
|− writedem
```

6.2.4 banque d'etats initiaux: start2archive.F

```
start2archive
   |- lectba
   |- readfi
   |        |--- tabfi
   |- iniconst
   |        |--- disvert
   |- inigeom
   |        |--- fxyhyp
   |- inifilr
   |        |--- inifgn
   |                   |--- jacobi
   |                   |--- acc
   |                   |--- eigen_sort
   |                   |--- jacobi
   |                   |--- acc
   |                   |--- eigen_sort
   |- exner
   |        |--- filtreg
   |- covnat
   |- wind_scal
   |- multipl
   |- gr_fi_dyn
   |- write_archive
```

6.3 Compilation du modèle

Le modèle est compilé au moyen de l'utilitaire MAKE de UNIX. Le fichier makefile qui décrit comment la compilation doit se faire est créé automatiquement par le script

`create_make_gcm`

Cet utilitaire recré le makefile quand necessaire, par exemple quand un fichier source a été rajouté ou enlevé depuis la précédente compilation.

Tout ceci est transparent à l'utilisateur. Il lui suffit, pour compiler le modèle, de lancer la commande

`makegcm`

Ce script, dont le fonctionnement est expliqué dans le manuel ci-dessous, permet de compiler le modèle en faisant appel à la fonction make.
(Tapez `makegcm -h`)

Manuel de la fonction makegcm

```
-----------------------------------------------------
Par default, la commande makegcm:
---------------------------------

1. compile une serie de sous programmes se trouvant dans des sous-repertoires
de $LMDGCM/libf
2. Ensuite, makegcm compile le programme prog.f se trouvant par default sur
$LMDGCM/libf/dyn3d et effectue le lien avec l'ensemble des librairies.
```

```
La commande makegcm est faite pour permettre de gerer en parallele des
versions differentes du modele, compilees avec des options de compilation
et des dimensions differentes sans avoir a chaque fois a recompiler tout le
modele.

Les librairies FORTRAN sont stoquees sur le directory $LIBOGCM.


OPTIONS:
--------

Les options suivantes peuvent etre definies soit par defaut en editant le
"script" makegcm, soit en interactif:

-d imxjmxlm  ou im, jm, et lm sont resp. le nombre de longitudes, latitudes
             et couches verticales.

-t ntrac   selectionne le nombre de traceur advectes par la dynamique.
           Dans les versions courantes du modele terrestre on a par exemple
           ntrac=2 pour l'eau vapeur et liquide

             L'effet des options -d et -t est d'ecraser le fichier
             $LMDGCM/libf/grid/dimensions.h
             qui contient sous forme de 4 PARAMETER FORTRAN les 3 dimensions
             de la grille horizontale im, jm, lm plus le nombre de traceurs
             advectes passivement par la dynamique ntrac, par un nouveau fichier
             $LMDGCM/libf/grid/dimension/dimensions.im.jm.lm.tntrac
             Si ce fichier n'existe pas encore, il est cree par le script
             $LMDGCM/libf/grid/dimension/makdim

-p PHYS    pour selectionner le jeu de parametrisations physiques avec
           lequel on veut compiler le modele.
           Le modele sera alors compile en prenant les sources des
           parametrisations physiques dans le repertoire:
            $LMDGCM/libf/phyPHYS

-g grille  selectionne le type de grille qu'on veut utiliser.
           L'effet de cette option est d'ecraser le fichier
           $LMDGCM/libf/grid/fxyprim.h avec le fichier
           $LMDGCM/libf/grid/fxy_grille.h
           La grille peut prendre les valeurs:
           1. reg pour la grille reguliere
           2. sin pour avoir des points equidistants en sinus de la latitude
           3. new pour pouvoir zoomer sur une partie du globe

-O "optimisation fortran" ou les optimisations fortran sont les options de la
            commande f77

-include path
           Dans le cas ou on a dans des sous programmes des fichiers
           #include (cpp) qui se trouve sur des repertoires non references
           par defaut
```

```
-adjnt     Pour compiler la l'adjoint du code dynamique

-filtre  filtre
           Pour choisir le filtre en longitude dans les regions polaires.
           "filtre" correspond au nom d'un repertoire se trouvant sur
           $LMDGCM/libf. Le filtre standard du modele est "filtrez" qui peut
           etre utilise aussi bien pour une grille reguliere que pour une
           grille zoomee en longitude.

-link "-Ldir1 -lfile1 -Ldir2 -lfile2 ..."
          Pour rajouter un lien avec les librairies FORTRAN
          libfile1.a, libfile2.a ... se trouvant respectivement sur les
          repertoires dir1, dir2 ...
          Si dirn est un repertoire dont le chemin est automatique (comme
          par exemple /usr/lib ...) il n'est pas besoin de specifier -Ldirn.

Auteur: Frederic Hourdin  (hourdin@lmd.jussieu.fr)
-------------------------------------------------------
```

Part II

Faire tourner le modèle

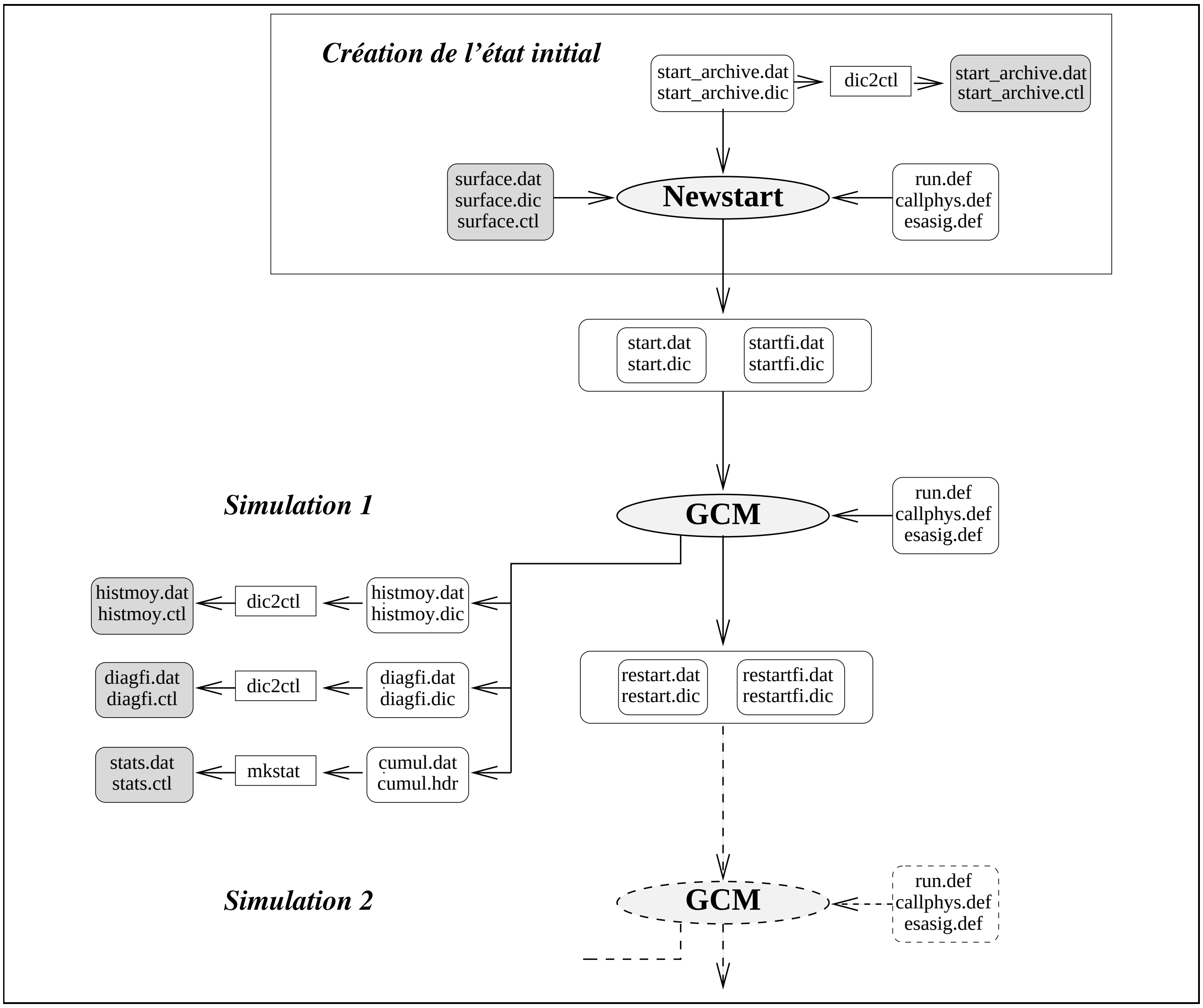

Figure 6.1: Entrées/Sorties

Chapter 7

première simulation

Ce chapitre est destinné aux personnes n'ayant jamais utilisé le modèle du LMD. Le meilleur moyen de prendre contact étant sans doute de réaliser une simulation, voici la marche à suivre. A partir de cet exemple, vous pourrez ensuite modifier à votre grés aussi bien les paramètres de controle du run que les conditions initiales. Vous trouverez dans le répertoire /dw/chourdin/MARS/version_en_cours/GCM/ tous les fichiers nécessaires à l'installation, la compilation et l'execution du modèle martien.

Pour le modèle terrestre, vous pouvez installer automatiquement le modèle dans votre home directory en allant exécuter la commande
EXPORTE_GCM
qui se trouve dans le répertoire /users/lmdz/ (lire la partie "Installer le modèle").

De plus, une démo est disponible sur /users/lmdz/DEMO/. Ce répertoire contient des fichiers de
"compilation",
"initialisation",
"simulation".
Il existe différentes versions de ces fichiers: une version standard, une version spécifique pour le cray, et une version pour les traceurs (_tr). Cette démo permet d'effectuer les trois étapes décrites ci après: Compiler le modèle, l'état initial, et lancement d'une simulation (faire tourner le modèle).

7.1 Installer le modèle

- Recopiez chez vous GCM le répertoire de base du modèle stocké dans /dw/chourdin/MARS/version_en_cours/GCM/ (Son contenu est décrit section 6). Executez pour cela la commande
  EXPORTE_GCM
  qui se trouve dans le répertoire /users/lmdz/.

- Affectez les variables d'environnement pour le modèle:

  LMDGCM Path du répertoire où vous avez installé le modèle (Path complet de GCM)

  `setenv  LMDGCM $PATH1/GCM`

  LIBOGCM Path du répertoire (ce répertoire est à créer: *libo* par exemple) dans lequel seront stockées les librairies objets créées lors de la compilation du modèle.

```
setenv  LIBOGCM $PATH2/libo
```

- Installez DRS et affectez la variable d'environnement DRSDIR:

  La dernière version de DRS est disponible dans le répertoire /dw/chourdin/MARS/drs/. Elle contient à la fois de quoi créer la librairie objet *libdrs.a*, les fichiers includes nécessaires, et de quoi compiler l'éditeur DRS *drsed*.

  Afin que le modèle, lors de la compilation, trouve les librairies et includes correspondant au type de machine utilisé, il convient de les placer dans un répertoire portant le nom des machine: **HP**, **SUN**, **IBM** ou **CRAY** sont les noms correspondant aux differentes platformes reconnues par le modèle. Ces répertoires doivent se trouver dans le répertoire pointé par la variable d'environnement **DRSDIR**.

  DRSDIR Nom du répertoire contenant librairie objet (libdrs.a) et fichier include (drsdef.h) DRS pour les différentes plateformes.

  ```
  setenv  DRSDIR $PATH3/drs
  ```

- Installez GrAdS

  Pour les gens travaillant au LMD, il suffit, grâce au brillantissime Laurent Fairhead, pour avoir accès à GrAdS et à son environnement, d'ajouter dans son *.cshrc* (c'est a priori fait dans *.env_soft*):

  ```
  source /distrib/local/grads/grads.env
  ```

- Compilez **dic2ctl**

  Les sources de ce programme sont disponibles dans le répertoire /dw/chourdin/MARS/version_en_cours/dic2ctl/.

- Pour le modèle Martien, Compilez en plus **mkstat**

  Les sources de ce programme sont disponibles dans le répertoire /dw/chourdin/MARS/version_en_cours/mkstat/.

- Recopiez **makegcm** le script pour la compilation (dans $HOME/bin par exemple). Ce script est disponible dans /dw/chourdin/MARS/version_en_cours/.

- Enfin, assurez vous que vous avez bien accès à tous les executables nécessaires pour l'utilisation du modèle. Pensez a indiquer le chemin correspondant dans le PATH.

  - fonction UNIX make
  - le compilateur Fortran f77
  - drsed
  - dic2ctl
  - mkstat (pour modèle martien uniquement)
  - grads

7.2 Compiler le modèle

- Pour compiler par exemple le modèle martien en résolution 64x48x25, tapez (conformément au manuel de la fonction makegcm décrit section 6.3)

```
makegcm -d 64x48x25 -p mars gcm
```

On récupère l'executable **gcm.e** (le modèle compilé) dans le répertoire où on a lancé la commande makegcm.

7.3 Les fichiers d'entrée (état initial)

- Récuperez dans le répertoire */dw/chourdin/MARS/version_en_cours/file.def/* le fichier de paramètres **run.def** (décrits section 5.2) necessaire au modèle. Récupérez en plus les fichiers **esasig.def et callphys.def** pour le modèle martien. Les paramètres sont affectés par des valeurs usuelles, et il est conseillé de réaliser une première simulation sans les modifier.

- Recopiez les fichiers **start et startfi** (décrits section 5.2) (ainsi que les fichiers **limit.dat et limit.dic** pour le modèle terrestre) stockés dans le répertoire */dw/chourdin/MARS/start/*. Ils contiennent des conditions initiales de scénario standard et permettront de valider l'installation du modèle.

7.4 Faire tourner le modèle

Placez dans un même répertoire le programme **gcm.e** et les fichiers d'entrée, puis tapez:

```
gcm.e
```

7.5 Visualisation des fichiers de sortie

Si vous n'avez jamais utilisé le logiciel graphique **GrAds**, il est vivement conseillé de consacrer une petite demi-heure à se familiariser avec lui en effectuant la demonstration prévue à cet effet. Facile, rapide, elle permet d'utiliser les commandes de base. Pour cela, allez lire le fichier

```
/distrib/local/grads/sample
```

7.5.1 Visualisation des fichiers histmoy, diagfi, histins...

On peut transformer les fichiers DRS histmoy, diagfi, histins... au format GrAdS à l'aide de l'utilitaire **dic2tl**. Il suffit de taper la commande, puis d' entrer le nom du fichier que l'on souhaite transformer. On obtient alors un fichier d'extention (**.ctl**) pour les variables temporelles (les vents, la temperature...), et un fichier d'extention (**.head**) pour les variables non temporelles telles que les conditions de surface.

Pour visualiser par exemple la carte de la temperature dans la 5eme couche à partir du fichier histmoy:

- Creation des fichiers histmoy au format GrAdS:

  `dic2ctl` *return*

`Fichier a lire ? : histmoy` *return*

`ls` *return* (pour verifier que les fichiers histmoy.ctl et histmoy.head ont bien été créés)

- Visualisation sous GrAdS:

`grads` *return*

return (pour ouvrir une fenêtre landscape)

`ga-> open histmoy` (grads ouvre par default le fichier de contrôle d'extension **.ctl**. tapez *open histmoy.head* pour visualiser les variables non temporelles)

`ga-> query file` (display des infos sur le fichier ouvert, dont le nom des variables stockées. Racourcis: *q file*)

`ga-> set z 5` (fixe l'altitude à la 5eme couche)

`ga-> set t 1` (fixe le temps à la premiere valeur stockée)

`ga-> query dims` (Indique les valeurs fixées pour les 4 dimensions. Racourcis: *q dims*)

`ga-> display T` (display la carte de temperature pour la 5eme couche et pour la premiere valeur de temps stockée. Racourcis: *d T*)

`ga-> clear` (pour effacer l'ecran. Racourcis: *c*)

`ga-> set gxout shaded` (mode d'affichage plein)

`ga-> display T`

`ga-> set gxout contour` (retour au mode contour pour afficher les niveaux)

`ga-> display T` (superpose les contours si on ne repasse pas par la commande clear)

7.5.2 Visualisation des fichiers cumul/stats

- Creation des fichiers **stats** (fichiers **cumul** au format GrAdS):

`mkstat` *return*

```
fichiers entree de type: cumula3m5.dat et cumula3m5.hdr
fichiers sortie de type: statsa3m5.dat et statsa3m5.ctl
entrer l'annee et le mois sous la forme a3m5
```

return (le fichier s'appelle cumul)

`ls` *return* (pour verifier que le fichier stats.ctl a bien été créé)

- Visualisation sous GrAdS:

Voir l'exemple de Visualisation sous GrAdS des fichiers DRS section 7.5.1

7.6 relancer la simulation

A la fin d'une simulation, le modèle génère des fichiers **restart** qui contiennent l'état final du modèle. Ces fichiers (au même format que les fichiers start) peuvent donc servir de conditions initiales à une nouvelle simulation.

Il suffit de les renommer:

mv restart.dat start.dat
mv restart.dic start.dic
mv restartfi.dat startfi.dat
mv restartfi.dic startfi.dic

et de relancer la simulation.

Chapter 8

Changer les conditions initiales

Bibliography

[1] Y. Fouquart and B. Bonnel. Computations of solar heating of the Earth's atmosphere: A new parametrization. *Contrib. Atmos. Phys.*, 53:35–62, 1980.

[2] J. R. Holton. *An introduction to dynamic meteorology*, volume 23 of *Internal geophysics series*. Academic Press, second edition, 1979.

[3] F. Hourdin. A new representation of the CO_2 15μm band for a Martian general circulation model. *J. Geophys. Res.*, 97(E11):18,319–18,335, 1992.

[4] I. Jacobsen and E. Heise. A new economical method for the computation of the surface temperature in numerical models. *Contrib. Atmos. Phys.*, 55:128–141, 1982.

[5] P. B. James, G. Briggs, J. Barnes, and A. Spruck. Seasonal recession of Mars' south polar cap as seen by Viking. *J. Geophys. Res.*, 84(B6):2889–2922, 1979.

[6] P. B. James, H. H. Kieffer, and D. A. Paige. The seasonal cycle of carbon dioxide on Mars. In *Mars*, pages pp. 934–968, Tucson, 1992. University of Arizona Press, Tucson.

[7] P. Le Van. Description de directives pour l'utilisation de la nouvelle dynamique du Modèle de Circulation Générale. Technical Report 147, Laboratoire de Météorologie Dynamique du CNRS, 24 rue Lhomond/75 231 Paris CEDEX 05, Note interne 1989.

[8] J.-F. Louis. A parametric model of vertical eddy fluxes in the atmosphere. *Boundary-layer Meteorol.*, 17:187–202, 1979.

[9] G. L. Mellor and T. Yamada. A hierarchy of turbulence closure models for planetary boundary layers. *J. Atmos. Sci.*, 31:1791–1806, 1974.

[10] J. J. Morcrette, L. Smith, and Y. Fouquart. Pressure and temperature dependence of the absorption in longwave radiation parametrizations. *Contrib. Atmos. Phys.*, 59(4):455–469, 1986.

[11] D. A. Paige and A. P. Ingersoll. Annual heat balance of Martian polar caps: Viking observations. *Science*, 228:1160–1168, 1985.

[12] J. B. Pollack, D. S. Colburn, F. M. Flasar, R. Kahn, C. E. Carlston, and D. Pidek. Properties and effects of dust particles suspended in the Martian atmosphere. *J. Geophys. Res.*, 84(B6):2929–2945, 1979.

[13] J. B. Pollack, R. M. Haberle, J. Schaeffer, and H. Lee. Simulations of the general circulation of the Martian atmosphere, 1, Polar processes. *J. Geophys. Res.*, 95:1447–1473, 1990.

[14] J. B. Pollack, C. B. Leovy, P. W. Greiman, and Y. Mintz. A Martian General Circulation Model experiment with large topography. *J. Atmos. Sci.*, 38:3–29, 1981.

[15] S. G. Warren, W. J. Wiscombe, and J. F. Firestone. Spectral albedo and emissivity of CO_2 in Martian polar caps: model results. *J. Geophys. Res.*, 95:717–741, 1990.

[16] D. A. Warrilow, A. B. Sangster, and A. Slingo. Modeling of land surface processes and their influence on European climate. Technical Report DCTN 38, UKMO, Meteorological Office, London Road, Bracknell, Berkshire RG12 2SZ, 1986.