

# **Code d'évolution stellaire CESAM2*k***

## **Aide mémoire : Installation, Utilisation**

Pierre Morel  
OCA Observatoire de Nice, BP. 4229, 06304 Nice Cedex 4  
UMR CNRS 6529

Version 3.12.15, 14 Novembre 2017.

*Quand un jour, tôt ou tard, il faut qu'on disparaisse,  
Quand on a plus ou moins vécu, souffert, aimé  
Il ne reste de soi que les enfants qu'on laisse  
Et le champ de l'Effort où l'on aura semé.*

Ch. De Gaulle

# Table des matières

<b>1</b>	<b>Installation</b>	<b>1</b>
1.1	CESAM2 <i>k</i> . . . . .	1
1.1.1	PGPLOT . . . . .	2
1.1.2	Variables quasi-statiques utilisées à partir de la version V3.5.0. . . . .	2
1.1.3	Particularités de la présente version . . . . .	3
1.2	Contenu de la distribution . . . . .	3
1.3	Installation . . . . .	4
1.4	Adaptation de CESAM2 <i>k</i> à l'environnement . . . . .	5
1.5	Création des exécutables . . . . .	5
1.6	Premier test de fonctionnement . . . . .	6
<b>2</b>	<b>SCRIPTS</b>	<b>13</b>
<b>3</b>	<b>SUN_STAR_DATA</b>	<b>15</b>
3.1	Opacités. . . . .	15
3.1.1	Opacités opa_yveline.f, opa_yveline_lisse.f . . . . .	15
3.1.2	Opacités opa_houdek12.f . . . . .	16
3.2	Equations d'état. . . . .	18
3.2.1	Equation d'état etat_opal.f . . . . .	18
3.2.2	Equations d'état etat_opalX.f, etat_opalZ.f . . . . .	20
3.2.3	Equation d'état etat_opal5.f . . . . .	21
3.2.4	Equation d'état etat_mhd.f . . . . .	21
3.2.5	Equation d'état analytique etat_irwin.f . . . . .	23
3.2.6	Equation d'état etat_saha.f . . . . .	24
3.3	Lois $T(\tau)$ . . . . .	25
3.3.1	Lois $T(\tau)$ MARCS . . . . .	25
3.3.2	Lois $T(\tau)$ PIAU . . . . .	25
3.3.3	Lois $T(\tau)$ Roger . . . . .	25
3.4	Divers . . . . .	25
3.4.1	Package Lapack . . . . .	25
3.4.2	Modèles ASCII d'initialisation PMS et ZAMS homogènes . . . . .	26
3.4.3	Données pour accélérations radiatives . . . . .	27
<b>4</b>	<b>Exploitation</b>	<b>29</b>
4.1	Conseils pour l'exploitation . . . . .	29
4.2	Création d'un espace de travail . . . . .	29
4.3	Le fichier de données : mon_modele.don . . . . .	30

4.4	Exemple de fichier de données . . . . .	31
4.5	Arguments de la NAMELIST NL_CESAM . . . . .	33
4.6	Arguments de la NAMELIST NL_MASS . . . . .	37
4.7	Arguments de la NAMELIST NL_EVOL . . . . .	38
4.7.1	Caractérisation des types de modèles . . . . .	38
4.8	Arguments de la NAMELIST NL_CHIM . . . . .	40
4.8.1	Alternative : conservation métal/H ou métal/Z . . . . .	41
4.9	Arguments de la NAMELIST NL_CONV . . . . .	42
4.10	Arguments de la NAMELIST NL_DIFF . . . . .	43
4.11	Arguments de la NAMELIST NL_ROT . . . . .	44
4.12	Arguments de la NAMELIST NL_ETAT . . . . .	45
4.13	Arguments de la NAMELIST NL_OPA . . . . .	46
4.14	Arguments de la NAMELIST NL_NUC . . . . .	47
4.15	Arguments de la NAMELIST NL_ATM . . . . .	48
4.16	Exploitation sous UNIX & LINUX . . . . .	49
4.16.1	Fichiers ASCII et binaires d'initialisation . . . . .	50
4.16.2	Exemple d'évolution avec PMS . . . . .	50
4.16.3	Limitation des sorties . . . . .	53
4.17	Contenu du package EXPLOIT . . . . .	53
4.17.1	Outils astrophysiques . . . . .	54
4.17.2	Programmes d'exploitation . . . . .	55
4.17.3	Shell-scripts d'exploitation . . . . .	55
4.17.4	Données diverses . . . . .	56
<b>5</b>	<b>Personnalisation</b> . . . . .	<b>59</b>
5.1	Langue . . . . .	60
5.2	Mixture . . . . .	60
5.3	Rapports isotopiques . . . . .	61
5.4	Personnalisation des rapports d'abondances . . . . .	62
5.5	Personnalisation de la composition chimique du vent. . . . .	62
5.6	Chute de planétoïdes . . . . .	63
5.7	Personnalisation des paramètres de free-eos . . . . .	64
5.8	Personnalisation du fichier Ascii . . . . .	64
5.9	Réglages . . . . .	65
5.10	Limites des shells source . . . . .	68
5.11	Zoom . . . . .	69
5.12	Device . . . . .	71
5.13	Limitation des écritures . . . . .	72
<b>6</b>	<b>Extensions</b> . . . . .	<b>73</b>
6.1	Ajout d'une chaîne de réactions thermonucléaires . . . . .	73
6.2	Ajout d'une mixture initiale . . . . .	74
<b>7</b>	<b>Debug</b> . . . . .	<b>75</b>
7.1	Debug . . . . .	75
7.2	Bugs connus . . . . .	77

<b>A</b>	<b>Liste des fichiers ASCII</b>	<b>83</b>
A.1	Fichiers pour oscillations . . . . .	83
A.1.1	Fichier pour oscillations adiabatiques . . . . .	83
A.1.2	Fichier pour oscillations non adiabatiques . . . . .	85
A.1.3	Fichier pour inversions . . . . .	86
A.1.4	Utilisation du nombre maximum de couches . . . . .	87
A.2	Fichier pour diagramme HR . . . . .	88
A.2.1	Fichier simplifié pour diagramme HR . . . . .	90
A.3	Fichier ASCII des variables de la diffusion du moment cinétique . . .	90
A.4	Fichier ASCII des variables de la zone d'instabilité des Céphéïdes . .	91
A.5	Fichier ASCII d'extension <code>_TR.csv</code> . . . . .	91
A.6	Fichier ASCII d'extension <code>_LB.csv</code> . . . . .	92



# Table des figures

1.1	Schéma de l'organigramme général de CESAM2 <i>k</i> . Le programme <code>cesam2k</code> , compilé une fois pour toutes, fait appel à la routine <code>cesam</code> qui constitue en fait, le programme principal. Dans <code>cesam</code> , après des initialisations, on sépare le traitement des 3 possibilités : i) poursuite d'une évolution, ii) initialisation sur la ZAMS homogène, iii) initialisation d'une PMS. Le calcul se poursuit par des opérations de gestion de l'évolution, listing, création de fichiers de sortie, dessin, etc... Est ensuite fait appel à <code>resout</code> , où les équations de l'équilibre quasi-statique sont résolues en alternance avec celles de l'évolution du moment cinétique et de la composition chimique. Ces dernières sont résolues séparément dans <code>evol</code> . Une fois la convergence obtenue, il y a retour vers les algorithmes de gestion de l'évolution de <code>cesam</code> et, de là, poursuite de l'évolution ou retour à <code>cesam</code> pour sortie. Les routines d'analyse numérique et de physique sont sollicitées à différents niveaux de calcul. . . . .	11
1.2	Tracés évolutifs incluant la pré-séquence principale pour des étoiles de $0.5M_{\odot}$ à $15M_{\odot}$ . Le flash de l'hélium a été dépassé pour les masses supérieures à $3M_{\odot}$ . Pour les modèles de $3M_{\odot}$ et de $5M_{\odot}$ les calculs ont été arrêtés après la disparition du carbone au centre. La phase de destruction de l'oxygène a été atteinte pour les modèles de $7M_{\odot}$ et de $10M_{\odot}$ . Avec la disparition de l'oxygène dans le noyau, la limite de validité de l'hypothèse d'équilibre quasi-statique a été atteinte pour le modèle de $15M_{\odot}$ . Ces tracés ont été obtenus avec le fichier de données du § 4.4 (Page 31). . . . .	12
3.1	Domaines des EOS OPAL2001, OPAL2005 et SAHA-S. Graphe communiqué par A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State University. . . . .	22
7.1	Profil de la fréquence de Brunt-Väissälä d'un modèle de $10M_{\odot}$ de séquence principale ( $X_c = 0.35$ ), calculé avec la procédure décrite § 16 (Page 80). . . . .	76



# Liste des tableaux

4.1	Valeurs des paramètres numériques de la <code>precision</code> par défaut ( <code>pr/PR</code> ) et valeurs adaptées aux autres précisions. Les dénominations s'interprètent de la façon suivante : <code>pr/PR</code> : précision réaliste (" <b>le défaut</b> "), <code>sp/SP</code> : super précision, <code>lm/LM</code> : pour petites masses (low mass), <code>av/AV</code> : pour modèles pouvant atteindre les stades avancés, <code>np/NP</code> : pour modèles de précision normale, <code>co/CO</code> : pour modèles destinés à l'exploitation des résultats du satellite CoRoT, <code>sa/SA</code> : pour modèles solaires. Pour les Céphéides utiliser le type de précision <code>ce/CE</code> , analogue à <code>av/AV</code> , et produisant des fichiers d'exploitation spécifiques. La signification des différents paramètres numériques est explicitée au § 5.9 (Page 65) et lignes 805-1237 dans la routine <code>cesam.f</code> du sous-directory <code>SOURCE</code> . La valeur initiale du pas temporel et ses limites de variations ultérieures dépendent de la masse et du type de précision requis; les valeurs portées dans ce tableau n'ont qu'une valeur indicative. L'utilisation des minucules/MAJUSCULES (Exemple : <code>pr/PR</code> ) du type de précision définit le type de variables quasi-statiques à utiliser, cf. § 4.5 (Page 33). Une astérisque* signale ces paramètres peuvent être modifiés lorsque la masse initiale diffère de l'unité ou que la diffusion microscopique est prise en compte. . . . .	36
4.2	Valeurs des paramètres numériques (suite). . . . .	37

# Index

ZFS\_interp\_EOS5.f, 21

sa, 36

A, 35

abondance d'hélium, 62

abondances, 62

abondances initiales, 40, 62

accélération radiatives, 27, 43

add\_Al, 62

add\_Ar, 62

add\_B, 62

add\_Be, 62

add\_C, 62

add\_Ca, 62

add\_Cl, 62

add\_Co, 62

add\_Cr, 62

add\_F, 62

add\_Fe, 62

add\_K, 62

add\_Li, 62

add\_Mg, 62

add\_Mn, 62

add\_N, 62

add\_Na, 62

add\_Ne, 62

add\_Ni, 62

add\_O, 62

add\_P, 62

add\_S, 62

add\_Sc, 62

add\_Si, 62

add\_Ti, 62

add\_V, 62

add\_Z, 62

Adelb, 48

age\_car, 67

agemax, 38

aide mémoire, 3

ajout de mixture, 74

ajout isotope, 73

ajuste, 68

alecian1.f, 43

alecian2.f, 43

all\_adia, 34

all\_ascii, 34

all\_invers, 34

all\_iter, 45

all\_mod, 45

all\_nadia, 34

alpha, 42

anglais, 60

arrêt de l'évolution, 80

arret, 38

ascii.f, 64

ascii2bin\_opa.f, 15, 16

atmosphère, 79

AV, 35

av, 36

baratine, 53, 72

bashrc, 5

big-endian, 17

blabla, 53, 72

blas, 23

Brunt-Väissälä, 80

bugs, 77

C-shell, 4

calib2k\_pms, 55

calib2k\_pms.f, 55

calib2k\_zams.f, 55

calibration solaire, 41

Cau-Fow, 48

CE, 35

cepheide, 35, 91

cesam2k.out, 1, 6, 49

co, 36

coca, 38

- cohe, 38
- compilateurs, 1
- cone, 38
- cons\_glob\_mnt\_cin, 44
- cons\_loc\_mnt\_cin, 44
- conseils, 29
- conv\_cgm, 49
- convergence forcée, 80
- coox, 38
- cpturb, 42
- create\_evol\_space, 29
- cshrc, 4
- ctel, 67
- ctem, 67
- ctep, 67
- cter, 68
- ctes\_31, 33
- ctes\_85.f, 33
- ctes\_94.f, 33
- ctes\_94m.f, 33
- ctes\_ba, 33
- ctes\_gaia, 33
- ctet, 68
  
- définition de la ZAMS, 50
- d\_lum, 67
- d\_press, 67
- d\_ray, 67
- d\_temp, 67
- d\_turb, 43
- d\_zc, 68
- dell, 10
- des.f, 4
- des2k\_abon.f, 55
- des2k\_abontc.f, 55
- des2k\_abonts.f, 55
- des2k\_bin.f, 55
- des2k\_dhve.f, 55
- des2k\_diff\_spl.f, 55
- des2k\_grad.f, 55
- des2k\_hr.f, 55
- des2k\_opa.f, 55
- des2k\_osc.f, 55
- des2k\_ZC.f, 55
- des\_instab, 68
- des\_m.f, 5, 33
- des\_mC, 33
  
- des\_r.f, 4, 5, 33
- dessin, 10
- device, 71
- diff\_mz04, 44
- diff\_tz97, 44
- difficultés, 77
- diffm\_0, 43
- diffm\_br.f, 43
- diffm\_mp.f, 43
- diff\_t\_gab.f, 43
- diff\_t\_nu.f, 43
- diff\_t\_nut.f, 43
- diff\_t\_smc.f, 43
- diffusion, 43
- diffusion microscopique, 43
- diffusion turbulente, 43
- diffw\_0.f, 44
- diffw\_mpz.f, 44
- diffw\_p03.f, 44
- diffw\_toul.f, 44
- divergence, 78
- dn\_fixe, 68
- données, 1, 15
- dpsi, 68
- dt0, 67
- dtlist, 38
- dtmax, 67
- dtmin, 67
- dws, 44
  
- edding.f, 48
- en\_masse, 67
- end\_evol, 45
- end\_mod, 45
- Endianness, 17
- enhan\_al, 40
- enhan\_cha, 40
- enhan\_w, 40
- equation d'état, 46
- espace de travail, 29
- etat\_eff.f, 46
- etat\_gong1.f, 46
- etat\_gong2.f, 46
- etat\_irwin.f, 23, 46, 64
- etat\_irwin\_explik, 46
- etat\_mhd.f, 21, 46
- etat\_opal.f, 18, 20, 46

- etat\_opal5, 21, 46
- etat\_opalX.f, 20, 46
- etat\_opalZ.f, 20, 46
- etat\_saha, 24, 46
- etat\_saha.f, 24
- evol2k\_pms, 55
- evol2k\_zams, 55
- exe2k, 4
- EXPLOIT, 1
- extension .hom, 50
- extension .mix, 60
- extension .pms, 50
- extension .zams, 50
  
- f\_eos, 45
- f\_opa, 46
- fcv, 68
- fichier de données, 31
- fichier de sorties ASCII, 34
- fichier\_vent.f, 55
- fichiers d'initialisation, 50
- fichiers d'oscillations, 83
- FORTTRAN95, 1
- free\_eos, 23
  
- gain de masse, 62
- garde\_xish, 40, 41
- genere\_cesam2k, 4
- grille\_fixe, 40
  
- H12, 16, 46
- He, 35
- he\_core, 38
- HHe, 33, 68
- hom, 38
- hopf.f, 48
- hsra.f, 48
  
- ini0, 67
  
- journal, 2
- jours, 44
- jpz, 42
  
- k5750.f, 48
- k5777.f, 48
- kind, 3
- kipp, 67
- kms/s, 44
  
- L, 35
- 10, 68
- l\_fac, 68
- langue, 60
- lapack, 23, 25
- Ledoux, 42
- lim\_atm.f, 48
- lim\_gong1.f, 48
- lim\_ro, 48, 49
- lim\_tau1.f, 48
- limitation des écritures, 72
- limitation des sorties, 53, 72
- limites zones radiatives / zones convectives, 79
- link, 16
- LINUX, 1, 4, 49
- lisse, 68
- lit\_osc.f, 79
- little-endian, 17
- LM, 35
- lm, 35, 36
- lnP varie trop, 78
- loc\_zc, 67
- log\_teff, 38
  
- météorites, 40
- métal/H, 41
- métal/Z, 41
- m\_ch, 67
- m\_ptm, 67
- m\_qs, 67
- m\_rot, 67
- m\_tds, 67
- majuscules/minuscules, 35
- Make, 6
- MARCS, 25, 49
- MARCS.f, 48
- mdot, 37
- mitler, 47
- mixture, 60, 74
- mod\_atm.f, 3
- mod\_cesam.f, 3, 4
- mod\_conv.f, 3
- mod\_donnees.f, 3
- mod\_etat.f, 3
- mod\_evol.f, 3
- mod\_exploit.f, 3

- mod\_kind.f, 3
- mod\_nuc.f, 3
- mod\_numerique.f, 3
- mod\_opa.f, 3
- mod\_static.f, 3
- mod\_variables.f, 3
- modèle de ZAMS, 79
- modif\_chim, 40
- modif\_mix, 40
- modif\_xish, 62
- mon\_modele-ad.osc, 34, 83
- mon\_modele-ascii, 34
- mon\_modele-inv.osc, 34, 86
- mon\_modele-nad.osc, 34, 85
- mon\_modele.cephe, 35, 91
- mon\_modele.don, 1, 30
- mon\_modele.HR, 88
- mon\_modele.LR, 90
- mtot, 37
- mu\_saha, 68
  
- n\_atm, 67
- n\_max, 33
- n\_mi\_ZC, 68
- NACRE, 48
- NACRE+, 48
- NACRU, 48
- nb\_max\_modeles, 38
- new\_bv, 68
- NL\_ATM, 30, 48
- NL\_CESAM, 30, 33
- NL\_CHIM, 30, 40
- NL\_CONV, 30, 42, 79
- NL\_DIFF, 30
- NL\_ETAT, 30, 45
- NL\_EVOL, 38
- NL\_MASS, 30, 37
- NL\_MODIF\_MIX, 62
- NL\_NOMS, 33
- NL\_NUC, 30, 47
- NL\_OPA, 30, 46
- NL\_RLG, 66
- NL\_ROT, 30, 44
- NL\_TEMPS, 30, 80
- no\_des, 33, 45
- no\_discon, 68
- no\_frad, 43
- no\_output, 34
- nom\_abon, 31, 40
- nom\_atm, 31, 48
- nom\_chemin, 31, 33
- nom\_conv, 31
- nom\_ctes, 31, 33
- nom\_des, 31
- nom\_des\_rot, 45
- nom\_diffm, 31, 43
- nom\_diff\_t, 31, 43
- nom\_diff\_w, 45
- nom\_etat, 31, 46
- nom\_frad, 43
- nom\_nuc, 31, 47
- nom\_nuc\_cpl, 47, 48
- nom\_opa, 31, 46
- nom\_opa\_cond, 46
- nom\_output, 34
- nom\_perte, 31
- nom\_pertm, 37
- nom\_pertw, 45
- nom\_tdetau, 31, 48
- nom\_thw, 44
- nombre de couches, 87
- NOTICE, 1, 3
- NP, 35
- np, 35, 79
- npt\_lisse, 68
- numéro de version, 2
- numérotation des modèles, 34, 38
  
- opa\_cond\_iben, 46
- opa\_cond\_mestel, 46
- opa\_gong.f, 46
- opa\_houdek12, 16
- opa\_houdek12.f, 46
- opa\_houdek9, 16
- opa\_int\_zsx.f, 46
- opa\_opal2\_cno, 46
- opa\_opal2\_co, 46
- opa\_opalCO.f, 46
- opa\_yveline.f, 46
- opa\_yveline\_jorgen.f, 46
- opa\_yveline\_lisse.f, 46
- opacité, 15, 46
- opacité conductive, 46
- opal5Z\_ascii\_bin.f, 21

- opalX\_ascii\_bin.f, 20
- opalZ\_ascii\_bin.f, 20
- optimisation, 79
- option mixture, 60
- ordre, 67
- osc\_adia.f, 34
- osc\_invers.f, 34
- osc\_nadia.f, 34
- ovshti, 42
- ovshts, 42
  
- p\_pertw, 45
- perte de masse, 62
- pertm\$, 37
- pertm\_dominik.f, 37
- pertm\_ext.f, 37
- pertm\_msol.f, 37
- pertm\_reim-domi.f, 37
- pertm\_reimer.f, 37
- pertm\_tot.f, 37
- pertm\_waldron.f, 37
- pertw\_0, 45
- pertw\_loc.f, 45
- pertw\_ptm.f, 45
- pertw\_sch.f, 45
- PGPLOT, 2, 4
- pgplot\_factice, 2
- PIAU, 25
- piau1/3, 49
- planétoïdes, 63
- PMS, 10, 50
- pms, 38
- post, 38
- pp1.f, 47
- pp3.f, 47
- ppcno10.f, 47
- ppcno10BeBFe.f, 47
- ppcno10Fe.f, 47
- ppcno10K.f, 47
- ppcno11.f, 47
- ppcno12.f, 47
- ppcno12Be.f, 47
- ppcno12BeBFe.f, 47
- ppcno12Li.f, 47
- ppcno3a12Ne.f, 47
- ppcno3a9.f, 47
- ppcno3aco.f, 47
  
- ppcno3acos.f, 47
- ppcno9.f, 47
- ppcno9Fe.f, 47
- PR, 35
- pr, 35, 36
- precic, 67
- precision, 35
- precit, 67
- precix, 67
- pression turbulente, 79
- psi0, 67
  
- q0, 68
  
- réactions thermonucléaires, 47, 73
- réglages, 65
- R\_stop, 38
- rad/s, 44
- rapport des abondances, 62
- rappports isotopiques, 61
- re\_nu, 43
- rep, 38
- restitution de l'atmosphère, 48
- rg, 35, 36, 66
- roger00.f, 48
- roger02.f, 48
- roger05.f, 48
- roger10a.f, 48
- rot\_0, 44
- rot\_cte, 44
- routine de dessin, 33
- routine de sorties ASCII, 34
- routines disponibles, 33
  
- SA, 35
- sa, 35
- SCRIPTS, 1
- semi convection, 79
- semi\_convection, 43
- shells source, 68
- solaire\_ags\_03, 40
- solaire\_ags\_05, 40
- solaire\_gn, 40
- solaire\_gs, 40
- sorties ASCII, 34
- SOURCE, 1
- SP, 35

sp, 35, 36  
suffix V, 43  
suffixe de convection S, 42  
suffixe de diffusion V, 43  
suffixe de précision A, 38  
suffixe de precision 0, 35  
suffixes, 32  
suffixes de dessin C, HHe & S, 33  
suffixes de output A, T & L, 34  
suffixes de précision A, L & M,  
35  
SUN\_STAR\_DATA, 1, 15  
  
t\_stop, 38  
tams, 38  
tau\_max, 48, 49  
TESTS, 1  
  
unit, 44  
UNIX, 1, 4, 49  
  
vent.f, 62  
  
w\_rot, 44  
WINDOWS, 1  
  
x0, 40  
x\_stop, 38  
  
y0, 40  
  
ZAMS, 6  
zams, 38  
zoom, 69  
zsx0, 40



# Chapitre 1

## Installation

*On n'exécute pas tout ce qui se propose  
Et le chemin est long du projet à la chose.*

Le Tartuffe.

**Pour extraire la version actuelle de CESAM2*k* du "trunk" de SVN, saisir :**

```
svn co https://svn.oca.eu/codes/CESAM2k/trunk/EXPLOIT
svn co https://svn.oca.eu/codes/CESAM2k/trunk/NOTICE
svn co https://svn.oca.eu/codes/CESAM2k/trunk/SCRIPTS
svn co https://svn.oca.eu/codes/CESAM2k/trunk/SOURCE
svn co https://svn.oca.eu/codes/CESAM2k/trunk/SUN_STAR_DATA
svn co https://svn.oca.eu/codes/CESAM2k/trunk/TESTS
```

Pour afficher cette partie de la notice, se placer dans le directory NOTICE, saisir Make puis `aide_mem2k.pdf`.

### 1.1 CESAM2*k*

L'exploitation de CESAM2*k* repose l'exécutable, `cesam2k.out`, créé une fois pour toutes. C'est dans un fichier de données, d'extension générique ".don", que la physique à utiliser est indiquée conjointement aux principaux paramètres définissant le modèle à calculer ; un exemple est donné au § 1.6 (Page 6). Il n'est donc pas nécessaire de créer un exécutable adapté à chaque utilisation<sup>1</sup>. Il est toutefois possible de modifier la plupart des options, numériques ou physiques retenues, grâce à des fichiers dits de "réglage" décrits au § 5 (Page 59). Ils permettent d'adapter, de façon externe CESAM2*k* aux moyens de calcul dont on dispose et, dans une large mesure, aux particularités physiques des modèles à calculer. Sans pour autant avoir recours au fichier de réglage, il est possible d'adapter certains paramètres du fichier de données comme, par exemple, fixer à son maximum le nombre de couches à utiliser, cf. § 4.5 (Page 33).

CESAM2*k* est programmé en FORTRAN95, ce qui permet un dimensionnement dynamique des tableaux. Les compilateurs suivants ont été utilisés avec succès (liste non exhaustive) :

---

1. Sauf, bien sûr, pour des modifications, debug, etc...

- sous LINUX/UNIX : `ifc`, `ifort`, Lahey-Fujitsu `lf95`, `gfortran`,
- sous WINDOWS : Compact Visual Fortran

La signification du numéro de `VERSION i.j.k` est la suivante :

**i** augmente d'une unité à chaque modification du contenu des fichiers binaires de reprise `*.pms`, `*.rep`, `*.dat`, *etc...* les indices suivants **j** et **k** étant remis à 0.

**j** augmente d'une unité à chaque modification du contenu d'un des fichier ASCII de données `*.don` ou de personnalisation, l'indice suivant **k** est remis à 0.

**k** augmente d'une unité pour chaque ensemble de modifications ayant une conséquence sur les résultats.

Dans le sous-directory `SOURCE`, le fichier "`journal`" contient une description succincte de la chronologie des aménagements effectués.

### 1.1.1 PGPLOT

`CESAM2k` utilise le logiciel de dessin `PGPLOT` pour effectuer des dessins "on line" permettant de suivre le déroulement des calculs et aussi de faciliter le debug. La configuration de ce package n'est pas prévue avec le compilateur `ifc`. Thierry Corbard, Observatoire de la Côte d'Azur, a construit les procédures nécessaires que l'on trouvera dans le sous-directory `SCRIPTS`. Avec les compilateur plus modernes, cette difficulté a été contournée. Sous LINUX, la commande : `apt-get install pgplot5` permet d'effectuer l'installation sans difficulté.

Il est possible d'éviter l'utilisation de dessin "on line". Pour ce faire, avant la formation du module exécutable décommenter la dernière ligne du module `MOD_CESAM`. Le fichier `pgplot_factice.f` contient des émulations des routines de `PGPLOT`. Coder `NOM_DES='no_des'` et `NOM_DES_ROT='no_des'` dans le fichier de données *cf.* § 4.3 (Page 30). Dans les Makefiles `Make`, `Make_expl`, `Make_exe`, dans les instructions de link, supprimer les chemins à la bibliothèque `PGPLOT` ainsi que dans le fichier `.bashrc`.

### 1.1.2 Variables quasi-statiques utilisées à partir de la version V3.5.0.

A cause de difficultés coercitives et non résolues de convergence, la possibilité d'utiliser le rayon comme variable principale d'espace a été supprimée à partir de la version V3.5.0. Les variables d'espace correspondantes, dites "eulériennes",  $R$ ,  $L$ ,  $M$  ont été remplacées par l'ensemble  $R$ ,  $L$ ,  $M^{1/3}$  qui, tout en évitant la singularité centrale<sup>2</sup>, permet de décrire les phases d'évolutions avec une luminosité locale négative. Cette situation peut survenir lorsque les pertes par neutrinos deviennent importantes, particulièrement dans les phases avancées ; ou encore au cours du calcul avec une correction devenant *anormalement* excessive.

Les routines `static_m.f` et `static_r.f` ont donc été respectivement remplacées par `static_m23.f` et `static_m13.f`. Les modifications qui ont été nécessaires sont

---

2. C'est une remarque pertinente de L.Piau qui est à l'origine de cette amélioration significative.

décrites dans le fichier `journal.f` du directory `SOURCE`. L'ensemble initial de variables  $R^2$ ,  $L^{2/3}$ ,  $M^{2/3}$ , conservé dans `static.m23.f`, permet de calculer des modèles dans les mêmes conditions que précédemment ; l'alternative est implicitement indiquée dans le fichier de données.

### 1.1.3 Particularités de la présente version

Dans la présente version les particularités suivantes ont été introduites :

1. Remplacement du contrôle de la variation temporelle de la génération d'énergie gravifique par ceux de la pression et de la température locales.
2. Imposition du positionnement des discontinuités dues au mélange convectif au voisinage des points de grille (sauf dans les couches les plus externes).
3. Introduction de `subscripts` aux arguments du fichier de données permettant d'adapter le calcul à des contraintes particulières comme, par exemple la génération d'un fichier `.HR` condensé (extension `.LR`).
4. En vue de debug, possibilité de dessins des variables principales à l'issue de chaque pas temporel ou à chaque itération.

## 1.2 Contenu de la distribution

Le package `CESAM2k`<sup>3</sup> contient les cinq sous-directory :

- `EXPLOIT` : ensemble de programmes, de scripts shell, d'exemples ainsi que le `Makefile`<sup>4</sup> pour l'installation, l'exploitation et la création de données ou de paramètres relevant de la physique ou du numérique, par exemple le calcul de  $Fe/H$ , ou de la correction bolométrique ou encore de la calibration solaire.
- `NOTICE` : sources `LATEX` et fichiers postscript de la notice d'utilisation, de l'aide-mémoire et de leurs `Makefiles` d'installation.
- `SCRIPTS` : ensemble de procédures `UNIX/LINUX` pour l'installation et l'exploitation avec le shell `bash`.
- `SOURCE` : ensemble des routines, modules et programmes de la source et le `Makefile` d'installation.
- `SUN_STAR_DATA` : ensemble de données physiques, essentiellement des packages, des programmes pour l'exploitation de tables d'EOS et d'opacité.
- `TESTS` : divers programmes de test<sup>5</sup> comme, par exemple tester la mise en place d'une nouvelle EOS, ou encore d'un réseau de réactions nucléaires...

Les modules regroupés dans la `SOURCE` ont les fonctionnalités suivantes :

- Module `mod_kind` : définit les types des variables utilisées.
- Module `mod_numerique` : regroupe les routines purement numériques et les routines d'exploitation des données physiques.
- Module `mod_donnees` : regroupe la plupart des quantités fixes au cours de l'évolution.

---

3. 2k pour "2000".

4. Les `Makefiles` sont construits suivant les règles de GNU Make, version 4.0.

5. Certains de ces programmes devront être adaptés à la présente version de `CESAM2k`.

- Module `mod_variatives` : regroupe la plupart des quantités variables au cours de l'évolution.
- Module `mod_etat_saha` : regroupe les routines concernant l'équation d'état `etat_saha` créée par A.V.Oreshina & V.A.Baturin.
- Module `mod_etat` : regroupe les routines concernant l'équation d'état.
- Module `mod_opa` : regroupe les routines concernant le calcul de l'opacité.
- Module `mod_conv` : regroupe les routines concernant la convection.
- Module `mod_atm` : regroupe les routines de restitution de l'atmosphère.
- Module `mod_nuc` : regroupe les routines concernant les réactions thermonucléaires.
- Module `mod_bp_for_alecian` : regroupe les routines concernant le calcul des accélérations radiatives, suivant le formalisme de G.Alécian.
- Module `mod_evol` : regroupe les routines concernant l'évolution temporelle de la composition chimique.
- Module `mod_static` : regroupe les routines concernées par la résolution de l'équilibre quasi-statique.
- Module `mod_cesam` : regroupe les routines concernant la gestion du calcul des évolutions.
- Module `mod_exploit` : regroupe des routines concernées par l'exploitation des résultats.
- Module `mod_photo` : regroupe des routines concernées par la photométrie.
- Module `mod_sismo` : regroupe des routines concernées par la sismologie.
- Module `mod_spectro` : regroupe des routines concernées par la spectroscopie.

Chaque routine `PRIVATE` et/ou `PUBLIC` est introduite par un `INCLUDE` dans le fichier de son module d'appartenance; le nom de ce dernier est indiqué dans les commentaires au début de chaque routine.

L'organigramme général du code est schématisé § 1.1 (Page 11).

## 1.3 Installation

On décrit l'installation sous UNIX/LINUX. Il n'est pas exclu que des difficultés d'ordre informatique puissent survenir lors de l'installation. Chaque système ayant ses particularités et ses chausse-trappes, l'assistance d'un expert est parfois salutaire.

- Détarer la distribution :

```
tar -xzvf CESAM2k.tar.gz
```

Le sous-directory `CESAM2k` est créé<sup>6</sup>. Il contient 5 sous-directory qu'il convient aussi de détater :

```
EXPLOIT.tar.gz
NOTICE.tar.gz
SCRIPTS.tar.gz
SOURCE.tar.gz
SUN_STAR_DATA.tar.gz
TESTS.tar.gz
```

---

6. Dans cette notice les numéros de version et les indications de date sont volontairement ignorés.

- Le contenu des sous-directory `SUN_STAR_DATA` et `SCRIPT` a la vocation d'être utilisé en dehors de la stricte exploitation de `CESAM2k`, il convient donc de leur donner un accès étendu et ainsi de les disposer dans le `HOME`.

Dans le fichier `.bashrc` :

1. Ajouter les chemins de la bibliothèque et des exécutables, par exemple :  

```
PATH=$PATH:~/SCRIPTS:~/SUN_STAR_DATA :~/CESAM2k/EXPLOIT
:~/CESAM2k/SOURCE:~/CESAM2k/NOTICE:~/CESAM2k/TESTS
export PATH
```
2. Déclarer le nom du compilateur Exemple : `FORTRAN=lf95`, et celui du système `OSnom='uname -n'`<sup>7</sup> ; ajouter ces noms dans une liste d'exportables, Exemple : `export FORTRAN OSnom`.
3. Si ce n'a pas été effectué par l'équipe du système, installer le logiciel de dessin `PGPLOT`<sup>8</sup> ; renseigner le chemin de sa bibliothèque : `LD_LIBRARY_PATH=/usr/lib/pgplot5:$LD_LIBRARY_PATH`. On donne au § 1.1.1 (Page 2) un supplément d'informations concernant l'installation de ce logiciel.
4. A défaut, de ce logiciel de dessin, dans le sous-directory `SOURCE` :
  - Supprimer les routines `boite.f`, `box.f`, `des_r.f`, `des_m.f` de la liste des `INCLUDE` du module `mod_numerique.f`.
  - Dans la subroutine `des.f`, ne conserver que `CASE ('no_des')` et, éventuellement, `CASE DEFAULT`.
  - Dans la subroutine `sortie.f`, supprimer `CALL pgend`.

**PB** La liste de ces corrections n'est peut être pas complète.

## 1.4 Adaptation de CESAM2k à l'environnement

Avant de créer les exécutables, il convient d'adapter des paramètres à l'installation. se placer dans le sous-directory `SOURCE` `cd ./CESAM2k/SOURCE`,

- Dans la routine `des_m.f`, vers la ligne 105, adapter le nom du "device" : `device='/xw'`, éventuellement coder `device='?'` de façon à obtenir la liste des devices disponibles.
- Faire de même dans la routine `des_r.f`, vers la ligne 89.
- Dans ces routines `des_m.f` et `des_r.f` il conviendra, par la suite, d'adapter la hauteur et la largeur des cadres, de façon à optimiser leur centrage dans la fenêtre dont on dispose, ou encore de personnaliser le dessin *cf.* § 5.12 (Page 71).

## 1.5 Création des exécutables

Dans les `Makefiles` d'installation `Make`, `Make_exp1`, `Make_exe`, les paramètres de compilation sont définis pour les compilateurs `lf95`, `ifort`, `gfortran`, avec et sans

7. Attention cotes inversées!

8. Avec son mode d'emploi, le logiciel `PGPLOT` est téléchargeable via `anonymous ftp` à l'adresse <http://astro.caltech.edu/~tpj/pgplot/>.

debug. Les instructions de link sont personnalisées suivant différentes machines identifiées par la variable shell exportable *OSnom*. Ces Makefiles sont dérivés d'une version initiale créée par Th. Corbard. Si un compilateur différent est utilisé il convient d'ajouter ses règles de compilation aux règles existantes. Les Makefiles utilisent les "implicit rules" de GNU Make version 4.0, il est donc impératif de respecter les "pattern rules" pour la dénomination des variables. Par exemple, "FFLAGS" doit être le nom de la variable définissant les arguments de compilation<sup>9</sup>.

Les exécutables d'exploitation du sous-directory EXPLOIT sont créés avec l'option 'debug'. Ils utilisent des modules de la SOURCE<sup>10</sup>. Il convient d'abord de créer ces modules en version 'debug', ensuite les exécutables de EXPLOIT enfin, ceux de la SOURCE, avec optimiseur<sup>11</sup>.

1. Se placer dans le sous-directory SOURCE (`cd ./CESAM2k/SOURCE`)

Sous SVN saisir : `svn update`, les mises à jour se mettent en place ;

puis saisir : `make -Bf Make DBG=2k-dbg`. Le Makefile s'exécute, il y a création de la bibliothèque `libcesam2k-dbg.a`, de l'exécutable `cesam2k-dbg.out` et des modules (extension `.mod`) avec "debug".

2. Se placer dans le sous-directory EXPLOIT : (`cd ../EXPLOIT`)

Sous SVN saisir : `svn update`, les mises à jours se mettent en place ;

puis saisir : `make -Bf Make_expl DB=2k`. Le Makefile s'exécute, il y a création des exécutables d'exploitation.

3. Revenir dans le sous-directory SOURCE (`cd ../SOURCE`), saisir : `make -Bf Make DBG=2k`. Le Makefile s'exécute, il y a création de la bibliothèque `libcesam2k.a`, de l'exécutable `cesam2k.out`. Les modules avec optimisation sont conservés dans l'environnement.

## 1.6 Premier test de fonctionnement

*Toutes les données physiques utilisées dans ce test de fonctionnement sont implantées de façon native dans la SOURCE, son exécution ne nécessite pas d'aménagement supplémentaire.*

Se placer dans HOME, saisir "create\_evol\_space TEST0" il y a création du sous-directory TEST0, s'y placer : `cd TEST0`. Les fichiers nécessaires pour l'exploitation de CESAM2k y ont été disposés.

Éditer le fichier `mon_modèle.don`, éventuellement le renommer en conservant l'extension ".don" ; y effectuer les adaptations suivantes :

- Remplacer "des\_m" par "no\_des" si PGPLOT n'est pas opérationnel.
- Adapter "nom\_chemin" au chemin du sous-directory SUN\_STAR\_DATA de l'installation.

---

9. Gnu Make offre la possibilité de redéfinir ces dénominations.

10. Pour limiter le nombre de bibliothèques, les fichiers d'extension `.mod` n'ont pas été regroupés dans une librairie de modules, mais laissés dans SOURCE.

11. Les instructions d'installations suivantes sont rappelées dans le README\_installation de la SOURCE.

- Établir les instructions suivantes (*leurs significations feront l'objet du §4.3 (Page 30), elles sont aussi succinctement rappelées dans le fichier "arguments\_du\_point\_don" qui a été recopié dans le présent directory.*) :

```

&NL_CESAM
NOM_CHEMIN='~/SUN_STAR_DATA/',    <==== à adapter suivant l'installation
NOM_CTES='ctes_94',
NOM_DES='des_m',    <==== à adapter, 'no_des' en l'absence de PGPLOT
NOM_OUTPUT='no_output',
N_MAX=2000,
PRECISION='PR'
/
&NL_MASS
MTOT=1.d0,
NOM_PERTM='no_pertm',
MDOT=0.d0
/
&NL_EVOL
AGEMAX=5000.d0,
ARRET='else',
DTLIST=1.d10,
LOG_TEFF=10.d0,
NB_MAX_MODELES=200,
HE_CORE=-1.d0,
R_STOP=-0.88d0,
T_STOP=5.d7,
X_STOP=-0.1d0
/
&NL_CHIM
GRILLE_FIXE=.FALSE.,
NOM_ABON='solaire_gn',
MODIF_CHIM=.FALSE.,
GARDE_XISH=.FALSE.,
X0=0.70d0,
Y0=0.28d0,
ZSX0=0.d0
/
&NL_CONV
NOM_CONV='conv_jmj',
ALPHA=1.8d0,
OVSHTS=0.d0,
OVSHTI=0.d0,
JPZ=.FALSE.,
CPTURB=0.d0,
LEDOUX=.FALSE.
/
&NL_DIFF

```

```

DIFFUSION=.FALSE.,
NOM_DIFFM='diffm_mp',
NOM_DIFFT='diff_t_nu',
D_TURB=10.d0,
RE_NU=1.d0,
NOM_FRAD='no_frad'
/
&NL_ROT
W_ROT=0.d0,
UNIT='jours',
NOM_DIFFW='diffw_0',
NOM_THW='rot_0',
NOM_PERTW='pertw_0',
P_PERTW=0.d0,
NOM_DES_ROT='no_des'
/
&NL_ETAT
NOM_ETAT='etat_eff',
F_EOS='eos_opal_250.bin',7*' ',
/
&NL_OPA
NOM_OPA='opa_int_zsx',
F_OPA='opa_yveline.bin',7*' ',
NOM_OPA_COND=' '
/
&NL_NUC
NOM_NUC='ppcno9',
NOM_NUC_CPL='NACRE',
MITLER=F
/
&NL_ATM
NOM_ATM='lim_atm',
NOM_TDETAU='hopf',
TAU_MAX=10.d0,
LIM_RO=.TRUE.
/

```

Saisir cesam2k.out

cesam2k.out <==

-----  
 CESAM speaks a bit of english if you include in  
 the working directory a file named langue with the statements :

```

&NL_LANGUE
langue='english'
/

```

cf. aide\_mem2k, chapter Personnalisation

```

-----

Pour arrêter : taper 0 puis RETURN
Pour poursuivre une évolution : taper 1 puis RETURN
Pour initialiser un modèle de ZAMS : taper 2 puis RETURN
Pour initialiser un modèle de PMS : taper 3 puis RETURN
2 <==
  le modèle initial de ZAMS est-il donné en binaire ? o/n
n <==
  entrer le nom du fichier ASCII du modèle initial
  Exemples: m010.zams, m020.zams, m050.zams

m010.zams <==

  CESAM utilise le modèle m010.zams
  entrer l'identificateur du modèle
  Exemple: mon_modele

mon_modele <==

  identificateur des fichiers du modèle : mon_modele

*****
  MODELE DE STRUCTURE INTERNE calculé par CESAM2k version : V3.12.1
*****

Début du calcul le : 20 Mai 2017 à 10h38

  NAMELISTS du fichier: mon_modele.don

  &NL_CESAM NOM_CHEMIN=~ /SUN_STAR_DATA/ .....

  .....
  .....
  .....
  .....
  .....
atmosphère iter. 1 err. max. 6.9E-06, var: Pt, couche 1, corr 1.0E+00
atmosphère iter. 2 err. max. 4.3E-10, var: Pt, couche 73, corr 1.0E+00
----- Restitution de l'atmosphère (fin) -----

  création du fichier ascii: mon_modele-ad.osc

*****
âge= 5.000E+03, LogTeff= 3.761E+00, LogL/Lsol=-2.124E-02, LogR/Rsol=-8.868E-03
Log g= 4.455E+00, Pc= 2.328E+17, Tc= 1.551E+07, Roc= 1.517E+02, Xc= 3.391E-01

```

```
ePP/eNUC= 99%, eCNO/eNUC= 1%, e3a+C+O/eNUC= 0%, eGRAV/eNUC= 0%, Yc= 6.403E-01
Var. rel. de masse : 0.000E+00, M*= 1.000E+00Msol, modèle de série principale
Modèle hors la zone instable / outside the instability strip (P= 4.121E-02day)
*****
```

```
Fin d'évolution avec CESAM2k version : V3.12.1
Sortie car agemax atteint
Fin du calcul le 20 Mai 2017 à 10h40
nom du modèle :
mon_modele
nom du fichier des NAMELISTs des données : mon_modele.don
```

```
-----
-----
-----
```

```
P.Morel, OCA. Août 2006, vers les stades avancés
B.Pichon, P.Morel OCA. Décembre 2008, lois T(tau) de MARCS
Y.Lebreton OBSPM, P.Morel OCA. Décembre 2008, mixtures+opacités AGS05
Y.Lebreton OBSPM, P.Morel OCA. Novembre 2009, etat_irwin
T.Guillot, P.Morel OCA Mai 2012, implantation de sorties *.csv
A.V.Oreshina, V.A.Baturin. P.Morel, Décembre 2016, EOS etat_saha
L.Bigot, C.Renaud, P.Morel, Décembre 2017, opacités Houdek12
*****
Formation ou remplacement du fichier binaire: mon_modele_B.dat
```

Type <RETURN> for next page:

Le test de fonctionnement est terminé. C'est un exemple d'une évolution solaire à partir de la ZAMS. Les résultats obtenus n'ont qu'une valeur de test, il ne convient pas de les utiliser comme "*modèle solaire*". Au § 4.16.2 (Page 50) on donne un exemple d'évolution incluant la PMS.

Lorsque le dessin "on line" est effectué, suivant la définition de l'écran utilisé, il arrive que les cadres soient tronqués et/ou n'utilisent pas la surface disponible de façon satisfaisante, il y a lieu d'en adapter les dimensions, *cf.* § 1.4 (Page 5). Pour ce faire, par tâtonnements adapter les cadres en recherchant les valeurs optimales des données du fichier "device" placé dans le directory local, *cf.* § 5.12 (Page 71). Une fois obtenues, reporter ces valeurs dans le fichier 'device' du directory EXPLOIT afin qu'elles puisse être dorénavant systématiquement prises en compte.

## ORGANIGRAMME GÉNÉRAL

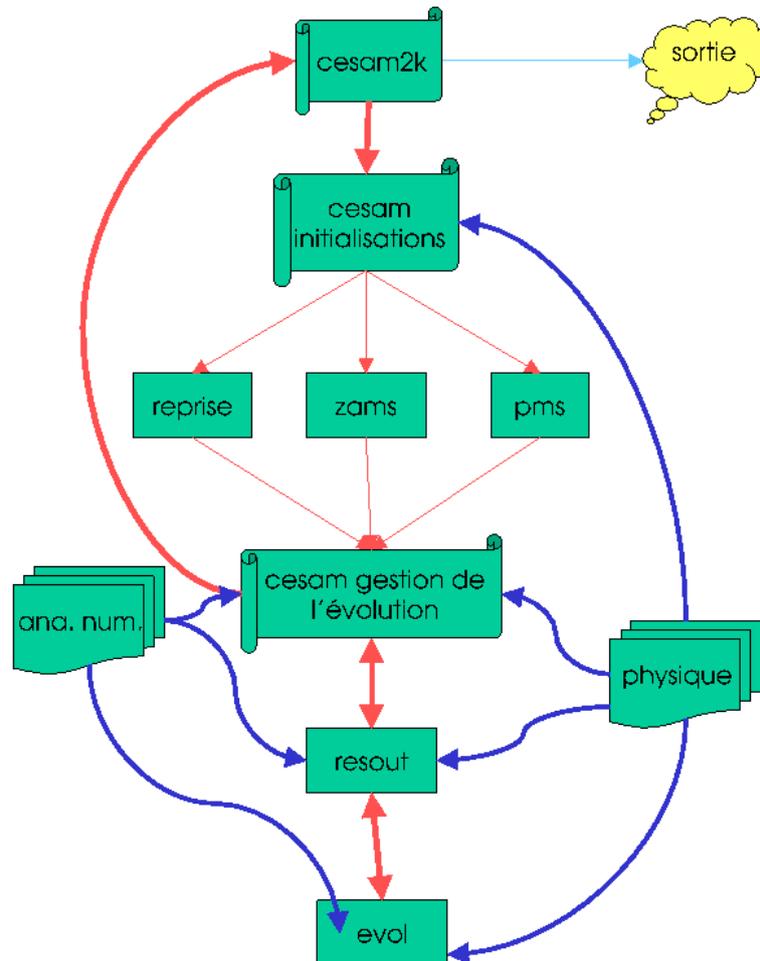


FIGURE 1.1 – Schéma de l’organigramme général de CESAM2*k*. Le programme **cesam2k**, compilé une fois pour toutes, fait appel à la routine **cesam** qui constitue en fait, le programme principal. Dans **cesam**, après des initialisations, on sépare le traitement des 3 possibilités : i) poursuite d’une évolution, ii) initialisation sur la ZAMS homogène, iii) initialisation d’une PMS. Le calcul se poursuit par des opérations de gestion de l’évolution, listing, création de fichiers de sortie, dessin, etc... Est ensuite fait appel à **resout**, où les équations de l’équilibre quasi-statique sont résolues en alternance avec celles de l’évolution du moment cinétique et de la composition chimique. Ces dernières sont résolues séparément dans **evol**. Une fois la convergence obtenue, il y a retour vers les algorithmes de gestion de l’évolution de **cesam** et, de là, poursuite de l’évolution ou retour à **cesam** pour sortie. Les routines d’analyse numérique et de physique sont sollicitées à différents niveaux de calcul.

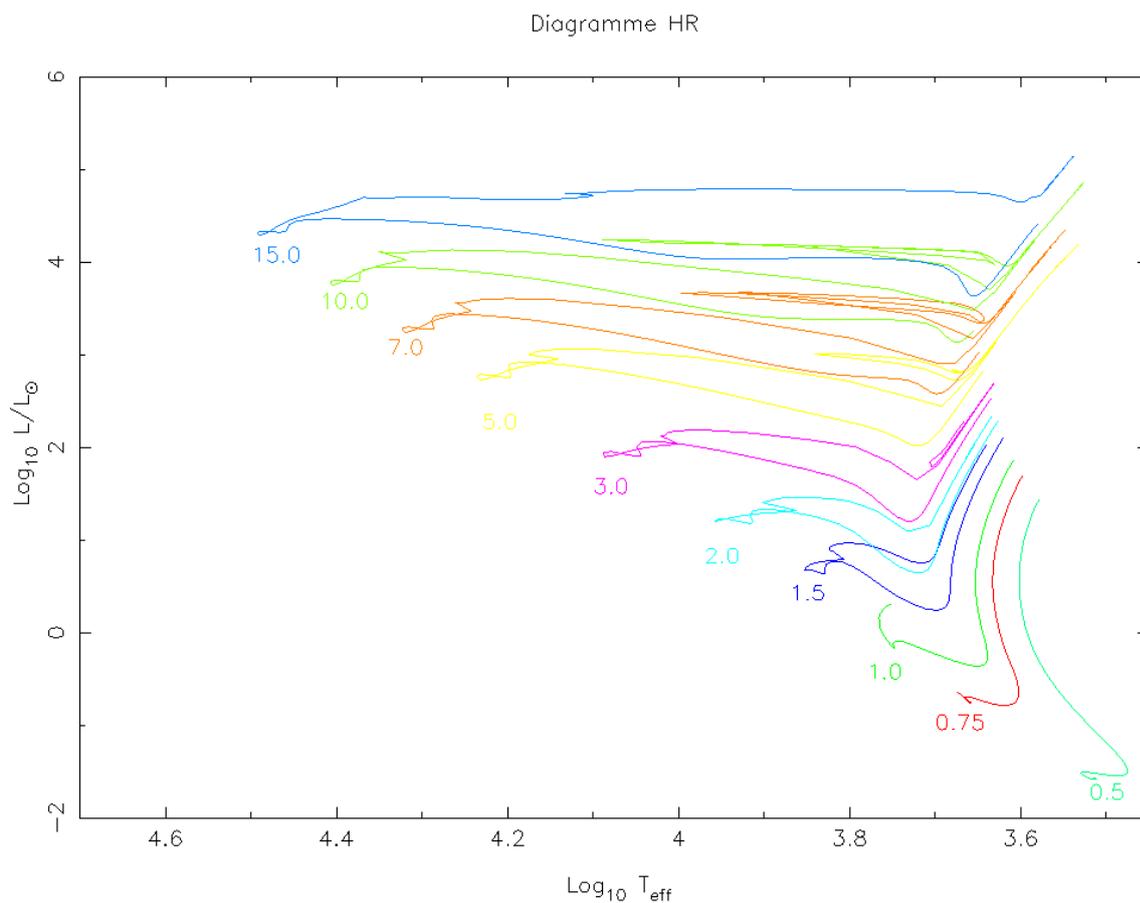


FIGURE 1.2 – Tracés évolutifs incluant la pré-séquence principale pour des étoiles de  $0.5M_{\odot}$  à  $15M_{\odot}$ . Le flash de l'hélium a été dépassé pour les masses supérieures à  $3M_{\odot}$ . Pour les modèles de  $3M_{\odot}$  et de  $5M_{\odot}$  les calculs ont été arrêtés après la disparition du carbone au centre. La phase de destruction de l'oxygène a été atteinte pour les modèles de  $7M_{\odot}$  et de  $10M_{\odot}$ . Avec la disparition de l'oxygène dans le noyau, la limite de validité de l'hypothèse d'équilibre quasi-statique a été atteinte pour le modèle de  $15M_{\odot}$ . Ces tracés ont été obtenus avec le fichier de données du § 4.4 (Page 31).

# Chapitre 2

## SCRIPTS

Le sous directory **SCRIPTS** contient la plupart des shell-scripts qui, initialement, permettaient l'exploitation de *CESAM2k*. Dans la version actuelle, bien que remplacés par des **MAKEFILEs** pour la plupart de leurs fonctions, ils gardent néanmoins leurs utilités. Leur simplicité et leur robustesse permettent parfois de surseoir aux caprices des **MAKEFILEs**. La fonction et l'usage de chaque script sont détaillés dans des commentaires situés dans l'entête.

**change\_noms** : remplacement d'une séquence de caractères dans un **script**. Exemple : `change_noms nom1 nom2 script` (remplace `nom1` par `nom2` partout dans `script`)

**change\_noms\_dir** : remplacement d'une séquence de caractères dans tous les **scripts** d'un directory. Exemple : `change_noms_dir nom1 nom2 mon_dir`

**compile2k** : compilation de programmes fortran avec optimiseur, formation des object-modules. Exemple : `compile2k programme1 programme2 ...`

**compile2k-dbg** : identique au précédent, compilation avec debug

**compile2k-dbg\_list** : script pour compiler avec options de debug tous les programmes de "list" avec insertion dans la bibliothèque "biblio1" de tous les objet-modules. Exemple : `compile2k-dbg_list biblio1 list`. Exemple de fichier list : `ssp1 ssp2 mod1` (en lignes ou colonnes pour les sous-programmes `ssp1.f`, `ssp2.f` et le module `mod1.f`).

**compile2k\_list** : identique au précédent avec optimiseur.

**evol2k\_pms** : évolution avec PMS. Exemple : `evol2k_pms mon_modele`.

**evol2k\_pms\_list** : évolutions avec PMS pour une liste de modèles. Exemple : `evol2k_pms_list list_models`.

**evol2k\_zams** : évolution à partir de la ZAMS homogène. Exemple : `evol2k_zams mon_modele`.

**evol2k\_zams\_list** : évolution à partir de la ZAMS homogène pour une liste de modèles.

**evolsun\_pms** : évolution d'un modèle de  $1M_{\odot}$  avec PMS. Exemple : `evolsun_pms mon_soleil`.

**exe2k** : compilation, link, exécution d'un programme utilisant des bibliothèques de la SOURCE.

**exe2k-dbg** : identique au précédent avec debug.

**genere\_cesam2k** : génération de la source de CESAM2k, équivalent au Makefile Make. Utilisation : **genere\_cesam2k**.

**genere\_cesam2k-dbg** : identique au précédent avec debug.

**gzip\_HR\_list** : compression d'une liste de fichiers HR. Exemple : **gzip\_HR\_list** liste\_hr.

**lib\_del** : suppression d'object-modules d'une bibliothèque bibli.a. Exemple : **lib\_del** bibli.a obj1 obj2 ...

**lib\_repl** : remplacement ou insertion d'object-modules dans une bibliothèque. Exemple : **lib\_repl** obj1 obj2 ...

**mod\_kind.f** : Module définissant les types simple et double précision *sp* et *dp*.

**on\_voit** : transforme le fichier dessin.dvi en dessin.ps puis le dessine avec ghostview. Exemple : **on\_voit** dessin.

**on\_voit\_pdf** : transforme dessin.dvi en dessin.pdf puis le dessine avec ghostview. Exemple : **on\_voit\_pdf** dessin.

**on\_voit\_ps** : transforme dessin.dvi en dessin.ps puis le dessine avec ghostview. Exemple : **on\_voit\_ps** dessin.

**recup\_NOTICE\_trunk** : récupération via SVN du directory NOTICE du trunk.

**recup\_SCRIPTS\_trunk** : récupération via SVN du directory SCRIPTS du trunk.

**recup\_SOURCE\_trunk** : récupération via SVN du directory SOURCE du trunk.

**recup\_SUN\_STAR\_DATA\_trunk** : récupération via SVN du directory du trunk SUN\_STAR\_DATA.

**recup\_TESTS\_trunk** : récupération via SVN du directory TESTS du trunk.

**rempl2k\_mod** : compilation et remplacement d'un module dans la librairie libcesam2k.a. Exemple : **rempl2k\_mod** mod\_cesam.

**rempl2k\_mod-dbg** : identique au précédent avec debug.

# Chapitre 3

## SUN\_STAR\_DATA

Le sous directory `SUN_STAR_DATA` regroupe des données astrophysiques tabulées ainsi que leurs programmes de mise en forme en vue de leur exploitation. L'opacité et l'équation d'état ne sont, le plus souvent, connues que sous la forme de tabulations d'origines diverses qu'il faut assembler au cas par cas. `CESAM2k` utilise ces données physiques sous forme de fichiers binaires dont la structure (*32k*, *64k*, *big/little endianness...*) dépend de l'environnement utilisé. Il y a lieu de les créer au fur et à mesure des besoins.<sup>1</sup> Le directory contient une redondance de données dont l'exploitation n'est pas décrite dans cette notice. Les exécutable (d'extension `.out`) peuvent être créés à l'aide du shell-script '`exe2k`' du sous-directory `SCRIPTS`. Par exemple, pour créer l'exécutable du programme `ascii2bin_opa.f`, saisir :

```
exe2k ascii2bin_opa
```

ou bien encore, à l'aide du Makefile '`Make_exe`' du sous-directory `EXPLOIT`

```
make -Bf Make_exe PRG=ascii2bin_opa
```

**PB** En raison de l'infini diversité des possibilités, il est incontournable de devoir utiliser des tables d'opacité et d'équation d'état calculées pour des compositions chimiques différentes de celles obtenues localement dans le modèle.

Les données sont regroupées par thèmes affectés à des directory distincts qu'il convient de décompresser au fur et à mesure des besoins.

### 3.1 Opacités.

#### 3.1.1 Opacités `opa_yveline.f`, `opa_yveline_lisse.f`

Ces deux routines utilisent les mêmes données regroupées dans le sous-directory `OPA_YVELINE`.

Décompresser le fichier `ascii2bin_opa.f.gz` et, suivant les besoins, l'un des fichiers `opa_yveline.data.gz` ou `opa_ags03.dat.gz` ou `opa_ags05.dat.gz`<sup>2</sup> :

```
gunzip opa_yveline.data.gz ascii2bin_opa.f.gz
```

---

1. Il convient d'actualiser ces données en les reprenant de leurs sites d'origines.

2. Ce fichier est disponible sur simple demande auprès de [Yveline.Lebreton@obspm.fr](mailto:Yveline.Lebreton@obspm.fr)

Comme indiqué ci-dessus créer l'exécutable `ascii2bin_opa.out`. Effectuer la transformation ASCII ==> binaire, en saisissant `ascii2bin_opa.out`. A la question :

```
traduction ASCII ---> binaire (o/n?)
o <==
```

on obtient :

```
ASCII ---> binaire
enter the name of the existing ASCII opacity table: opa_yveline.data
```

répondre :

```
opa_yveline.data <== à adapter!
```

on obtient :

```
enter the name of the binary table: opa_yveline.bin
```

répondre :

```
opa_yveline.bin <== le choix du nom libre, le placer dans l'item F_OPA
```

à la question suivante répondre c, éventuellement recompresser le fichier en ASCII et supprimer l'exécutable. On obtient :

```
PAUSE: OK c'est fait
pour arreter entrer q, poursuivre entrer c
c <==
```

Recompresser les données ASCII : `gzip opa_yveline.data <==` et effacer l'exécutable : `rm ascii2bin_opa.out <==`

On trouve un complément d'information dans le fichier :

```
extract_opa_yveline.explik
```

du sous-directory `SUN_STAR_DATA`.

### 3.1.2 Opacités `opa_houdek12.f`

Dans cette version de `CESAM2k`, l'interpolation des tables d'opacité utilisant le package `opa_houdek9` a été remplacée par sa version actualisée `opa_houdek12`. Cette dernière permet d'exploiter des données d'opacité plus récentes incluant celles utilisées par la version 9.

Le package, de nom `opint_v12.tar.gz` a été repris du site :

```
http://users-phys.au.dk/hg62/opint.html
```

Il a été disposé dans le sous-directory `OPA_H12`.

Après détarage et décompression : `tar -xzvf opint_v12.tar.gz` le module obtenu `gh_int.v12` contient les outils et les données nécessaires à la formation et l'exploitation de tables d'opacité utilisant diverses compositions. Pour des raisons de cohérence il est pratique d'en disposer directement dans le sous-directory `SUN_STAR_DATA` sous le nom de `H12`.

On trouve dans `notes_v12.pdf` les directives pour la compilation et l'exploitation de ce module. Les éléments dans lesquels il est nécessaire d'intervenir sont :

doc --> notes\_v12.pdf, lib, Makefile, OPINPATH

lib contient l'ensemble des routines nécessaires à la construction des tables et à leur exploitation.<sup>3</sup>

Le Makefile permet la construction des tables et des modules exécutables.

Les fichiers OPINPATH contiendront les noms et les chemins des tables. Ce sont les noms des sources d'opacité utilisées pour l'intérieur et l'atmosphère.

En suivant les instructions des notes, éditer le makefile<sup>4</sup>. Adapter le compilateur et l'optimiseur à la machine dont on dispose, lignes 57 à 78.

**Choix des tables d'opacité :** Un choix entre différentes compilations d'opacité est proposé à l'utilisateur. Les tables à utiliser étant définies, le Makefile permet de construire les fichiers et les programmes nécessaires pour effectuer les interpolations. La version originale du makefile permet la construction et l'interpolation des tables d'opacité "GN93" pour l'intérieur et "AG89Fe" pour l'atmosphère (lignes 111 et 128) :

```
INTOPC_DIR    = op05_GN93
ATMOPC_DIR    = rt11_AG89Fe
```

Pour modifier ces choix, il conviendra d'adapter ces 2 lignes ainsi qu'il est décrit plus avant.

**Endianness :** Suivant l'architecture du processeur utilisé, l'organisation des octets en mémoire peut être "big-endian", ou "little-endian". En mémoire, la lecture/écriture des octets s'effectuant dans un sens ou dans l'autre. Il est donc nécessaire qu'il y ait cohérence entre l'endianness des données et celle du processeur. Pour faciliter l'exploitation, le package de G.Houdek utilise par défaut l'endianness du processeur et y adapte celle des données. Sauf, peut être, pour les experts, il est sage d'en laisser l'initiative au package.

Pour une mise en route il est donc conseillé de ne rien changer au makefile et de l'exécuter tel quel en saisissant : **make**. Les exécutables et les fichiers d'interpolation sont créés. Un nouveau fichier OPINPATH est écrit dans l'environnement, il contient les noms des fichiers binaires d'interpolation pour les tables GN93 et AG89Fe désignées aux lignes 111 et 127. Ces fichiers seront utilisés par la routine `opa_houdek12.f` lors des appels à l'opacité. Dans ce nouveau fichier OPINPATH, il est nécessaire d'actualiser les chemins de ces fichiers; pour ce faire, remplacer les `"/.` par le chemin complet du fichier H12 par exemple :

```
./op05_GN93/little_endian/opalxe.bin --->
/home/pierre/SUN_STAR_DATA/H12/op05_GN93/little_endian/opalxe.bin
```

La bibliothèque `libopint.a` est créée dans `/H12/lib`, il conviendra d'en indiquer le chemin dans les instructions de `link`, EXEMPLE :

```
-L /SUN_STAR_DATA/H12/lib -lopint
```

3. Ces routines sont parfois désignées par un nom différent de celui donné au sous-programme lui même. Ainsi la routine de nom `dopint.f` est le sous-programme `opint.f`

4. Lors de la décompression du package `opint.v12.tar.gz`, ou de l'édition du Makefile, il peut arriver que certaines lignes soient mal coupées. Ce à quoi il faudra remédier en fonction des message d'erreur obtenus lors de l'exécution du Makefile.

Pour construire les tables d'interpolation pour d'autres associations de données d'opacité il faut adapter les intructions du Makefile lignes 110-114 pour les tables d'intérieur et 127-131 pour celles d'atmosphère. Il est nécessaire d'utiliser des mixtures identiques. Différentes possibilités sont indiquées dans autres fichiers "OPINTPATH" par exemple :

```
OPINTPATH_op05_GN93+af05_GN93-big
```

Lors de l'exécution d'un MAKEFILE les définitions externes prévalent sur les définitions internes. Ainsi qu'il est indiqué dans le mode d'emploi cf. `notes_v12.pdf` il est donc possible de préciser ces tables lors de la saisie de l'appel à la construction du Makefile. Le fichier OPINPATH créé mentionnera alors ces tables.

Dans le fichier de données, il suffira d'identifier les tables interpolées par 'OPINT-PATH', Exemple : `F_OPA='opa_yveline.bin', 'H12/OPINTPATH', 6*` '

## 3.2 Equations d'état.

### 3.2.1 Equation d'état `etat_opal.f`

La source (Iglesias & Rogers, 1991) et les tables de l'équation d'état ont été reprises du site WEB `ftp://www-phys.llnl.gov/pub/opal/eos/`, où l'on trouve un "README" expliquant la marche à suivre pour construire une table ASCII d'équation d'état pour une valeur de  $Z$  fixée. On transforme ensuite cette table ASCII en fichier binaire.

**PB** Certains des sous-programmes du package original d'OPAL ont été modifiés pour l'exploitation avec CESAM2k; en particulier, le `blockdata` a été supprimé.

Les tables de données et les routines d'exploitation se trouvent dans le sous-directory `ETAT_OPAL1`. Le décompresser.

Les fichiers ASCII compressés `peos*.gz` contiennent les données pour l'équation d'état OPAL. Se placer dans le sous-directory `SUN_STAR_DATA`, décompresser tous ces fichiers : `gunzip peos*.gz`.

Exécuter le programme `ZFSinterppeos` après en avoir décompressé la source : `ZFSinterppeos.f.gz`

```
exe2k ZFSinterppeos <==
Encountered 0 errors, 0 warnings in file ZFSinterppeos.f.
type Z:
0.02 <==
```

entrer la valeur de  $Z$ , fraction de masse de l'abondance des métaux; on obtient :

	0.020		
0.0000	0.0000	3.0000000	4.0036974
0.0000	0.0200	3.0301077	4.0629535
0.0000	0.0400	3.0611377	4.1251554
0.2000	0.0000	2.5018048	2.5114787
0.2000	0.0200	2.5158978	2.5350964
0.2000	0.0400	2.5302570	2.5591605

0.4000	0.0000	2.2741616	1.8296305
0.4000	0.0200	2.2828465	1.8421309
0.4000	0.0400	2.2916493	1.8548033
0.6000	0.0000	2.1437435	1.4389958
0.6000	0.0200	2.1498623	1.4467140
0.6000	0.0400	2.1560483	1.4545176
0.8000	0.0000	2.0592246	1.1858406
0.8000	0.0200	2.0638888	1.1910768
0.8000	0.0400	2.0685947	1.1963595

Le fichier ASCII EOSdata occupant environ 9Mo a été créé.

Editer la source du programme `opal_ascii_bin`, et donner au fichier binaire le nom désiré, par exemple :

```

.....

        CLOSE(unit=60)    <===   ligne 117 de opal_ascii_bin.f

c   f_eos='eos_opal_190.bin'
    f_eos='eos_opal_195.bin'
c   f_eos='eos_opal_330.bin'
c   f_eos='eos_opal_180.bin'
c   f_eos='eos_opal_170.bin'
c   f_eos='eos_opal_130.bin'
c   f_eos='eos_opal_050.bin'

        WRITE(*,6)f_eos
6   FORMAT('debut ecriture sur le fichier binaire: ',a50)
        CLOSE(unit=60)

.....

```

Exécuter le programme `opal_ascii_bin` ainsi modifié :

```

exe2k opal_ascii_bin <==
Encountered 0 errors, 0 warnings in file opal_ASCII_bin.f.
donnees prises dans le fichier: EOSdata
lecture, et c'est long, de ce fichier
fin de lecture des tables OPAL
debut ecriture sur le fichier binaire: eos_opal_195.bin
fin ecriture sur le fichier binaire
donnees prises dans le fichier binaire: eos_opal_195.bin
lecture, des donnees EOS opal, et c'est long
fin de lecture des donnees EOS opal en binaire
test relecture effectue

```

Supprimer les modules exécutables :

```
rm *.out <==
rm: remove regular file 'ZFSinterppeos.out'? y <==
rm: remove regular file 'opal_ascii_bin.out'? y <==
```

Enfin, recompresser les fichiers ASCII, et supprimer les fichiers EOSdata et fort.2 désormais inutiles :

```
gzip peos* <==
rm EOSdata <==
rm: remove regular file 'EOSdata'? y <==
rm fort.2 <==
rm: remove regular file 'fort.2'? y <==
```

### 3.2.2 Equations d'état `etat_opalX.f`, `etat_opalZ.f`

Les équations d'état opal 2001 offrent une alternative pour la métallicité : `etat_opalZ`, comme dans la version 1991, crée une table pour une métallicité donnée, et l'abondance d'hydrogène doit être dans l'intervalle  $X \in [0, 0.8]$ , tandis que `etat_opalX`, crée une table pour  $X \in [0, 1]$  donné mais avec  $Z = 0$ , *cf.* §4.8 (Page 40).

Les données et les programme de construction des tables se trouvent dans le sous-directory `ETAT_OPAL_XZ` qu'il convient de décompresser et où on se placera.

- Avec  $Z \neq 0$ , créer le fichier d'équation d'état correspondant à la métallicité requise en décompressant puis exécutant le programme `Z_interp_IEOS` : saisir `exe2k Z_interp_IEOS`, on obtient :

```
Encountered 0 errors, 0 warnings in file Z_interp_IEOS.f.
type Z:  0.025  <===== entrer la valeur de Z
          0.025000000
```

Il y a création d'un fichier ASCII compressé de nom `EOSdata.gz`. Transformer ce fichier en binaire, en lui donnant le nom avec lequel il sera identifié par la suite. Pour ce faire, entrer ce nom dans le programme `opalZ_ascii_bin` (vers la ligne 135) :

```
c f_eos='eos_opal_190.bin'
c f_eos='eos_opal_195.bin'
f_eos='eos_opal_250.bin' <=====
c f_eos='eos_opal_330.bin'
c f_eos='eos_opal_180.bin'
```

puis exécuter ce programme qui comporte un test de lecture :

```
saisir exe2k opalZ_ascii_bin, on obtient :
Encountered 0 errors, 0 warnings in file opalZ_ascii_bin.f.
Fichier de données inconnu : EOSdata
décompression du fichier ASCII de données : EOSdata.gz
données ASCII prises dans le fichier: EOSdata
lecture, et c'est long, de ce fichier
fin de lecture des tables OPAL
début des écritures sur le fichier binaire: eos_opal_250.bin
fin des écritures en binaire, test de relecture
données prises dans le fichier binaire: eos_opal_250.bin
```

```

décompression du fichier
lecture, des données EOS opal, et c'est long
recompression du fichier binaire eos_opal_250.bin
fin de lecture des données EOS opal en binaire
test de relecture réussi

```

Eventuellement supprimer EOSdata.gz :

```
rm EOSdata.gz
```

- Avec  $Z = 0$ , l'interpolation utilise le fichier EOSdata\_H-He qu'il suffit de transformer en binaire en exécutant le programme opalX\_ascii\_bin du sous-directory SUN\_STAR\_DATA.

### 3.2.3 Equation d'état etat\_opal5.f

Equation d'état similaire aux précédentes se trouvant dans le sous-directory ETAT\_OPAL5. Le README original donne quelques explications pour l'utilisation. La table est créée par les deux programmes ZFS\_interp\_EOS5 et opal5Z\_ascii\_bin dont on formera les exécutables. Le script create\_EOS5 lance la formation de la table en binaire dont le nom op5\_iiii.bin utilise la valeur iiii déduite de  $Z$  (ici  $Z = 0.0174$ ) qui figurera dans l'appel à la procédure : create\_EOS5 0.0174. Placer ensuite la table compressée de nom op5\_0174.bin.gz dans le directory SUN\_STAR\_DATA. Dans le fichier de données il ne sera pas nécessaire d'indiquer que la table est compressée. L'EOS opal5 a été communiquée et poue l'essentiel, adaptée à CESAM2k par V.A. Baturin & A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State.

### 3.2.4 Equation d'état etat\_mhd.f

Package de l'équation d'état MHD (Mihalas et al., 1988) appelé par la routine etat\_mhd (*vérifier que ce package, fourni par W. Däppen, est actualisé*). Les 8 tables mhd1\_oc\_tau.tab ... mhd8\_oc\_tau.tab qui se trouvent sur le serveur anonyme usc.edu, dans le répertoire pub/astro-physics/mhd-oc-tau/wd-evo, sont disposées dans le sous-directory ETAT\_MHD sous les noms mhd1.tab.gz ... mhd8.tab.gz. Décompresser ETAT\_MHD, et s'y placer. Décompresser les tables en ASCII : gzip mhd\* <== et la source du programme fmttob : gzip fmttob.f.gz. Transformer les tables en binaire en exécutant fmttob <==. Le README rappelle cette procédure.

```
exe2k fmttob
```

```
Encountered 0 errors, 0 warnings in file fmttob.f.
```

```
decompression
```

```
Press Enter to Continue. <==
```

```
enter main
```

```
ZAMS-type: filename of input (formatted) table # 1
```

```
enter filename of output (unformatted) table # 1
```

```
ZAMS-type: filename of input (formatted) table # 2
```

```
enter filename of output (unformatted) table # 2
```

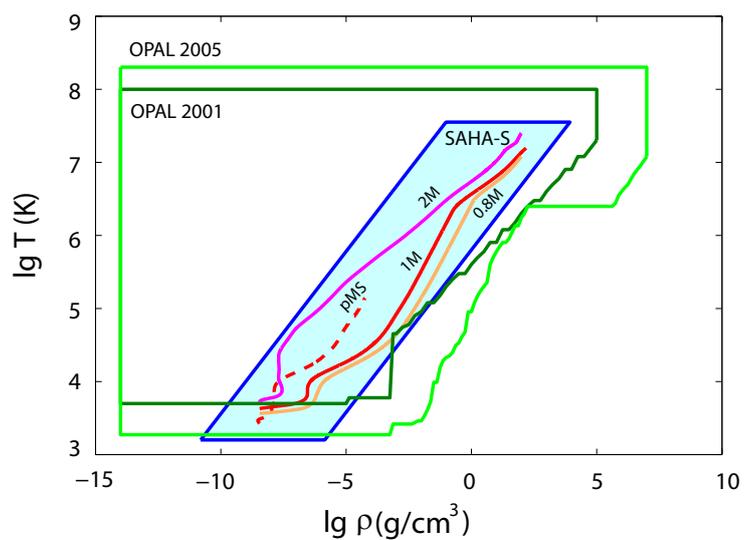


FIGURE 3.1 – Domaines des EOS OPAL2001, OPAL2005 et SAHA-S. Graphe communiqué par A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State University.

```
ZAMS-type: filename of input (formatted) table # 3
enter filename of output (unformatted) table # 3
center-type: filename of input (formatted) table # 4
```

.....

4.002600	7.7222100E-02	0.2299999
12.01100	4.2665000E-04	3.8132454E-03
14.00670	1.0717000E-04	1.1169993E-03
15.99940	9.1215000E-04	1.0859599E-02
20.17900	2.8039000E-04	4.2102317E-03

```
mean molecular weight = 1.2455340
```

```
bilou@port-morel%
```

Supprimer les exécutables :

```
rm *.out <==
rm: remove regular file fmttob.out ? y <==
```

Placer les tables binaires obtenues dans /SUN\_STAR\_DATA.

### 3.2.5 Equation d'état analytique `etat_irwin.f`

Cette équation d'état "analytique" est plus sophistiquée que `etat_eff` ou `etat_ceff` au sens où elle permet de tenir compte séparément de chacun des éléments lourds de la mixture et non pas globalement en tant que "Z" et ce au prix d'un effort de calcul beaucoup plus considérable. Suivant le système d'exploitation utilisé il est possible de l'implanter de différentes façons. Le mode d'emploi qui suit est essentiellement dû à Y.Lebreton.

Un préalable est de vérifier si les bibliothèques d'algèbre linéaire `liblapack.a` et `libblas.a` sont déjà implantées, souvent dans le directory `/usr/lib`. Si tel n'est pas le cas les implanter à l'aide d'`apt-get` ou des paquets `synaptic`, cf. §3.4.1 (Page 25).

**Pour implémenter `free_eos`**, la procédure la plus directe consiste à se placer dans le sous-directory `ETAT_irwin` et d'y suivre les instructions du `README_installation` :

1. Se placer dans `free_eos2.2.1/src`.
2. Saisir : `../scripts/compile2k-irw_list lib_eos list_compil`.
3. Copier `libeos.a` : `cp libeos.a $HOME/SUN_STAR_DATA/libeos.a`.
4. Se placer dans `utils` : `cd ../utils`.

5. Saisir : `../scripts/compile2k-irw_list libutils list_compil.`
6. Copier `libutils.a` : `cp libeos.a $HOME/SUN_STAR_DATA/libutils.a.`

En cas d'échec il conviendra de suivre les instructions des auteurs :

1. Créer cette bibliothèque en suivant la procédure des MAKEFILES décrite dans le README du fichier `free_eos-2.2.1.`
2. Créer la bibliothèque avec le compilateur à disposition :
  - Se placer dans le sous-directory `scripts`. Dans le script (bash) `compile`, adapter les paramètres de compilation au compilateur utilisé.
  - Se placer dans le sous-directory `cd ../src`, exécuter :  
`../scripts/compile_list libeos list_compil` et copier la bibliothèque `libeos.a` dans le directory `SUN_STAR_DATA` :  
`cp libeos.a ~/SUN_STAR_DATA/libeos.a`
  - Se placer dans le sous-directory `cd ../utils`, exécuter :  
`../scripts/compile_list libutils list_compil` et copier la bibliothèque `libutils.a` dans le directory `SUN_STAR_DATA` :  
`cp libutils.a ~/SUN_STAR_DATA/libutils.a.`

3. Un type d'implantation adapté à `CESAM2k` est décrit dans le fichier `README_YL_EOS_IRW`. Le fichier `README` donne des informations générales concernant l'implantation et l'exploitation de `free_eos`. Dans le script `exe2k`, cf. §4.17.3 (Page 55), et/ou dans les Makefiles indiquer le chemin des bibliothèques `liblapack`, `libblas`, `libeos` :

```
-L ~/SUN_STAR_DATA -leos \  
-lm -llapack -lblas
```

4. Il est évidemment possible de suivre et d'exécuter les instructions développées dans les différents README de cette source.

Une possibilité de personnalisation des paramètres de cette équation d'état est décrite au §5.7 (Page 64) et, pour son exploitation, §4.12 (Page 45).

### 3.2.6 Equation d'état `etat_saha.f`

Equation d'état tabulée à  $Z$  variable. L'EOS est calculée à partir des tables de Gryaznov, Iosilevsky & Starostin, communiquées et adaptées à `CESAM2k` par V.A. Baturin & A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State University. On en trouvera les données et les explications originales dans le sous-directory `ETAT_SAHA` dans lequel les 8 tables ASCII `eos_saha...dat.gz` compressées doivent être placées. Ne **pas** indiquer leur nom dans le fichier de données. La métallicité initiale doit être dans l'intervalle  $[0.00 < Z < 0.020]$ . Au premier appel les tables entourant la métallicité initiale sont reprises des données. La dépendance en  $Z$  est obtenue par des interpolations linéaires. Les dérivées des grandeurs thermodynamiques sont déduites des interpolations des tables par B-splines en utilisant des routines reprises de (De Boor, 1978).

## 3.3 Lois $T(\tau)$

### 3.3.1 Lois $T(\tau)$ MARCS

Les lois  $T(\tau)$  de MARCS, *cf.* § 4.15 (Page 48), sont disposées dans le sous-directory ATM\_MARCS.

### 3.3.2 Lois $T(\tau)$ PIAU

Les lois  $T(\tau)$  de L.Piau, *cf.* § 4.15 (Page 48), sont disposées dans le sous-directory ATM\_PIAU.

### 3.3.3 Lois $T(\tau)$ Roger

Les lois  $T(\tau)$  non purement radiatives de R.Cayrel, *cf.* § 4.15 (Page 48), sont disposées dans le sous-directory ATM\_ROGER. Ces lois sont dérivées des modèles d'atmosphère de Kurucz pour différentes métallicités. Les fonctions  $q(\tau)$  sont extraites des données originales et mises sous la forme

$$T = T_{\text{eff}}(3/4(\tau + q(\tau)))^{1/4}$$

puis interpolées en fonction de la profondeur optique  $\tau$  Rosseland.

## 3.4 Divers

### 3.4.1 Package Lapack

Le package d'analyse numérique linéaire **Lapack** est souvent introduit de façon native dans les installations si tel n'est pas le cas, il est nécessaire de construire les bibliothèques et les exécutables. A cette la version 3.7.1 est disposée dans le sous-directory LAPACK-3.7.1. Un préalable à son installation est de vérifier si les bibliothèques d'algèbre linéaire **lapack** et **blas** ne sont pas déjà implantées. Sinon, sous LINUX, une première possibilité est les installer avec :

```
apt-get install lapack3-dev blas3-dev
```

Il y a échec si la bibliothèque **apt-get** n'est pas compatible. Dans cas tenter d'utiliser le *gestionnaire de paquets synaptic*, pour l'installer saisir :

```
sudo apt-get install synaptic
```

l'activer à partir des "favoris" regroupées dans le "système".

En cas d'échec, suivre la procédure d'installation contenue dans le README du package LAPACK :

- Rechercher dans le sous-directory INSTALL s'il existe un Makefile de nom `make.inc.cmpl` correspondant au compilateur 'cmpl' utilisé; Exemple pour le compilateur ifort : `make.inc.ifort`.
- S'il existe, le recopier dans le directory LAPACK sous le nom de `make.inc`.
- S'il n'existe pas, dans le directory LAPACK, copier le makefile `make.inc.example` dans `make.inc` et adapter ce dernier au compilateur utilisé.

- Saisir "make" pour exécuter le Makefile créant les bibliothèques et les exécutable.

### 3.4.2 Modèles ASCII d'initialisation PMS et ZAMS homogènes

Des modèles ASCII d'initialisation PMS et ZAMS homogènes sont disposés dans le sous-directory `HOM_PMS`.

Trois modèles ASCII d'initialisation PMS :

1. `2d-2.pms` : modèle initial de PMS calculé avec une valeur de la constante de contraction égale à  $2 \cdot 10^{-2}$ . La température centrale initiale étant de l'ordre de  $2 \cdot 10^5$  K cette initialisation est à utiliser si on recherche des effets liés aux molécules.
2. `5d-4.pms` : modèle initial de PMS calculé avec une valeur de la constante de contraction égale à  $5 \cdot 10^{-4}$ . La température centrale initiale étant de l'ordre de  $5 \cdot 10^5$  K cette initialisation convient pour la plupart des situations.
3. `8d-5.pms` : modèle initial de PMS calculé avec une valeur de la constante de contraction égale à  $8 \cdot 10^{-5}$ . La température centrale initiale étant de l'ordre de quelques  $10^6$  K, à réserver pour les masse élevées.

Les modèles de ZAMS homogène<sup>5</sup> n'ont pas de sens physique au sens où les réactions nucléaires ne sont pas en équilibre avec les abondances des éléments chimiques. Ils se différencient des modèles de ZAMS caractérisés par l'équilibre entre la production d'énergie radiative et d'énergie gravitationnelle. Les incohérences physiques des modèles initialisés sur la ZAMS homogène sont relaxées à l'issue d'une vingtaine de pas temporels environ et l'évolution ultérieure coïncide avec celle d'un modèle de mêmes caractéristiques initialisé sur la PMS. Une réserve toutefois à cette assertion concerne les modèles comportant des éléments dont l'abondance *cosmique initiale* conduit à des réactions nucléaires explosives dues aux conditions physiques centrales. Le lithium et le deutérium étant particulièrement concernés.

Trois modèles ASCII d'initialisation ZAMS :

1. `m010.zams` : modèle initial de ZAMS homogène d'une masse  $1M_{\odot}$ , convient pour l'initialisation de modèles entre  $0.5M_{\odot}$  et  $1.5M_{\odot}$ . En deça, calculer des modèles initiaux de masse inférieure en procédant par étapes successives avec des masses décroissantes.
2. `m020.zams` : modèle initial de ZAMS homogène d'une masse  $2M_{\odot}$ , convient pour l'initialisation de modèles entre  $1.5M_{\odot}$  et  $3.5M_{\odot}$ .
3. `m050.zams` : modèle initial de ZAMS homogène d'une masse  $5M_{\odot}$ , convient pour l'initialisation de modèles entre  $3.5M_{\odot}$  et  $10M_{\odot}$ . Au delà, calculer des modèles initiaux de masse supérieure en procédant par étapes successives avec des masses croissantes.

---

5. Ces modèles de noms génériques `mon_modele.B.hom` diffèrent des modèles de ZAMS issus d'une évolution pré-séquence principale, noms génériques `mon_modele.B.zams`.

### 3.4.3 Données pour accélérations radiatives

Ces données permettant le calcul des accélérations radiatives selon le formalisme de G.Alécian, *cf.* § 4.10 (Page 43). Elles sont regroupées dans les fichiers du sous-directory `ACC_RAD`. Les accélérations radiatives n'ont d'efficacité que pour des étoiles de types spectral au delà de **F**.



# Chapitre 4

## Exploitation

*Va despacio, estoy de prisa.*  
Proverbe Mexicain.

### 4.1 Conseils pour l'exploitation

Un code numérique, aussi bien fait soit-il, ne peut être "clefs en mains". Pour entamer une exploitation, on doit toujours être guidé par la remarque triviale :

*Un programme qui ne fonctionne pas, ou mal, pour des cas simples ne fonctionnera pas, ou encore plus mal, pour des cas compliqués.*

Un code de structure interne n'échappe, hélas pas, à cette règle. A cet effet, bien que CESAM2*k* ait prévu différents types de "précision" standards, il est presque toujours nécessaire, à l'aide de "réglages", d'optimiser les paramètres numériques en fonction des conditions rencontrées au cours du calcul. Aborder un problème en introduisant *ad initio* toute la complexité que l'on désire atteindre est une perte de temps. Afin d'acquérir une expérience sur la façon dont l'évolution se déroule, il faudra très rapidement reprendre des cas simples. Il est conseillé de "dégrossir" le problème en utilisant des options numériques et physiques robustes et efficaces, afin d'apprécier rapidement "ce qui se passe", puis d'introduire progressivement la complexité souhaitée.

**PB** Les descriptions des routines données plus avant ne sont que succinctes, se reporter aux commentaires figurants au début des routines pour plus d'informations.

### 4.2 Création d'un espace de travail

Pour un ensemble d'applications, il est pratique de créer un espace de travail spécifique. CESAM2*k* offre la possibilité de simplifier l'initialisation de l'environnement du calcul. Par exemple, pour calculer un ensemble de modèles et effectuer des exploitations, de w-virginis, on créera et on se placera dans un directory nommé W\_VIR :

1. Saisir : `create_evol_space W_VIR`. Le directory `W_VIR` est créé. Le `README_usage_de_evol_space` décrit l'utilisation des exécutables qui y ont été introduits. Les données générales (modèles initiaux etc..) y sont recopiées de `CESAM2k/EXPLOIT`. Se placer dans `W_VIR` qui devient l'environnement des calculs.
2. Renommer le fichier de données `mon_modele.don` qu'on y trouvera : `cp mon_modele.don w_vir.don` et adapter les paramètres. A cette fin, on trouvera dans le fichier `arguments_du_point.don` les arguments reconnus par la présente version `CESAM2k`. Ils seront détaillés ci-après.
3. Pour lancer un calcul saisir : `cesam2k.out`

Suivant la définition du moniteur, le graphique peut être tronqué ou mal positionné. Il convient alors adapter la hauteur et la largeur des cadres en bricolant dans le fichier `device`, cf. §5.12 (Page 71). Une fois des réglages satisfaisants obtenus, les reproduire dans le fichier `device` du directory `CESAM2k/EXPLOIT` afin de les récupérer à chaque fois qu'un nouvel espace de travail sera créé par `create_evol_space`.

Le fichier `mon_modele.zoom` permet d'adapter, si besoin, les coordonnées du dessin *on line* du diagramme HR, cf. §5.11 (Page 69).

### 4.3 Le fichier de données : `mon_modele.don`

Les options physiques à utiliser et les données sont transmises au programme principal par des `NAMELIST` regroupées dans un fichier en ASCII dont le nom a **nécessairement** l'extension `.don`, par exemple : `mon_modele.don`. On a disposé un exemple de ce fichier dans le sous-directory `EXPLOIT`.

- `& NL_CESAM` : regroupe les conditions générales du calcul.
- `& NL_MASS` : regroupe les paramètres physiques relatifs à la variable spatiale.
- `& NL_EVOL` : regroupe les paramètres physiques relatifs à la variable temporelle.
- `& NL_CHIM` : regroupe les paramètres physiques relatifs à la composition chimique.
- `& NL_CONV` : regroupe les paramètres physiques relatifs à la convection.
- `& NL_DIFF` : regroupe les paramètres physiques relatifs à la diffusion.
- `& NL_ROT` : regroupe les paramètres physiques relatifs à la rotation.
- `& NL_ETAT` : regroupe les noms des fichiers, *d'accès externe*, éventuellement nécessaires au calcul de l'équation d'état.
- `& NL_OPA` : regroupe les noms des fichiers, *d'accès externe*, éventuellement nécessaires au calcul de l'opacité.
- `& NL_NUC` : définit des options relatives au calcul des taux de réactions thermonucléaires.
- `& NL_ATM` : regroupe les paramètres physiques relatifs à la restitution de l'atmosphère.

## 4.4 Exemple de fichier de données

Cet exemple est volontairement différent de celui utilisé au §1.6 (Page 6).

Les types, real, integer... des paramètres numériques et logiques est facilement identifiable par les valeurs données ci dessous à titre indicatif.

```

&NL_CESAM
NOM_CHEMIN='~/CESAM2k/SUN_STAR_DATA/',
NOM_CTES='ctes_94',
NOM_DES='des_m',
NOM_OUTPUT='no_outputR',
N_MAX=6000,
PRECISION='AV'
/
&NL_MASS
MTOT=2.d0,
NOM_PERTM='pertm_ext$',
MDOT=0.d0
/
&NL_EVOL
AGEMAX=12.d3,
ARRET='else',
DTLIST=1.d10,
LOG_TEFF=10.d0,
NB_MAX_MODELES=40,
HE_CORE=-0.1d0,
R_STOP=3.0d0,
T_STOP=2.0d9,
X_STOP=-0.1d0
/
&NL_CHIM
GRILLE_FIXE=.FALSE.,
NOM_ABON='solaire_gs',
MODIF_CHIM=.FALSE.,
GARDE_XISH=.FALSE.,
X0=0.70,
Y0=0.28,
ZSX0=0.0d0
/
&NL_CONV
NOM_CONV='conv_jmj',
ALPHA=1.8d0,
OVSHTS=0.d0,
OVSHTI=0.d0,
JPZ=F,
CPTURB=0.d0,
LEDOUX=F
/

```

```

&NL_DIFF
DIFFUSION=.TRUE.,
NOM_DIFFM='diffm_mp',
NOM_DIFFT='diff_t_nu',
D_TURB=10.d0,
RE_NU=1.d0,
NOM_FRAD='no_frad'
/
&NL_ROT
W_ROT=0.d0,
UNIT='kms/s',
NOM_DIFFW='diffw_0',
NOM_THW='rot_0',
NOM_PERTW='pertw_0',
NOM_DES_ROT='end_mod'
/
&NL_ETAT
NOM_ETAT='etat_eff',
F_EOS='eos_opal_250.bin',7*' ',
/
&NL_OPA
NOM_OPA='opa_houdek12',
F_OPA='opa_yveline.bin','H12/OPINTPATH',6*' ',
NOM_OPA_COND='opa_cond_mestel'
/
&NL_NUC
NOM_NUC='ppcno3aco',
NOM_NUC_CPL='NACRE',
MITLER=F
/
&NL_ATM
NOM_ATM='lim_atm',
NOM_TDETAU='hopf',
TAU_MAX=30.d0,
LIM_RO=.TRUE.
/

```

Certains arguments des NAMELISTs acceptent des "SUFFIXES"<sup>1</sup> permettant différentes actions "secondaires", Exemple : avec le suffixe **M** de l'item **PRECISION** de la NAMELIST **NL\_CESAM**, le calcul sera effectué avec le nombre de couches fixé à la valeur maximale **N\_MAX** indiquée dans le fichier de données. Exemple : **PRECISION='AVM'**

S'il y a reprise ou poursuite d'une évolution, **CESAM2k** accepte que des suffixes apparaissent/disparaissent des NAMELISTs, leurs actions seront alors prises en compte/ignorées dans l'évolution ultérieure.

---

1. Toujours en "upper case".

Ces SUFFIXES sont indiqués pour mémoire dans le fichier `arguments_du_point_don` et leurs actions détaillées dans les paragraphes suivants,

**PB** Les noms permettant des affectations sont *case sensitive*, par exemple, il faut coder "NACRE" et non "nacre"; il en est de même pour les chemins.

## 4.5 Arguments de la NAMELIST NL\_CESAM

**PB** Les arguments utilisables dans les NAMELISTs du fichier de données sont indiqués dans le fichier `argument_du_point_don` des sous-directory SOURCE et EXPLOIT.

- `NOM_CHEMIN='~/SUN_STAR_DATA/'` : chemin et nom du répertoire où sont disposées les données physiques, les tables d'opacité par exemple. Il est parfois nécessaire de coder le chemin complet, Ex : `'/home/morel/CESAM2k/SUN_STAR_DATA/'`. Parfois `'$HOME/SUN_STAR_DATA/'` n'est pas accepté.
- `NOM_CTES` : indique le nom de la routine d'initialisation des principales constantes de physique à utiliser<sup>2</sup> :
  - `ctes_31` : constantes du LATT, Castro et al. (2007).
  - `ctes_85` : constantes de GONG, Christensen-Dalsgaard (1988).
  - `ctes_94` : constantes de CESAM2k (défaut).
  - `ctes_94m` : identique à `ctes_94` avec des valeurs des masses des noyaux posées égales aux valeurs entières les plus proches.
  - `ctes_ba` : constantes de Basu & Antia (2008).
  - `ctes_gaia` : constantes de GAIA.
- `NOM_DES` : nom de la routine de dessin à utiliser :
  - `des_m/des_r` : dessin en fonction de la masse/rayon, utilise PGPLOT. Dans le cadre des abondances, les valeurs indiquées sont normalisées par rapport à leur maximum sauf pour l'hydrogène et l'hélium indiquées en valeurs réelles.
  - `zoom` : forme personnalisable de `des_m` décrite au § 5.11 (Page 69), pratique lorsqu'on désire atteindre des conditions particulières, par exemple une localisation dans le diagramme HR.

**Suffixes C, HHe, S :**

**C** : `des_mC/des_rC` permet d'indiquer, dans le cadre des abondances, les valeurs centrales en lieu et place des maximales (utile pour les stades avancés).

**S** : `des_mS/des_rS` permet d'indiquer les valeurs de surface des abondances en lieu et place des maximales (utile avec diffusion).

**HHe** : CESAM2k formera un fichier, de nom `mon_modele.mHHe`, contenant les limites des shells source H et He, cf. § 5.10 (Page 68). Exemples : `des_mHHe/des_rHHe`.

---

2. Dans chaque cas, sont seules implémentées les différences connues avec les constantes par défaut de CESAM2k, *i.e.* `ctes_94`.

- `no_des` : sans dessin *on line*, à utiliser nécessairement si on ne dispose pas du logiciel PGPLOT, voir § 1.3 (Page 4).  
**Suffixes 0, 1, 2** : `no_des0` : sans dessin *on line*. `no_des1` : graphes des variables quasi-statiques à chaque itération Newton-Raphson. `no_des2` : graphes des variables quasi-statiques à l'issue de chaque pas temporel.
- `NOM_OUTPUT` : type de fichier ASCII à générer à l'issue du calcul :
  - `osc_adia` : génération du fichier ASCII de nom `mon_modele-ad.osc` pour le calcul des oscillations adiabatiques et la réalisation de certains dessins,
  - `all_adia` : génération des fichiers ASCII, pour le calcul des oscillations adiabatiques et des dessins, pour **TOUS** les modèles. Ces fichiers ont pour dénomination `nnnn-mon_modele-ad.osc`, `nnnn` étant le numéro du modèle,
  - `osc_invers` : génération des fichier ASCII de nom `mon_modele-inv.osc` pour les inversions,
  - `all_invers` : génération de **TOUS** les fichiers ASCII de noms `nnnn-mon_modele-inv.osc` pour les inversions, `nnnn` étant le numéro du modèle,
  - `osc_nadia` : génération du fichier ASCII de nom `mon_modele-nad.osc` pour le calcul des oscillations non adiabatiques,
  - `all_nadia` : génération de **TOUS** les fichiers ASCII de noms `nnnn-mon_modele-nad.osc` pour le calcul des oscillations non adiabatiques, `nnnn` étant le numéro du modèle,
  - `no_output` : il n'y a pas de génération de fichier ASCII,
  - `ascii` : forme à personnaliser du fichier ASCII décrite au § 5.8 (Page 64), avec la dénomination `mon_modele-ascii`
  - `all_ascii` : génération de **TOUS** les fichiers ASCII de forme à personnaliser de nom `nnnn-mon_modele-ascii`, `nnnn` étant le numéro du modèle.

**Suffixes A, T, L, R :**

**A** Le suffixe A (Ex : `osc_adiaA`) permet d'utiliser l'ancienne formulation de la fréquence de Brunt-Väissälä.

**T, L** Les suffixes T & L (Ex : `osc_adiaT`) permettent de générer des fichiers de sortie ASCII exploitables directement par IDL. Ils ont été créés à l'initiative de T.Guillot selon les desiderata respectivement de T.Guillot (TR) et L.Bigot (LB); cf. § A.5 (Page 91), cf. § A.6 (Page 92) **Les suffixes T & L sont incompatibles.**

**R** Le suffixe R (Ex : `no_outputR`) permet de créer un fichier condensé pour le tracé du diagramme HR cf. § A.2 (Page 88) ; il ne contient que l'âge,  $L$  et  $T_{\text{eff}}$ . Ce fichier a pour extension `.LR`, Ex : `mon_modele.LR`, cf. § A.2.1 (Page 90) .
- `N_MAX=1000` : le nombre maximum de couches est défini par `MAX(N_MAX, nnnn)`, `nnnn` est un nombre maximal de couches fixé dans la routine `cesam.f` suivant le niveau de précision requis; le nombre minimal de couches est fixé à 500. **A condition** que la formation d'un fichier ASCII **soit requise** (`NOM_OUTPUT≠no_output`), une valeur négative de `N_MAX` permet de calculer le dernier modèle de l'évolution avec un nombre de couches égal à  $|\text{N\_MAX}|$ .
- `PRECISION='pr'` : niveau de précision requis. On trouve dans la Table 4.1 (Page 36), les valeurs des paramètres numériques utilisées suivant les

différentes options définies dans la routine `cesam.f`. Si le type de précision est codé en minuscules, Ex : 'sa', les variables quasi-statiques utilisées sont :  $\ln P$ ,  $\ln T$ ,  $(R/R_\odot)^2$ ,  $(L/L_\odot)^{2/3}$ ,  $(M/M_\odot)^{2/3}$ . S'il est codé en majuscules, Ex : 'SA', elles seront  $\ln P$ ,  $\ln T$ ,  $R/R_\odot$ ,  $L/L_\odot$ ,  $(M/M_\odot)^{1/3}$ .

**Suffixes A, T, L, M :** Tous les types de précision peuvent être affectés des suffixes A et/ou L et/ou M, Exemples : `prA`, `saLA`, `SALA ceM`. L'ordre des suffixes est indifférent, Exemple : `PRAL`.

**A :** ("ajuste") indique que le pas temporel sera ajusté de façon à atteindre précisément les objectifs d'arrêt de la NAMELIST NL\_EVOL *cf.* § 4.7 (Page 38).

**L :** ("lisse") indique qu'un lissage local des discontinuités de composition chimique sera effectué à la suite de chaque ajustement temporel.

**M :** ("maximum") indique que le nombre de couches sera inchangé au long de l'évolution. Il sera fixé au nombre maximal `N_MAX` indiqué dans la NAMELIST NL\_CESAM *cf.* § 4.5 (Page 33).

De plus des suffixes spécifiques peuvent affecter certains des types de précision implantés dans `CESAM2k3` :

- 'np/NP' (précision normale) utile pour se donner l'idée générale d'une évolution,
- 'pr/PR' (précision réaliste) pour une évolution sans une recherche particulière de précision c'est le "**DEFAULT**",
- 'sp/SP' (super précision) et 'sa/SA' (précision solaire) si on recherche une précision extrême. Avec 'sa/SA', à condition que la formation d'un fichier ASCII soit requise, les derniers modèles d'une évolution sont calculés avec le nombre maximum de couches `N_MAX`,
- 'av/AV, ce/CE, He' précisions pour des évolutions vers les stades avancés. 'ce/CE' (Céphéides) identique à 'AV' avec un contrôle du pas temporel dans la zone d'instabilité de façon à la traverser en 10 pas temporel environ, et création d'un fichier `mon_modele.ceph`, *cf.* § A.4 (Page 91).

**Suffixe O :** En codant 'ceO/CEO' le fichier ASCII de chaque modèle de la bande d'instabilité sera formé.

- 'He' (flash de l'hélium) identique à 'AV', le pas temporel minimal pouvant descendre au niveau de l'heure. La luminosité pouvant devenir négative dans le noyau la précision 'he' n'est pas prévue.
- 'rg' (réglages) avec ce type de précision le choix des paramètres numériques est confié à l'utilisateur/trice *averti/e*. Ils sont introduits par l'intermédiaire d'un fichier de nom 'reglages' ou, mieux 'mon\_modele.rg', placé dans l'environnement du calcul. On trouvera au § 5.9 (Page 65) un exemple ainsi que la description de son contenu. La précision 'rg' accepte les extensions de précision.

Les constantes de répartition sont fixées suivant le type de précision. Les valeurs les plus utilisées sont : `ctel=0`, `ctep=-1`, `ctem=15`, `cter=0`, `ctet=-1`. Pour utiliser des paramètres différents *cf.* § 5.9 (Page 65).

---

3. Les "/" indiquent une alternative.

TABLE 4.1 – Valeurs des paramètres numériques de la `precision` par défaut (`pr/PR`) et valeurs adaptées aux autres précisions. Les dénominations s’interprètent de la façon suivante : `pr/PR` : précision réaliste (**”le défaut”**), `sp/SP` : super précision, `lm/LM` : pour petites masses (low mass), `av/AV` : pour modèles pouvant atteindre les stades avancés, `np/NP` : pour modèles de précision normale, `co/CO` : pour modèles destinés à l’exploitation des résultats du satellite CoRoT, `sa/SA` : pour modèles solaires. Pour les Céphéides utiliser le type de précision `ce/CE`, analogue à `av/AV`, et produisant des fichiers d’exploitation spécifiques. La signification des différents paramètres numériques est explicitée au § 5.9 (Page 65) et lignes 805-1237 dans la routine `cesam.f` du sous-directory `SOURCE`. La valeur initiale du pas temporel et ses limites de variations ultérieures dépendent de la masse et du type de précision requis ; les valeurs portées dans ce tableau n’ont qu’une valeur indicative. L’utilisation des minuscules/MAJUSCULES (Exemple : `pr/PR`) du type de précision définit le type de variables quasi-statiques à utiliser, *cf.* § 4.5 (Page 33). Une astérisque\* signale ces paramètres peuvent être modifiés lorsque la masse initiale diffère de l’unité ou que la diffusion microscopique est prise en compte.

	pr/PR	sp/SP	lm/LM	av/AV	np/NP	co/CO	sa/SA
<code>m_qs</code>	2				1		
<code>m_ch*</code>	2	3				3	3
<code>m_rot</code>	2				1	3	
<code>m_tds</code>	2						
<code>m_ptm</code>	2						
<code>ordre</code>	4				1		
<code>precic</code>	$10^{-4}$						
<code>precix*</code>	$10^{-3}$	$10^{-4}$			0.005	$10^{-4}$	$10^{-5}$
<code>precit*</code>	0.1	0.02				0.05	0.02
<code>psi0</code>	0.03	0.02			0.1		0.02
<code>loc_zc</code>	$10^{-3}$	$10^{-4}$			0.005	$10^{-4}$	$10^{-5}$
<code>dt0*</code>	10						
<code>dtmin*</code>	1an			1sec			
<code>dtmax*</code>	100		300		300	50	50
<code>d_lum</code>	0.1						
<code>d_press</code>	0.1						
<code>d_ray</code>	0.1						
<code>d_temp</code>	0.1						
<code>age_car*</code>	1000						
<code>ini0</code>	3						
<code>n_atm</code>	75	100			50	100	100
<code>kipp</code>	.TRUE.	.FALSE.				.FALSE.	.FALSE.
<code>ctel</code>	0						
<code>ctep</code>	-1						
<code>ctem</code>	15						
<code>cter</code>	0.						
<code>ctet</code>	-1.						

TABLE 4.2 – Valeurs des paramètres numériques (suite).

	pr/PR	sp/SP	lm/LM	av/AV	np/NP	co/CO	sa/SA
dn_fixe	0.05						
dpsi	0.05						
mu_saha	.TRUE.		.FALSE.		.FALSE.		
ajuste*	.FALSE.						
lisse*	.FALSE.						
npt_lisse*	10						
q0	0.05	0.01	0.	0.	0.	0.01	0.01
l0	0					5	5
new_bv	.TRUE.						.FALSE.
fcv	.FALSE.		.TRUE.		.TRUE.		
no_discon	.FALSE.			.TRUE.			
n_min_ZC	5						
d_zc	10	5	100		100	2	2
l_fac	.FALSE.	.TRUE.				.TRUE.	.TRUE.

## 4.6 Arguments de la NAMELIST NL\_MASS

- MTOT=1.d0 : masse totale initiale, en  $M_{\odot}$ ,
  - NOM\_PERTM : nom du formalisme de perte de masse externe :
    - pertm\_dominik : perte de masse selon l'article "stationary dust-driven winds", de Dominik & al. 1990 A&A, 240, 365
    - pertm\_ext : perte/gain de masse externe linéaire en fonction du temps.
    - pertm\_msol : perte/gain de masse externe linéaire en fonction du temps, la masse de l'étoile restant au moins égale à  $1 M_{\odot}$ ,
    - no\_pertm : pas de perte/gain de masse externe,
    - pertm\_reimers : perte de masse solaire empirique de Reimers cité dans l'article Rasio & al. ApJ 470, 1187, 1996 (taux=0.6d-13 Msol/y)
    - pertm\_reim-domi : association de pertm\_reimers et pertm\_dominik,
    - pertm\_waldron : perte de masse empirique de Waldron (1985).
- Si le nom du formalisme de perte de masse externe est affecté d'un '\$', Exemple : pertm\_msol\$, il sera tenu compte de la perte de masse 'interne' due aux réactions thermonucléaires ( $E = mc^2$ ), dont on tiendra compte avec no\_pertm\$<sup>4</sup>.
- MDOT=1.d-14 : taux de perte/gain de masse, en  $M_{\odot}/\text{an}$ , ce taux est positif pour un gain de masse, négatif pour une perte de masse. Dans le cas standard, la composition chimique de l'apport/retrait de masse est celle de la couche la

4. Cette perte est de l'ordre de  $10^{-14} M_{\odot}/\text{an}$  pour le soleil.

plus externe du modèle. Voir § 5.5 (Page 62) comment personnaliser, si besoin, cette composition chimique.

## 4.7 Arguments de la NAMELIST NL\_EVOL

- AGEMAX=4.5d4 : âge maximal à atteindre en million d’années,
- ARRET='post' : arrêt sur la ZAMS ('zams'), à la fin de la ZAMS ('post'<sup>5</sup>), au début de la combustion de l’hélium ('cohe') ou du carbone ('coca') ou du néon ('cone') ou de l’oxygène ('coox') ou autre ('else').
- DTLIST=1.d3 : intervalle de temps minimum, en million d’années, séparant deux listes détaillées du modèle (fichier `mon_modele.lis`). Avec `DTLIST < 0`, Ex : `DTLIST=-50.d0`, possibilité de construire des fichiers de sortie ASCII ou de reprise à intervalles réguliers ici tous les 50 modèles. Ces fichiers seront dénommés `nnnn-mon_modele-ad.osc`, et/ou `nnnn-mon_modele.B.rep`; `nnnn` étant le numéro du modèle. Avec `DTLIST=-1.d0` ces fichiers intermédiaires seront formés pour tous les modèles.
- LOG\_TEFF=-3.7d0 : arrêt si cette valeur de  $\log_{10} T_{\text{eff}}$  a été traversée, dans le sens croissant si `log_teff` est positif, dans le sens décroissant si `log_teff` est négatif,
- NB\_MAX\_MODELES=5000 : arrêt après avoir calculé le nombre de modèles indiqué, éventuellement 0. A l’issue de chaque pas temporel le fichier binaire `mon_modele.B.rep` remplace celui obtenu au pas temporel précédent. Le nombre maximal est 99999. Il est possible d’écrire et de conserver ces fichiers binaires à intervalles réguliers. Pour ce faire coder, par exemple, `NB_MAX_MODELES=-5000` et `DTLIST=-10.d0`, alors tous les 10 modèles, le fichier binaire de reprise sera écrit et conservé dans l’environnement avec le nom `nnnn-mon_modele.B.rep` Exemple : `0145-sun.B.rep`. Le numéro 0000 est affecté aux modèles d’initialisation de séquence principale d’âge zéro ou de pré-séquence principale ; en cas de reprise, les numéros reprennent à partir de celui du modèle repris.
- HE\_CORE=0.1d0 : arrêt, dès que la masse du cœur d’hélium atteint la valeur `he_core` ( $M_{\odot}$ ),
- R\_STOP=0.9d0 : arrêt, lorsque la valeur du rayon bolométrique traverse `R_STOP` en  $R_{\odot}$ ,
- T\_STOP=1.d7 : arrêt, si au centre, la température dépasse cette valeur,
- X\_STOP=0.1d0 : arrêt si cette valeur de X, au centre, est traversée.

Si le suffixe A affecte la variable de précision, (`PRECISION='saA'`) cf. § 4.5 (Page 33), le pas temporel est ajusté de façon à obtenir les valeurs requises.

### 4.7.1 Caractérisation des types de modèles

Au cours d’une évolution, à l’issue de chaque initialisation et de chaque pas temporel, un fichier binaire contenant l’ensemble des variables du modèle est créé. Ces fichiers servent d’une part à poursuivre le calcul de l’évolution et d’autre part à exploiter les modèles obtenus, calcul des fréquences d’oscillation par exemple. A l’issue

---

5. 'tams' est accepté.

de chaque pas temporel le fichier du nouveau modèle remplace celui du précédent. Les modèles qui correspondent à des époques caractéristiques d'une évolution sont conservés, par exemple à l'amorçage du  $3\alpha$ . Ces fichiers dits *permanents* permettent, en particulier, de reprendre l'évolution à ces époques caractéristiques. Sous réserve d'acceptation par CESAM2k, il est alors possible de modifier un ou plusieurs paramètres ou d'introduire des modifications de la physique, par exemple la longueur de mélange ou l'opacité; modifications qui seront prises en compte dans la suite de l'évolution. Ces modèles binaires se différencient par leurs extensions qui sont les suivantes :

- 'hom' : Extension du nom d'un fichier de modèle de ZAMS homogène d'âge 0, Exemple : `mon_modele.hom`, ces fichiers sont permanents.
- 'pms' : Extension du nom d'un fichier de modèle de PMS homogène d'âge 0, Exemple : `mon_modele.pms`, ces fichiers sont permanents.
- 'rep' : Extension du nom d'un fichier d'un modèle en cours d'évolution, Exemple : `mon_modele.rep`, Au fur et à mesure de l'évolution ces fichiers, dits de "reprise", sont effacés et remplacés par celui du suivant, ils correspondent donc au dernier modèle calculé.
- 'zams' : Un modèle décrivant la pré-séquence principale est identifié comme modèle de ZAMS dès que la génération d'énergie d'origine nucléaire surpasse celle d'origine graviphique. A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension `zams`, Exemple : `mon_modele.zams`.
- 'post' ou 'tams' : Un modèle décrivant la séquence principale est identifié comme modèle de post-main séquence dès que l'abondance centrale d'hydrogène devient inférieure à 0.01, A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension 'post' ou 'tams'. Exemple : `mon_modele.post` ou `mon_modele.tams`.
- 'cohe' : Un modèle décrivant la post-séquence principale est identifié comme modèle avec combustion d'hélium dès que la température centrale dépasse  $1.10^8\text{K}$ . A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension 'cohe', Exemple : `mon_modele.cohe`
- 'coca' : Un modèle décrivant la séquence de combustion de l'hélium est identifié comme modèle avec combustion du carbone dès que la température centrale dépasse  $7.10^8\text{K}$ . A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension 'coca'.
- 'cone' : Un modèle décrivant la séquence de combustion de carbone est identifié comme modèle avec combustion du néon dès que la température centrale dépasse  $1.10^9\text{K}$ . A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension 'cone'.
- 'coox' : Un modèle décrivant la séquence de combustion du néon est identifié comme modèle avec combustion d'oxygène dès que la température centrale dépasse  $1.910^9\text{K}$ . A l'époque de la transition le fichier de reprise est copié dans un fichier permanent d'extension 'coox'.

## 4.8 Arguments de la NAMELIST NL\_CHIM

- GRILLE\_FIXE=.TRUE. : pour utiliser une grille fixe pour la composition chimique et la diffusion du moment cinétique<sup>6</sup>,
- NOM\_ABON : nom de la mixture initiale<sup>7</sup> :
  - enhan\_al : abondances  $\alpha$ -enhanced de Allard,
  - enhan\_cha : abondances  $\alpha$ -enhanced de Chaboyer,
  - enhan\_w : abondances  $\alpha$ -enhanced de Weiss,
  - meteorites\_ag : abondances météoritiques de Anders & Grevesse (1989),
  - meteorites\_gs : abondances météoritiques de Grevesse & Sauval (1998),
  - meteorites\_ags05 : abondances météoritiques de Asplund et al. (2005),
  - meteorites\_agss : abondances météoritiques de Asplund et al. (2009)
  - mixture : forme à personnaliser de la compilation d'abondances décrite au § 5.2 (Page 60),
  - solaire\_gn : mixture solaire de Grevesse & Noels (1993),
  - solaire\_gs : mixture solaire de Grevesse & Sauval (1998).
  - solaire\_ags\_03 : mixture solaire de Asplund (2003)
  - solaire\_ags\_05 : mixture solaire de Asplund et al. (2005)
  - solaire\_agss : abondances météoritiques de Asplund et al. (2009).

Ces abondances sont initialisées dans la routine `abon_ini.f`.

- MODIF\_CHIM=.FALSE. : signifie que CESAM2k devra utiliser, sans modification, les données du sous programme `abon_ini.f`. Si tel n'est pas le cas, coder MODIF\_CHIM=.TRUE., CESAM2k recherchera alors dans l'environnement les fichiers permettant les modifications souhaitées. Ces fichiers de modification sont :
  - modif\_mix : composition chimique initiale modifiée, cf. § 5.4 (Page 62),
  - rap\_iso : rapports isotopiques modifiés, cf. § 5.3 (Page 61),
  - planet : composition chimique des planétoïdes différente de celle du modèle, cf. § 5.6 (Page 63),
  - vent : composition chimique du vent différente de celle du modèle, cf. § 5.5 (Page 62).

CESAM2k invitera à supprimer ces fichiers avant d'effectuer des calculs qui n'en tiendraient pas compte. Cette disposition **à but sécuritaire** a pour fin d'éviter de prendre en compte des fichiers qui, par mégarde, n'auraient pas été soustraits de l'environnement. Si aucun de ces fichiers n'est détecté, CESAM2k le signalera et l'instruction MODIF\_CHIM=.TRUE. sera ignorée.

- GARDE\_XISH=.FALSE. : ce paramètre fait l'objet du paragraphe suivant,
- X0=0.7 : abondance initiale, par unité de masse, de H,
- Y0=0.28 : abondance initiale, par unité de masse, de He,
- ZSX0=0.d0 : valeur initiale de Z/X,
  - Les abondances du modèle initial sont déterminées à partir des quantités X0, Y0 et ZSX0 ; en raison de la relation  $1 = X + Y + Z$ , seulement 2 de ces 3 quantités sont à prendre en considération. La valeur initiale de Y utilisée dans le calcul est **toujours** Y0 ; elle est fractionnée en les divers isotopes retenus pour décrire l'évolution de l'hélium,

6. Inactivée dans la présente version de CESAM2k.

7. cf. équation d'état OPAL : WEB site [http://www-phys.llnl.gov/V\\_Div/OPAL/opal.html](http://www-phys.llnl.gov/V_Div/OPAL/opal.html).

- si  $ZSX0 \leq 0$ , l'abondance initiale d'hydrogène est initialisée à  $X0$  et  $Z = 1 - X - Y$  est déduit de  $X0$  et  $Y0$ ,
- sinon, *i.e.*  $ZSX0 > 0$ , l'abondance initiale d'hydrogène est déduite de  $Y0$  et de  $ZSX0$ , la valeur indiquée pour  $X0$  étant ignorée.

Les valeurs globales initiales  $X0$ ,  $Y0$  et  $Z0$  sont fractionnées entre les divers éléments et chacun d'eux entre leurs isotopes.

Avec diffusion et/ou après un premier dredge-up, la composition chimique des couches extérieures, *i.e.* celle observée, diffère de la composition chimique initiale. Il y a lieu d'ajuster la valeur initiale de la composition chimique, afin d'obtenir celle observée à l'issue d'une évolution. Les observations portant sur les rapports métal/H, ou encore  $[Fe/H]$ , à l'initiative de F.Thévenin, CESAM2k offre divers outils pour réaliser cet ajustement :

- Conserver le rapport  $Z/X$  de la mixture initiale, au lieu de  $ZSX0$ , objet du § 4.8.1 (Page 41),
- utiliser une mixture originale, *cf.* § 5.2 (Page 60),
- modifier les rapports isotopiques, *cf.* § 5.3 (Page 61),
- modifier les rapports d'abondances de la mixture initiale, *cf.* § 5.4 (Page 62).

#### 4.8.1 Alternative : conservation métal/H ou métal/Z

Dans CESAM2k, l'utilisation de tabulations est incontournable pour le calcul de l'opacité et de l'équation d'état. Dans les tables les valeurs des paramètres globaux de composition chimique  $X$ ,  $Y$ ,  $Z$  sont fixés. Il y a des incohérences chroniques entre les valeurs *fixes* utilisées pour les tabulations et celles *variables* obtenus au cours de l'évolution. Le rôle de GARDE\_XISH est de préciser la nature de ces incohérences que l'on doit accepter, faute de mieux.

On désigne par *mixture de base* la mixture proposée par CESAM2k, Exemple `solaire_gn`, ou celle fournie par l'utilisateur *cf.* § 5.2 (Page 60). Elle inclut éventuellement les aménagements décrits aux § 5.4 (Page 62), ou encore § 5.3 (Page 61).

La *mixture initiale* est celle utilisée pour initialiser une évolution. Elle est déduite de la mixture de base en tenant compte des indications codées dans le fichier de données.

La valeur initiale de  $Y$  est **TOUJOURS** la valeur lue dans le fichier de données, *i.e.*  $Y0$ , par exemple la valeur primordiale  $Y0=0.27$ .

`GARDE_XISH=.TRUE.` (*respt.* `GARDE_XISH=.FALSE.`) permet de fixer la valeur initiale de  $Z$  (*respt.*  $X$ ) de façon à conserver le rapport métal/H (*respt.* métal/Z)<sup>8</sup>.

Ainsi, en codant<sup>9</sup> :

- `GARDE_XISH=.TRUE.`, la valeur de  $Z/X$  de la mixture initiale sera déduite des rapports d'abondances métal/H de la mixture de base et non pas de celle,  $ZSX0$ , lue dans le fichier de données. La métallicité utilisée  $Z/X$ , **sera différente** de celle  $ZSX0$  indiquée dans le fichier de données, sauf dans le cas particulier où les valeurs de  $Y0$  et de  $ZSX0$  correspondent **exactement** à celles de la mixture

8. Alternative réalisée dans la routine `abon_ini.f`.

9. De meilleures dénominations seraient, par exemple : `GARDE_XISH_MIX`, `GARDE_XISH_MIX_INI`, `GARDE_ZSX` ou encore `GARDE_ZSX_MIX_INI`, *etc...*

de base. Codage qu'il convient d'utiliser quand on désire que la mixture initiale et la mixture de base soient "*physiquement*" identiques.

- GARDE\_XISH=.FALSE., la valeur ZSX0 lue dans le fichier de données, sera utilisée. Les rapports métal/Z des éléments chimiques seront ceux de la mixture de base. Codage qu'il convient d'utiliser pour une *calibration solaire*, pour laquelle on vise à atteindre une valeur fixée de  $Z/X$ , en utilisant, pour l'initialisation, le Y0 du fichier de données et les rapports métal/Z de la mixture de base qui peuvent éventuellement différer du  $Z/X$  à atteindre.

**PB** L'utilisation de GARDE\_XISH=.TRUE. *EST DELICATE*, il est conseillé de bien vérifier, dans le fichier mon\_modele.lis, que les valeurs initiales des abondances utilisées par CESAM2k correspondent bien à ce que l'on désire.

**PB** Il faut être conscient que les modifications de la mixture ne peuvent être prises en compte, ni dans les opacités, ni dans l'équation d'état.

## 4.9 Arguments de la NAMELIST NL\_CONV

- NOM\_CONV : nom de la routine calculant le gradient de température dans les zones convectives.
- conv\_a0 : convection MLT, avec longueur de mélange  $l \rightarrow 0$  aux limites ZR/ZC,
- conv\_cgm\_reza : convection suivant Canuto & Mazitelli (1991), avec  $l = \alpha H_p$ , tient compte de la prescription de Bernkopf,
- conv\_cm : convection suivant Canuto & Mazitelli (1991), avec  $l = \alpha H_p$ ,
- conv\_cml : convection suivant Canuto & Mazitelli avec avec longueur de mélange égale à la plus courte distance des bords de la ZC,
- conv\_cm\_reza : convection suivant Canuto & Mazitelli (1991), avec  $l = \alpha H_p$ , tient compte de  $\delta$ ,
- conv\_jmj : convection MLT avec  $l = \alpha H_p$  (Böhm-Vitense, 1958).
- Suffixe **S** : Pour les modèles utilisant les types de précision "corot" (co, CO), "super précision" (sp, SP), "précision solaire" (sa, SA), CESAM2k place chaque limite ZR/ZC à une distance d'un point de grille au plus égale à une valeur fixée. Pour éviter cette disposition qui ralentit le déroulement du calcul, accoler le suffixe **S** au nom de la routine de convection, Exemple : conv\_jmjS.
- alpha=1.7d0 : longueur de mélange,
- ovshts=0.05d0 : coefficient d'overshoot supérieur, avec une valeur positive il y aura une **discontinuité** du gradient pris égal au gradient adiabatique dans la partie overshootée, avec une valeur négative il y a **continuité** du gradient pris égal au gradient radiatif dans la partie overshootée,
- ovshti=0.1d0 : coefficient d'overshoot inférieur, comme pour l'overshoot supérieur, dans la partie overshootée une valeur positive affecte le gradient adiabatique, une valeur négative affecte le gradient radiatif,
- jpz=.TRUE. : utilisation des prescriptions de JpZ,

- `cpturb=0.d0` : coefficient de pression turbulente, **indisponible dans cette version** de CESAM2*k*,
- `ledoux=.TRUE.` : utilisation du critère de Ledoux. Avec semi-convection on doit coder `ledoux=.FALSE.` le formalisme de Ledoux faisant partie intégrante de la description de la semi-convection, *cf.* §4.10 (Page 43).

## 4.10 Arguments de la NAMELIST NL\_DIFF

- `DIFFUSION=.TRUE.` : on tiendra compte de la diffusion microscopique des éléments chimiques.
- `NOM_DIFFM` : nom de la routine calculant les coefficients de diffusion microscopique :
  - `diffm_br` : calcul des coefficients de diffusion microscopique suivant le formalisme de Burgers en tenant compte, éventuellement, des accélérations radiatives,
  - `diffm_mp` : calcul des coefficients de diffusion microscopique suivant le formalisme simplifié de Michaud & Proffit,
  - `diffm_0` : coefficients de diffusion microscopique nuls.
- `NOM_DIFFT` : nom de la routine calculant les coefficients de diffusion turbulente :
  - `difft_nu` : calcul des coefficients de diffusion turbulente incluant la diffusivité radiative.
  - `difft_nut` : calcul des coefficients de diffusion turbulente incluant la diffusivité radiative et la diffusivité turbulente dans la tachocline de la zone convective externe suivant Castro et al. (2007).
  - `difft_gab` : calcul des coefficients de diffusion turbulente selon une prescription simple de M.Gabriel.
  - `difft_smc` : calcul des coefficients de diffusion turbulente incluant la semi-convection et la diffusivité radiative.
  - `difft_sun` : coefficients de diffusion turbulente, sous la zone convectivesolaire suivant Gabriel (1997).
- **Suffixe V** : Quand le suffixe "V" est accolé à la fin du nom de la routine de diffusion turbulente à utiliser, Exemple : `difft_nuV`, le coefficient de diffusion permettant le mélange convectif est pris égal à  $1/3V_{\text{conv}}L_{\text{mel}}$ .
- `D_TURB=1.d1` : coefficient de diffusion turbulente isotrope,
- `RE_NU=1.d0` : coefficient de diffusivité radiative,
- `NOM_FRAD` : nom de la routine de calcul des accélérations radiatives,
  - `frad_test` : utilisation d'accélérations radiatives pour tests,
  - `alecian_bp` : calcul des accélérations radiatives suivant le second formalisme de G.Alécian<sup>10</sup>,
  - `no_frad` : les accélérations radiatives seront ignorées.

---

10. Les données peuvent être obtenues sur simple demande auprès de Bernard.Pichon@oca.eu

## 4.11 Arguments de la NAMELIST NL\_ROT

**PB** La diffusion du moment cinétique n'est pas disponible dans la présente version de CESAM2*k*.

- W\_ROT=50.d0 : vitesse angulaire initiale.
- UNIT : unité utilisée pour la vitesse angulaire initiale. différents choix sont proposés de façon à s'adapter facilement aux observations :
  - jours : période initiale en jours, correspond à la vitesse angulaire initiale de rotation de la couche externe,
  - kms/s : vitesse de la couche externe en kilomètres/seconde, le rayon du modèle d'initialisation étant utilisé pour déterminer la vitesse angulaire initiale, il convient d'ajuster ce rayon en reprenant plusieurs fois le calcul du modèle d'âge 0,
  - rad/s : la vitesse angulaire initiale est en radians/seconde.
- NOM\_DIFFW : nom de la routine du calcul des coefficients de diffusion du moment cinétique :
  - diffw\_p03 : coefficients de diffusion du moment cinétique suivant Palacios et al. (2003)
  - diffw\_mpz : coefficients de diffusion du moment cinétique suivant Mathis et al. (2004)
  - diffw\_0 : coefficients de diffusion du moment cinétique nuls.
  - diffw\_cte : coefficients de diffusion du moment cinétique constants : ( $D_{\text{eff}} = 300$ ,  $D_{\text{h}} = 1\,000\,000$ ,  $D_{\text{v}} = 250$ ).
  - diffw\_toul : coefficients de diffusion du moment cinétique suivant Castro et al. (2007)
- NOM\_THW : Nom de la théorie utilisée pour l'évolution de la vitesse angulaire :
  - rot\_0 : Vitesse angulaire nulle, le modèle ne tient pas compte de la rotation.
  - rot\_cte : Vitesse angulaire constante au cours de l'évolution, *i.e.* rotation rigide.
  - cons\_glob\_mnt\_cin : Evolution avec rotation rigide, et conservation globale du moment cinétique. A l'issue de chaque pas temporel, la vitesse angulaire, constante dans tout le modèle, est réajustée de façon à conserver le moment cinétique total.
  - diff\_tz97 : Evolution avec diffusion du moment cinétique, suivant la théorie de Talon et al. (1997), **indisponible dans cette version** de CESAM2*k*.
  - diff\_mz04 : Evolution avec diffusion du moment cinétique, suivant la théorie de Mathis & Zahn (2004), **indisponible dans cette version** de CESAM2*k*.
  - cons\_loc\_mnt\_cin : évolution avec conservation locale du moment cinétique avec rotation rigide des zones convectives.
- NOM\_PERTW : Nom de la routine de calcul de perte de moment cinétique,
  - pertw\_sch : variation de vitesse angulaire proportionnelle à  $\Omega^3$  (Schumannish), **indisponible dans cette version** de CESAM2*k*,
  - pertw\_loc : perte de moment cinétique proportionnelle à l'énergie cinétique de rotation locale, **indisponible dans cette version** de CESAM2*k*,
  - pertw\_ptm : perte de moment cinétique, conséquence d'une perte de masse, **indisponible dans cette version** de CESAM2*k*,

- `pertw_0` : pas de perte de moment cinétique <sup>11</sup>.
- `P_PERTW=-1.d-9` : paramètre de perte de moment cinétique attaché respectivement à chacune des routines précédentes
- `NOM_DES_ROT='end_mod'` : Contrôle de la génération de fichiers ASCII des variables de la diffusion du moment cinétique, *cf.* § A.3 (Page 90). Si le dessin on-line n'est pas requis (`NOM_DES=no_des`), le fichier postscript des dessins des variables de la rotation est néanmoins automatiquement créé, **indisponible dans cette version** de CESAM2*k*.
- `no_des` : Pas de formation de fichier.
- `end_evol` : Formation du fichier à l'issue de l'évolution, le nom du fichier est `mon_modele_coeff_rota.dat`.
- `all_mod` : Formation du fichier à l'issue de chaque pas temporel, le nom du fichier est `nnnn-mon_modele_coeff_rota.dat`, *nnnn* étant le numéro du modèle. Tous les fichiers ainsi créés sont permanents.
- `end_mod` : Formation du fichier à l'issue de chaque pas temporel, le nom du fichier est `mon_modele_coeff_rota.dat`, seul le dernier fichier créé étant disponible.
- `all_iter` : Formation du fichier à l'issue de chaque itération de la résolution du système d'équations de la diffusion du moment cinétique ; fichiers destinés aux mises au point.

## 4.12 Arguments de la NAMELIST NL\_ETAT

- `NOM_ETAT` : nom de la routine d'équation d'état :
  - `etat_ceff` : équation d'état de Eggleton et al. (1973), avec corrections Coulombiennes (*appel à etat\_eff en cas de difficulté*).
  - `etat_eff` : équation d'état de Eggleton et al. (1973), (*appel à etat\_gong2 en cas de difficulté*).
  - `etat_gong1` : équation d'état de `gong1` selon Christensen-Dalsgaard (1988), seuls H et He sont pris en compte et supposés totalement ionisés.
  - `etat_gong2` : équation d'état de `gong2` selon Christensen-Dalsgaard (1988), seuls H et He4 sont pris en compte, dégénérescence ignorée.
  - `etat_mhd` : équation d'état de Mihalas, Hummer & Dappen (*appel à etat\_eff en cas de difficulté*) utilise les tables en binaire `mhd1.bin ... mhd7.bin` construites dans le sous-directory `SUN_STAR_DATA`, voir § 3.2.4 (Page 21).
  - `etat_irwin` : équation d'état FreeEOS de A. Irwin. La source en est téléchargeable sur <http://freeeos.sourceforge.net>. Les extensions 1, 2, 3, 4, 5, 6 Ex : `etat_irwin1a`, donnent accès à différents niveaux de précision/rapidité et de la cohérence avec d'autres source d'EOS et/ou d'Opacité existantes. Une possibilité de personnalisation est offerte par l'intermédiaire d'un fichier de nom `opteos` disposé dans l'environnement du calcul. Ces options sont décrites dans la routine `etat_irwin.f` et dans le fichier (texte) `etat_irwin_explik.f` du sous-directory `SOURCE`. L'option 1 : `etat_irwin1` est la plus précise et la plus lente, l'option 5 : `etat_irwin5`,

---

11. Seule routine de perte de moment cinétique disponible dans la présente version de CESAM2*k*.

est la plus rapide et la moins précise. Cette équation d'état utilise la bibliothèque dont la création est décrite §3.2.5 (Page 23). *Ce sont les directives de Y.Lebreton qui ont permis l'implantation de cette EOS dans CESAM2k.*

- `etat_opal` : équation d'état opal 1991, (*appel à `etat_eff` en cas de difficulté*) utilise les tables en binaire `eos_opal*.bin` construites dans le sous-directory `SUN_STAR_DATA`, voir §3.2.1 (Page 18).
- `etat_opalZ` : équation d'état de opal 2001, (*appel à `etat_eff` en cas de difficulté*) utilise les tables en binaire `eos_opalZ*.bin` construites pour **Z fixé** dans le sous-directory `SUN_STAR_DATA`, cf. §3.2.1 (Page 18). Par rapport à la version 1991, les données de la version 2001 ont été étendues et améliorées, des bugs ont été corrigés. Les données ne permettent pas de dépasser  $X = 0.8$ . Les fichiers de données 2001 et 1991 sont incompatibles.
- `etat_opalX` : équation d'état de opal 2001, se différenciant de `etat_opalZ` par l'extension du domaine de l'abondance d'hydrogène,  $X \in [0, 1]$ , dans le but de calculer des modèles sans zone convective externe étendue *avec diffusion*. Le prix à payer est  $Z \equiv 0$ , ce qui n'est pas très pénalisant, l'équation d'état étant peu sensible à  $Z$ . `etat_opalX` utilise la table en binaire `eos_opalX` construite pour **Z=0 fixé** dans le sous-directory `SUN_STAR_DATA`, cf. §3.2.1 (Page 18).
- `etat_opal5` : Equation d'état similaire aux précédentes. La construction de la table, pour une valeur de métallicité  $Z$  donnée, est effectuée dans le sous directory `OPAL2005_tables.gz`, cf. §3.2.3 (Page 21). Dans le fichier de données, il ne sera pas nécessaire d'indiquer que la table de nom `op5_0174.bin` (ici  $Z = 0.0174$ ) est ou non compressée.
- `etat_saha` : Equation d'état tabulée à  $Z$  variable communiquée et adaptée à CESAM2k par V.A. Baturin & A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State. cf. §3.2.6 (Page 24)
- `F_EOS='eos_opal_250.bin',7*' '` : noms des fichiers d'équation d'état (au plus 8). Ne **pas** indiquer le nom des fichiers pour l'EOS `etat_saha`.

### 4.13 Arguments de la NAMELIST NL\_OPA

- `NOM_OPA` : nom de la routine de calcul des opacités moyennes de Rosseland :
- `opa_gong` : opacités simplifiées (Kramers amélioré) selon Christensen-Dalsgaard (1988).
- `opa_houdek12` : opacités de Houdek version 12, interpolation par rationnal B-spline; utilise les tables en binaire construites dans le sous-directory `SUN_STAR_DATA/H12`, voir §3.1.2 (Page 16). Les opacités Houdek contiennent *ad initio* des opacités conductives.
- `opa_int_zsx` : Opacités OPAL93+Kurucz raccord d'Yveline, interpolations linéaires, très utiles pour des tests.
- `opa_opal2_co`, `opa_opal2_cno` : Opacités OPAL avec corrections pour  $Z > 0.1$ , cf. §?? (Page ??)
- `opa_opalC0` : Opacités OPAL avec corrections pour  $Z > 0.1$  ne portant que sur C et O.

- `opa_yveline` : opacités OPAL+Alexander interpolation et raccord d'Yveline; utilise les tables en binaire `opa_yveline*.bin` construites dans le sous directory `SUN_STAR_DATA`, voir § 3.1.1 (Page 15).
- `opa_yveline_jorgen` : identique à la précédente pour les opacités Asplund et al. (2005) avec la correction de Jorgen (astro-ph 0811.100v1 6 Nov 2008) qui permet de retrouver le "modèle S".
- `opa_yveline_lisse` : opacités OPAL+Alexander raccord d'Yveline, interpolation linéaire ou lissage; utilise les tables en binaire `opa_yveline*.bin` construites dans le sous-directory `SUN_STAR_DATA`, voir § 3.1.1 (Page 15).
- `F_OPA='opa_yveline.bin', '/H12/OPINTPATH', 6*` : noms des fichiers d'opacité (au plus 8). Pour les opacités Houdek, voir les commentaires au début de la routine `opa_houdek12` de la SOURCE, en cas de difficulté les opacités `opa_yveline` étant utilisées par défaut.
- `NOM_OPA_COND` : nom de la routine de calcul de l'opacité conductive.
  - `opa_cond_iben` : opacité conductives de Iben (1975)
  - `opa_cond_mestel` : opacité conductives de Mestel (Cox & Giuli, 1968)
  - ' ' : Opacité conductive ignorée.

Lorsque la métallicité  $Z > 0.1$ , Ex : au centre lors de la combustion de l'hélium, la matière est supposée totalement ionisée, `CESAM2k` simplifie le calcul de l'opacité en utilisant le formalisme free-free de Kramers, Kippenhahn & Weigert (1991) (17.5-17.6), avec la correction de Compton, Cox & Giuli (1968) (§16.a, table 16.1).

## 4.14 Arguments de la NAMELIST NL\_NUC

- `NOM_NUC` : nom de la routine de calcul des taux des réactions thermonucléaires et d'initialiation de la composition chimique :
  - `pp1` : calcul simplifié du cycle PP (ne permet pas la diffusion microscopique).
  - `pp3` : réactions PP, 3 éléments H, He3, He4 avec H2, Li7, Be7 à l'équilibre; a pour vocation les tests de mise au point, permet la diffusion microscopique, intervalle de tabulation : [1MK,80MK]..
  - `ppcno9` : réactions PP+CNO, 9 éléments, H2, Li7, Be7 à l'équilibre, intervalle de tabulation : [1MK,40MK].
  - `ppcno9Fe` : réactions PP+CNO, 9 éléments+Fe, H2, Li7, Be7 à l'équilibre, intervalle de tabulation : [1MK,80MK].
  - `ppcno10` : réactions PP+CNO, 10 éléments, H2, Be7 à l'équilibre, intervalle de tabulation : [0.5MK,80MK].
  - `ppcno10Fe` : réactions PP+CNO, 10 éléments+Fe, H2, Be7 à l'équilibre, intervalle de tabulation : [0.5MK,80MK].
  - `ppcno10K` : réactions PP+CNO, 10 éléments+K, H2, Be7 à l'équilibre, intervalle de tabulation : [0.5MK,80MK].
  - `ppcno10BeBFe` : réactions PP+CNO, 10 éléments + Li6, Be9, B11, Fe avec H2 et Be7 à l'équilibre, intervalle de tabulation : [0.5MK,80MK].
  - `ppcno11` : réactions PP+CNO, 11 éléments, Be7 à l'équilibre, intervalle de tabulation : [0.5MK,80MK].
  - `ppcno12` : réactions PP+CNO, 12 éléments, intervalle de tabulation : [1MK,80MK].

- `ppcno12Be` : réactions PP+CNO, 12 éléments + Be9, intervalle de tabulation :[0.5MK,80MK].
  - `ppcno12Li` : réactions PP+CNO, 12 éléments + Li6, intervalle de tabulation :[0.5MK,80MK].
  - `ppcno12BeBFe` : réactions PP+CNO, 12 éléments + Li6, Be9, B11, Fe, intervalle de tabulation :[0.5MK,80MK].
  - `ppcno3a9` : réactions PP+CNO+3 $\alpha$ , 9 éléments, H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,80MK].
  - `ppcno3a12Ne` : réactions PP+CNO+3 $\alpha$ +carbone, 12 éléments dont Ne22, H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,800MK].
  - `ppcno3aco` : réactions PP+CNO+3 $\alpha$ +carbone+néon+oxygène, 17 éléments H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,3GK].
  - `ppcno3acos` : réactions PP+CNO+3 $\alpha$ +carbone+néon+oxygène, 19 éléments H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,3GK].
- L'étendue de l'intervalle de tabulation est fixée en fonction de la zone d'efficacité du réseau nucléaire retenu.
- `nom_nuc_cpl` : nom de la compilation des réactions thermonucléaires :
    - `Adelb` : Compilation Adelberger et al. (1998).
    - `Cau-Fow` : Compilation de Caughlan & Fowler (1988).
    - `NACRE` : Compilation de NACRE (Angulo et al., 1999).
    - `NACRE+` : Compilation de NACRE (Angulo et al., 1999) avec correction du 3 $\alpha$  selon Ogata et al. (2009).
    - `NACRU` : Compilation de NACRE actualisée.
  - `MITLER=.TRUE.` : effet d'écran selon Mitler (1997).

## 4.15 Arguments de la NAMELIST NL\_ATM

- `NOM_ATM` : nom de la routine de restitution de l'atmosphère :
  - `lim_atm` : restitution de l'atmosphère à partir d'une loi  $T(\tau)$ ,
  - `lim_gong1` : restitution simplifiée de l'atmosphère, cas de GONG1,
  - `lim_tau1` : restitution simplifiée de l'atmosphère, atmosphère mono-couche.
- `NOM_TDETAU` : nom de la loi  $T(\tau)$  utilisée pour la restitution de l'atmosphère :
  - `edding` : loi  $T(\tau)$  totalement radiative d'Eddington.
  - `hopf` : loi  $T(\tau)$  totalement radiative de Hopf.
  - `hsra` : loi  $T(\tau)$  empirique, non totalement radiative de HSRA.
  - `K5750` : loi  $T(\tau)$  non totalement radiative, dérivée du modèle d'atmosphère solaire de Kurucz,  $T_{\text{eff}} = 5750$  K.
  - `K5777` : loi  $T(\tau)$  non totalement radiative, dérivée du modèle d'atmosphère solaire de Kurucz,  $T_{\text{eff}} = 5777$  K.
  - `MARCS_Z+0.00` : lois  $T(\tau)$  non totalement radiatives, dérivées des modèles d'atmosphère de MARCS. Elles couvrent l'intervalle  $3000\text{K} \leq T_{\text{eff}} \leq 8000\text{K}$ . Des tables<sup>12</sup> pour les métallicités  $[Z/X]$  :
    - 3.00, -2.00, -1.50, -0.75, -0.50, -0.25, +0.00, +0.25, +0.75, +1.00
 et des gravités ( $\log_{10} g$ ) recouvrant l'intervalle  $[+3.0, +5.0]$  avec

12. Codifiées `MARCS_Zsx.yy`, Exemple : `MARCS_Z-2.00`.

un pas de +0.5 sont disponibles sur simple demande auprès de Bernard.Pichon@oca.eu.

Une table supplémentaire : `MARCS.Z+0.00_NEW`, calculée avec  $[Z/X] = 0.0$ , pour les conditions conditions de faible gravité et de forte température effective (atmosphères d'AGB) est aussi disponible.

- `piau1`, `piau3` : lois  $T(\tau)$  non totalement radiatives, dérivées de modèles d'atmosphère de Kurucz (ATLAS12), de composition solaire As05(`piau1`) ou As09(`piau3`), popI. Pour la cohérence, utiliser la routine de convection Canuto Goldman Mazitelli (`conv_cgm`,  $\alpha = 0.5$ ) ApJ 473, 550, 1996. Les tables nécessaires à l'exploitation sont disponibles sur simple demande auprès de Laurent.Piau@cea.fr, elles sont placées dans le sous-directory `SUN_STAR_DATA/PIAU`.

Domaines couverts Teff / logg :

Teff	logg	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
4000		+	+	+	+	+	+	+	+	+	+	+
		+	+	+	+	+	+	+	+	+	+	+
		+	+	+	+	+	+	+	+	+	+	+
6400		+	+	+	+	+	+	+	+	+	+	+

- `roger+00` : loi  $T(\tau)$  non totalement radiative, dérivée de modèles d'atmosphère de Kurucz, de métallicité  $[Fe/H] = 0.0$ . Pour l'exploitation
- `roger+02` : loi  $T(\tau)$  non totalement radiative, dérivée de modèles d'atmosphère de Kurucz, de métallicité  $[Fe/H] = +0.2$ .
- `roger-05` : loi  $T(\tau)$  non totalement radiative, dérivée de modèles d'atmosphère de Kurucz, de métallicité  $[Fe/H] = -0.5$ .
- `roger-10a` : loi  $T(\tau)$  non totalement radiative, dérivée de modèles d'atmosphère de Kurucz, de métallicité  $[Fe/H] = -1.0$  avec éléments  $\alpha$  "enhanced".
- `TAU_MAX=10.d0` : profondeur optique du raccord avec l'enveloppe,
- `LIM_RO=.TRUE.` : condition limite externe en densité, sinon en pression.

## 4.16 Exploitation sous UNIX & LINUX

Sous UNIX/LINUX, il est conseillé<sup>13</sup> de créer et de laisser le module exécutable `cesam2k.out` dans le sous-directory `SOURCE`. Pour l'exploitation, utiliser un sous-directory propre à chaque application dans lequel on disposera les fichiers de données (`mon_modele.don`), les fichiers d'initialisation tels que `m020.zams`, et, si besoin, les fichiers de réglage et de personnalisation.

Par exemple, pour calculer des modèles de  $\xi$  Hya ( $3M_{\odot}$ ) de la ZAMS jusqu'à 400 My, saisir `create_evol_space KSI_HYA`. Le sous-directory `KSI_HYA` est créé. Il contient tous les éléments nécessaires à l'exploitation de `CESAM2k` ainsi que des README d'explication. Une autre possibilité est de :

- Créer un sous-directory `KSI_HYA`,
- Y copier les fichiers `m020.zams`<sup>14</sup> et `mon_modele.don` du sous-directory `EXPLOIT`.

Puis :

13. Voir aussi § 4.2 (Page 29).

14. Utiliser des fichiers d'initialisation ASCII ou binaire de  $3M_{\odot}$  si on en dispose.

- Renommer le fichier de données `mon_modele.don` -- > `ksi_hya.don`.
- Adapter la physique et les paramètres de ce fichier au problème de  $\xi$  Hya.

Lancer ensuite deux calculs successifs d'un modèle préliminaire de ZAMS avec (`AGEMAX = 0.d0`) adapté aux paramètres et à la physique adoptés :

1. le premier avec une précision réduite (`PRECISION='NP'`) et en utilisant le modèle d'initialisation en ASCII,
2. et le second avec la précision requise par exemple (`PRECISION='AV'`) en utilisant le modèle binaire `ksi_hya_B.hom` obtenu à l'étape précédente.

Conserver le fichier permanent `ksi_hya_B.hom` obtenu à l'issue de cette dernière étape dans le fichier `ksi_hya_ini_B.hom`. C'est avec ce dernier que pourront être initialisés les calculs des modèles d'exploitation.

Dans le fichier `ksi_hya.don` indiquer l'âge à atteindre : (`AGEMAX = 400.d0`) et lancer l'exécution : `cesam2k.out` en l'initialisant avec `ksi_hya_ini_B.hom`.

Pour faire par la suite des modèles en changeant un ou plusieurs paramètres (âge, overshoot, *etc...*), il suffit d'adapter le fichier `ksi_hya.don` à chaque cas particulier.

#### 4.16.1 Fichiers ASCII et binaires d'initialisation

Une évolution peut être initialisée par un modèle de pré-séquence principale ou par un modèle de main-séquence **homogène**. Ces modèles sont sous forme de fichiers ASCII ou binaires ; ces derniers étant des sous-produits d'évolutions ultérieures. Des fichiers ASCII sont disposés dans le sous-directory `EXPLOIT` et sont implémentés dans le directory de travail si la procédure `create_evol_space` est utilisée, *cf.* § 1.6 (Page 6). L'initialisation PMS permet de faire évoluer la composition chimique dès que le réseau nucléaire est activé ( $T \sim 1MK$ ). A l'âge de la ZAMS, époque où la production d'énergie d'origine nucléaire égalise celle d'origine gravitationnelle, la composition chimique n'est plus homogène alors qu'elle l'est dans le modèle d'initialisation de ZAMS homogène qui, de ce fait, n'a pas de sens physique en soi. Ces conditions non physiques étant relaxées après quelques pas temporels, typiquement une vingtaine. Les fichiers binaires permanents d'extensions `.pms` et `.zams` créés lors d'initialisations ou d'évolutions avec PMS peuvent servir à des initialisations ou à des reprises ultérieures ; il en est de même des fichiers binaires permanents d'extension `.hom` créés à l'issue d'initialisations sur la ZAMS. De façon générale ces fichiers binaires servent à optimiser les initialisations.

#### 4.16.2 Exemple d'évolution avec PMS

On se place dans le directory `TEST` créé au § 1.6 (Page 6).  
Saisir `cesam2k.out`

```
-----
CESAM speaks a bit of english if you include in the
working directory a file named 'langue' with the
statements : &NL_LANGUE
langue='english'
/
cf. aide_mem2k, chapter Personnalisation
```

```

-----
Pour arrêter : taper 0 puis RETURN
Pour poursuivre une évolution : taper 1 puis RETURN
Pour initialiser un modèle de ZAMS : taper 2 puis RETURN
Pour initialiser un modèle de PMS : taper 3 puis RETURN
3 <==
Le modèle initial de PMS est-il donné en binaire ? o/n
n <==
CESAM utilise un modèle initial de PMS en ASCII
entrer le nom du fichier ASCII du modèle PMS initial
Exemples: 2d-2.pms(Tc=0.2MK), 5d-4.pms(Tc=0.5MK)
8d-5.pms(Tc=1.0MK), 4d-2.pms (pour M*>10Msol)
5d-4.pms <==
CESAM utilise le modèle ASCII : 5d-4.pms
entrer l'identificateur du modèle :
Exemple : mon_modele
identificateur des fichiers du modèle : mon_modele <==

```

```

*****
MODELE DE STRUCTURE INTERNE calculé par CESAM2k version : V3.7.24
*****
Début du calcul le : 17 Décembre 2012 à 17h59
NAMELISTS du fichier: mon_modele.don

```

```
&NL_CESAM NOM_CHEMIN=~ /SUN_STAR_DATA/
```

```

.....
.....
.....

```

```

-----
calcul du modèle initial de PMS
entrer la constante de contraction c_iben, exemples :
pour Tc=100 000 K, entrer c_iben= 2.d-2
pour Tc=500 000 K, entrer c_iben= 5.d-4
pour Tc=1 000 000 K, entrer c_iben= 8.d-5
pour M*>10 Msol, entrer c_iben= 4.d-2
5.d-4 <==
----- Paramètres pour le modèle quasi-statique ----

```

saut de la fonction de répartition par couche = 5.000E-02  
 modification du nombre de couches si la constante de  
 répartition varie de plus de 5%

.....

.....

.....

Pré Séquence Principale premier modèle : C= 5.000E-04

Rext= 1.652E+01, Lext= 6.893E+01, Tc= 5.010E+05

ok? entrer o/n, pour arrêter entrer s

o <==

calcul d'un nouveau modèle avec 1.1 C

Intégration du modèle quasi-statique de PMS(début)

.....

.....

.....

Pré Séquence Principale second modèle : C= 5.500E-04

Rext= 1.714E+01, Lext= 7.322E+01, Tc= 4.838E+05, dt= 9.701E-04

ok? entrer o/n, pour arrêter entrer s

o <==

Pas de chute de planétoïdes

&NL\_DEVICE

H=6.00000000,DH=1.50000000,LD=8.50000000,DL=2.00000000,XLEFT=1.79999995,  
 YBOT=1.39999998,DEVICE=/xw

/

-----Restitution de l'atmosphère (début)-----

Lrac= 7.322E+01, Rrac= 1.714E+01, Teff= 4.082E+03, grav= 9.325E+01, tau\*=  
 6.454E-01

atmosphère iter. 1 err. max. 1.5E-05, var: Pt, couche 59, corr 1.0E+00

atmosphère iter. 2 err. max. 1.0E-09, var: Pt, couche 66, corr 1.0E+00

atmosphère iter. 3 err. max. 7.8E-16, var: M, couche 31, corr 1.0E+00

----- Restitution de l'atmosphère (fin) -----

\*\*\*\*\*

âge= 0.000E+00, LogTeff= 3.611E+00, LogL/Lsol= 1.865E+00, LogR/Rsol= 1.235E+00

Log g= 1.968E+00, Pc= 1.384E+11, Tc= 4.838E+05, Roc= 2.138E-03, Xc= 7.000E-01

ePP/eNUC= 0%, eCNO/eNUC= 0%, e3a+C+O/eNUC= 0%, eGRAV/eNUC= 100%, Yc=  
 2.799E-01

Var. rel. de masse : 0.000E+00, M\*= 1.000E+00Msol, modèle de pré-série

```

principale
Modèle hors la zone instable / outside the instability strip (P= 3.016E+00day)
*****

-- Intégration du modèle quasi-statique (début)----

Modèle n° 1, pas temporel estimé, dt= 9.701E-04

.....

.....

.....

Y.Lebreton OBSPM, P.Morel OCA. Décembre 2008, mixtures+opacités AGS05
Y.Lebreton OBSPM, P.Morel OCA. Novembre 2009, etat_irwin
T.Guillot, P.Morel OCA. Mai 2012, implantation de sorties *.csv

*****

Type <RETURN> for next page:

```

### 4.16.3 Limitation des sorties

Les informations qui apparaissent *on line* sur le moniteur permettent de suivre en détail le déroulement des calculs. Elles peuvent s'avérer non pertinentes pour une exploitation "batch" et encombrer inutilement. A cet effet, la variable logique `baratine` du module `mod_donnees`, lorsqu'elle est `.FALSE.`, permet de dérouter sur les fichiers `mon_modele_static`, `mon_modele_atmos`, `mon_modeleevol` la plupart des informations concernant le déroulement des calculs en ce qui concerne, respectivement, la résolution de l'équilibre quasi-statique, la restitution de l'atmosphère, l'évolution de la composition chimique et de la vitesse angulaire. Pour ce faire, il est nécessaire de coder `baratine=.FALSE.` dans le module `mod_donnees` et d'effectuer une recompilation partielle de ce module, cf. § 4.17.3 (Page 55). Quand la nature de cette limitation de sorties n'est qu'occasionnelle, il n'est pas nécessaire d'intervenir dans le module `mod_donnees` et de refaire une compilation ; il suffit de disposer, dans l'environnement du calcul, le fichier `blabla` décrit au § 5.13 (Page 72).

## 4.17 Contenu du package EXPLOIT

L'exploitation de `CESAM2k` nécessite l'utilisation d'outils astrophysique, comme par exemple le calcul de la valeur du rapport  $[Z/X]$  d'une mixture, ou encore de transformer une donnée photométrique et une parallaxe en luminosité. Des programmes fournissant ces données sont regroupés dans le package `EXPLOIT` avec des programmes d'exploitation des résultats des programmes de tracés par exemple ainsi que des shell-script de leur gestion comme la calibration solaire. Ces programmes

font appel aux bibliothèques du sous directory **SOURCE**. Les exécutables ont été créés globalement lors de l'exécution du Makefile `make_exp1` cf. § 1.5 (Page 5), le Makefile `Make_exe` permettant de les créer individuellement.

### 4.17.1 Outils astrophysiques

Ces outils sont les suivants, une description en est donnée dans leur sources respectives :

**calc\_nu\_reac** : calcul de taux de réactions et de leur dérivée/T

**calc\_alpha\_enhanced** : facteur d'amplification pour les opacités

**calc\_bv\_tycho** : B, V, Vh à partir des observations de Tycho

**calc\_corr\_bol** : corrections bolométriques et indices de couleur

**calc\_dnu0** : calcul de  $\delta\nu_0$  et de  $\nu_{\max}$

**calc\_dzdy** :  $\Delta Z/\Delta Y$  à partir de Y et [Fe/H] initial et final

**calc\_dtl** : calcul de  $\Delta \log_{10} T_{\text{eff}}$  et  $\Delta L$

**calc\_fesh** : calcul de [Fe/H]

**calc\_fesh\_zsx** :  $\text{Fe/H} \iff \text{Z/X}$

**calc\_glmt** : calcul de  $\log_{10} g(L, T_{\text{eff}}, M/M_{\odot})$  ou de  $L(\log_{10} g, T_{\text{eff}}, M/M_{\odot})$

**calc\_L2Mv** : calcul de Mv à partir de  $L/L_{\odot}$  et inverse

**calc\_lmbol** :  $L/L_{\odot} \iff M_{\text{bol}}$

**calc\_l\_r\_teff** :  $L/L_{\odot}, R/R_{\odot} \implies T_{\text{eff}}$ , ou  $L/L_{\odot}, T_{\text{eff}} \implies R/R_{\odot}$ , ou  $R/R_{\odot}, T_{\text{eff}} \implies L/L_{\odot}$

**calc\_l\_g\_teff\_m** :  $(T_{\text{eff}}, M/M_{\odot}, \log_{10} g) \implies L/L_{\odot}$  ou  $(T_{\text{eff}}, M/M_{\odot}, L/L_{\odot}) \implies \log_{10} g$

**calc\_lum\_teff** :  $(\pi, T_{\text{eff}}, \text{Ind.Coul.}) \implies L/L_{\odot}$

**calc\_mag\_abs** : calcul de la magnitude absolue

**calc\_masses** : pour une binaire, masses individuelles avec erreurs à partir de la somme et de la fraction de masse

**calc\_mbol\_lum** :  $M_{\text{bol}}$  et  $L/L_{\odot}$  à partir de V et du module de distance

**calc\_mod\_dis** : calcul du module de distance

**calc\_moyenne\_ponderee** : calcul de la moyenne pondérée et du sigma

**calc\_new\_x** : [Fe/H] initial et X pour avoir fesh0 a la fin d'une évolution

**calc\_rltgm** : Détermination de 2 quantités parmi  $T_{\text{eff}}, \log_{10} g, M/M_{\odot}, R/R_{\odot}, L/L_{\odot}$  à partir des 3 autres

**calc\_r\_theta** : diamètre +  $\pi$  en mas  $\implies R/R_{\odot}$

**calc\_som\_mag** : pour une binaire, calcul de la magnitude de A+B

**calc\_teff** :  $T_{\text{eff}}$  à partir de calibrations photométriques

**calc\_vsini** :  $R/R_{\odot}, V \sin i, i \implies$  période de rotation en jours, radians/sec et Hz

**calc\_XisH** : calcul des [Xi/H]

**calc\_XYZ** : X, Y  $\implies$  Z et Z/X ou X, Z  $\implies$  Y et Z/X ou Y, Z  $\implies$  X et Z/X

**calc\_z\_sur\_x** : calcul de  $[Z / X]$  pour différentes mixtures

### 4.17.2 Programmes d'exploitation

D'autres programmes du sous directory `EXPLOIT` permettent une exploitation des fichiers ASCII ou binaires créés par `CESAM2k` :

- `calib2k_pms` : Formation du fichier de données `mon_modele.don` pour calibration d'un modèle solaire avec PMS.
- `calib2k_zams` : Formation du fichier de données `mon_modele.don` pour calibration d'un modèle solaire initialisé sur la ZAMS.
- `des2k_ZC` : Dessin de l'évolution des zones convectives en fonction du temps.
- `des2k_abon` : Dessin des abondances en fonction de la masse ou du rayon.
- `des2k_abontc` : Dessin de l'évolution de X, Y, Z au centre, en fonction du temps.
- `des2k_abonts` : Dessin de l'évolution de X, Y, Z en surface, en fonction du temps.
- `des2k_bin`, `des2k_bin_tds` : Dessin des variables quasi-statiques à partir d'un fichier binaire.
- `des2k_diff_spl` : Dessin des différences entre deux modèles avec interpolation par B-splines.
- `des2k_grad` : Dessins des différents gradients d'un modèle.
- `des2k_hr` / `des2k_lr` : Dessin du diagramme HR à partir d'un fichier d'extension `.HR` / `.LR`.
- `des2k_opa` : Dessin des différences relatives entre deux tables d'opacité.
- `des2k_osc` : Dessin de 1, 2 ou 3 modèles en fonction du rayon, de la masse avec des possibilités de zoom.
- `des2k_rot` : Dessins séparés des variables de la diffusion du moment cinétique.
- `des2k_rot_ext` : Dessin du profil, en fonction du temps, de la vitesse angulaire de la couche externe en km/s ou en rad/sec.
- `des2k_vaiss` : Dessin du profil de la fréquence de Brunt-Väissälä.
- `fichier_vent` : Programme de construction d'un fichier de composition chimique du vent.

Dans chacun de ces programmes, des informations concernant leur utilisation sont données sous la forme de commentaires.

### 4.17.3 Shell-scripts d'exploitation

Des shell-scripts `bash` permettent l'exploitation<sup>15</sup> de `CESAM2k` essentiellement la calibration solaire et de l'évolution à partir de la ZAMS ou de la PMS.

- `calib_pms` : procédure de calibration d'un modèle solaire initialisé sur la PMS homogène.
- `calib_zams` : procédure de calibration d'un modèle solaire initialisé sur la ZAMS homogène.
- `create_evol_space` : création d'un espace de travail contenant les fichiers nécessaires à l'exploitation de `CESAM2k`, les fichiers de données et de réglage types et des modèles initiaux de PMS et de ZAMS.
- `evol2k_pms` : script pour une évolution initialisée sur la PMS

---

15. Les options de compilation sont celles du compilateur LINUX LF95, il convient de les adapter à celles du compilateur utilisé.

- `evol2k_zams` : script pour une évolution initialisée sur la ZAMS
- `evolsun_pms` : script pour une évolution solaire initialisée sur la PMS
- `evolsun_pms_list` : évolution d'une liste de modèles solaires initialisés sur la PMS
- `evolsun_zams` : script pour une évolution solaire initialisée sur la ZAMS
- `evolsun_zams_list` : évolution d'une liste de modèles solaire initialisés sur la ZAMS
- `Make_exe` : makefile pour compilation et création d'un exécutable
- `Make_expl` : makefile pour compilation et création des exécutables d'EXPLOIT
- `Make` : makefile pour compilation et création des exécutables de la SOURCE

#### 4.17.4 Données diverses

- `2d-2_pms` : fichier de modèle initial de PMS, avec constante de contraction  $c_{iben}=2.10^{-2}$ , soit  $T_c \sim 1.10^5$  K.
- `4d-2_pms` : fichier de modèle initial de PMS, avec constante de contraction  $c_{iben}=3.10^{-2}$ , soit  $T_c \sim 3.10^5$  K.
- `5d-4_pms` : fichier de modèle initial de PMS, avec constante de contraction  $c_{iben}=5.10^{-4}$ , soit  $T_c \sim 5.10^5$  K (combustion du deutérium).
- `8d-5_pms` :  $8.10^{-5}$  fichier de modèle initial de PMS, avec constante de contraction  $c_{iben}=\text{}$ , soit  $T_c \sim 1.10^6$  K (amorçage des réactions nucléaires du cycle PP).
- `blabla` : exemple de NAMELIST "NL\_BLABLA" pour la limitation des écritures lors d'une exécution.
- `device` : exemple de NAMELIST "NL\_device" permettant le cadrage des graphes "on line".
- `langue` : exemple de NAMELIST "NL\_LANGUE" définissant la langue à utiliser pour les commentaires produits en cours d'exécution, cf. § 5.1 (Page 60).
- `m010.zams` : fichier de modèle initial de ZAMS, de  $1.M_{\odot}$ .
- `m020.zams` : fichier de modèle initial de ZAMS, de  $2.M_{\odot}$ .
- `m050.zams` : fichier de modèle initial de ZAMS, de  $5.M_{\odot}$ .
- `m150.zams` : fichier de modèle initial de ZAMS, de  $15.M_{\odot}$ .
- `mixture` : exemple de NAMELIST "NL\_MIXTURE" pour définition d'une nouvelle mixture initiale, cf. § 5.2 (Page 60).
- `mon_modele_modif_mix` : exemple de NAMELIST "nl\_modif\_mix" de modification d'une mixture initiale, cf. § 5.4 (Page 62).
- `mon_modele.don` : exemple de fichier de données.
- `mon_modele.rap_iso` : exemple de NAMELIST "NL\_rap\_iso" de modification des rapports isotopiques initiaux, cf. § 5.3 (Page 61).
- `mon_modele.rg` : valeurs par défaut du fichier de réglages, cf. § 5.9 (Page 65).
- `mon_modele.vent` : exemple de de NAMELIST "NL\_VENT" de composition chimique du vent, cf. § 5.5 (Page 62).
- `mon_modele.zoom` : exemple de de NAMELIST "NL\_DES" pour adaptation du graphe on line.
- `opteos` : exemple de la NAMELIST "nl\_oes" permettant d'adapter les arguments de l'équation d'état Irwin, cf. § ?? (Page ??).

- `planet` : exemple de de NAMELIST "NL\_PLANET" de notification des caractéristiques de planétoïdes, *cf.* § 5.6 (Page 63).
- `vson_rho_sun_data` : profils de la vitesse du son et de la densité solaires.



# Chapitre 5

## Personnalisation

*Le plaisir de la connaissance est à nul autre pareil.*

Bien que l'exécutable soit construit un fois pour toutes, il est possible d'adapter de façon externe un certain nombre de paramètres. Ces adaptations s'effectuent, à la demande, par l'intermédiaire de fichiers *ad hoc* à disposer dans le sous-directory où s'effectue le calcul. Ne seront concernés que les modèles calculés dans cet environnement. De façon générale, dans le directory concerné, une personnalisation peut affecter, soit un modèle particulier, soit tous les modèles à calculer dans ce directory. CESAM2k recherche d'abord si une personnalisation existe pour le modèle à calculer puis, à défaut, si elle existe pour tous les modèles du directory. Des exemples de tels fichiers de personnalisation se trouvent dans le sous-directory **EXPLOIT** de la distribution. On peut ainsi :

- Avoir des commentaires d'interactivité en une langue différente du français<sup>1</sup>.
- Utiliser une mixture initiale différente de celles implémentées.
- Modifier les rapports d'abondances d'une mixture.
- Redéfinir les rapports isotopiques.
- Créer un fichier de sortie en ASCII contenant des quantités en nombre et ordre différents de ce qui est implémenté.
- Utiliser des réglages personnalisés.
- Dessiner des boîtes correspondant à des objectifs à atteindre, et adapter les échelles du diagramme HR tracé "on line".
- Adapter les dimensions des cadres et le nom du device à utiliser.
- Détourner sur des fichiers en ASCII une partie des informations apparaissant 'on line' sur le moniteur.

**Par sécurité**, ainsi qu'il est explicité pour chaque cas particulier, certains des fichiers de personnalisation concernant la composition chimique ne seront pris en compte que si le paramètre `modif_chim` de la NAMELIST `n1_chim` du fichier de données est `.TRUE.` cf. § 4.8 (Page 40).

---

1. Seulement en anglais dans la présente version.

## 5.1 Langue

Une alternative pour avoir des commentaires d'interactivité en une langue différente du français :

1. Si on désire **toujours** avoir des commentaires en une langue différente du français, **avant** la compilation, dans la routine `cesam` du sous-directory `SOURCE`, commenter ou supprimer les instructions (vers les lignes 270-278) :

```

    INQUIRE(file='langue',exist=ok)
    IF(ok)THEN
      OPEN(unit=30,form='formatted',status='old',file='langue')
      READ(30,nl_langue) ; CLOSE(UNIT=30)
    ELSE
      WRITE(*,30)
30   FORMAT(/,'Sometimes CESAM can speak english,','/,
1   'see aide_mem2k, chapter personnalisation ')
      langue='français'
    ENDIF

```

et décommenter l'instruction de la ligne suivante (279) :

```
c   langue='english'
```

Les commentaires en anglais seront désormais utilisés pour toute exploitation.

2. Si on désire n'avoir qu'occasionnellement les commentaires en anglais, introduire dans l'environnement d'exploitation le fichier de nom `langue` que l'on trouvera dans le sous directory `EXPLOIT`, ou encore l'y créer, *cf.* §4.17.4 (Page 56) :

```

&NL_LANGUE
langue='english'
/

```

Dans toutes les exploitations effectuées dans cet environnement les commentaires apparaitront en anglais.

Par défaut, *i.e.* en l'absence de l'une des dispositions précédentes, les commentaires seront en français.

**PB** Dans la version actuelle de `CESAM2k`, seulement une partie des principaux commentaires a été traduite en anglais.

**PB** **Call for man power** : il est facile de transposer dans une autre langue les algorithmes créés pour l'anglais ou encore compléter et/ou améliorer la syntaxe des ces derniers, les volontaires sont bienvenus.

## 5.2 Mixture

Pour utiliser une mixture initiale différente<sup>2</sup> de celles implémentées, coder `nom_abon='mixture'` dans le fichier de données et, dans l'environnement, placer

---

2. Voir aussi *cf.* §6.2 (Page 74).

un fichier de nom `mixture` contenant la mixture désirée en Dex. *CESAM2k refusera* toute mixture dont la normalisation est différente de H=12. L'exemple suivant se trouve dans le sous-directory `EXPLOIT`, cf. § 4.17.4 (Page 56) :

```
&NL_MIXTURE
ab(1)=12.00d0 , ab(2)=10.d0 , ab(3)=2.69d0 , ab(4)=2.15d0 , ab(5)=2.60d0
ab(6)=4.55d0 , ab(7)=3.97d0 , ab(8)=4.87d0 , ab(9)=1.56d0 , ab(10)=4.08d0
ab(11)=2.33d0 , ab(12)=3.58d0 , ab(13)=2.47d0 , ab(14)=3.55d0 , ab(15)=1.45d0
ab(16)=3.21d0 , ab(17)=1.5d0 , ab(18)=2.52d0 , ab(19)=1.12d0 , ab(20)=2.36d0
ab(21)=-1.17d0 , ab(22)=1.02d0 , ab(23)=0.d0 , ab(24)=1.67d0 , ab(25)=1.39d0
ab(26)=3.5d0 , ab(27)=0.92d0 , ab(28)=2.25d0
/
```

On trouve, dans la routine `abon_ini`, l'ordre et l'identification des éléments par leurs indices, en fait, celle des opacités `OPAL`. Dès lors, *tous* les modèles calculés dans ce directory, s'ils utilisent l'option `nom_abon='mixture'`, auront des abondances initiales conformes à celles du fichier `mixture`. Une autre possibilité permet de personnaliser davantage; elle consiste à donner au fichier `mixture` le nom générique du modèle avec l'extension `.mix`, Ex : `mon_modele.mix`. Dès lors, *seul* le modèle `mon_modele` aura la composition chimique initiale désirée. Avec l'option `mixture`, *CESAM2k* cherche *en priorité* à utiliser le fichier `mon_modele.mix`; en son absence, c'est le fichier `mixture` qui est utilisé; le calcul n'est effectué que si l'un de ces deux fichiers existe dans l'environnement.

Cette option nécessite de coder `MODIF_CHIM=.TRUE.`, cf. § 4.8 (Page 40).

**PB** Avec l'option `nom_abon='mixture'`, les abondances relatives des métaux devraient, en principe, correspondre à celles des opacités, au moins, faute de mieux, pouvoir être prises en charge par la routine d'opacité utilisée.

## 5.3 Rapports isotopiques

Pour utiliser des rapports isotopiques différents de ceux pris par défaut par *CESAM2k*, placer dans l'environnement du calcul, un fichier de nom `mon_modele.rap_iso` dans lequel seront précisés les rapports isotopiques à utiliser. Si *tous* les modèles de l'environnement doivent être calculés avec des rapports modifiés, donner au fichier précédent le nom de `rap_iso`, cf. § 4.17.4 (Page 56). Exemple :

```
&NL_RAP_ISO
be7sbe9=1.d-25,
be7sz=1.d-29,
c13sc12=1.10d-2,
h2sh1=3.01d-5,
he3she4=1.1d-4,
he3she4z=4.185d-4,
li6sli7=7.5d-2,
mg25smg24=0.0125,
mg26smg25=0.013,
ne22sne20=6.79d-2,
n15sn14=0.366d-2,
o17so16=0.038d-2
o18so16=0.008d-2
/
```

Les noms utilisés pour les variables suggèrent les notations des rapports isotopiques `be7sbe9` : rapport isotopique  $^7\text{Be}/^9\text{Be}$ . `he3she4z` est le rapport isotopique  $^3\text{He}/^4\text{He}$  sur la ZAMS lorsque le deutérium initial a été transformé en  $^3\text{He}$ , `be7sz` est l'abondance, quasi nulle, du  $^7\text{Be}$  initial dans Z.

*Par sécurité*, ce fichier ne sera pris en compte, que si le paramètre `modif_chim` de la NAMELIST `n1_chim` du fichier de données est `.TRUE.` cf. § 4.8 (Page 40).

## 5.4 Personnalisation des rapports d'abondances

La mixture une fois définie, par exemple `solaire_gn` (Grevesse & Noels, 1993), il est possible d'y modifier les abondances. Pour ce faire, créer dans l'environnement du calcul, un fichier de nom `mon_modele.modif_mix`, dans lequel seront indiquées les modifications en DeX à apporter. Si tous les modèles de l'environnement doivent être calculés avec ces modifications, donner au fichier précédent le nom de `modif_mix`. Exemple se trouvant dans le directory EXPLOIT, cf. § 4.17.4 (Page 56) :

```
&n1_modif_mix
add_Li=2.d0 ; add_Be=0.d0 ; add_B=0.d0
add_C=-1.0d0 ; add_N=0.d0 ; add_O=-0.3d0 ; add_F=0.d0 ; add_Ne=0.d0
add_Na=0.d0 ; add_Mg=0.d0 ; add_Al=0.d0 ; add_Si=0.d0 ; add_P=0.d0
add_S=0.d0 ; add_Cl=0.d0 ; add_Ar=0.d0 ; add_K=0.d0 ; add_Ca=0.d0
add_Sc=0.d0 ; add_Ti=0.d0 ; add_V=0.d0 ; add_Cr=0.d0 ; add_Mn=0.d0
add_Fe=0.d0 ; add_Co=0.d0 ; add_Ni=0.d0 ; add_Z=-1.d0
/
```

Les notations suggèrent les quantités à modifier ; ainsi `add_Li=2.d0` est le nombre de DeX à ajouter à l'abondance du lithium. `add_Z=-1.d0` signifie que les abondances *de tous les métaux* seront diminuées d'un DeX. *Par sécurité*, ce fichier ne sera pris en compte que si le paramètre `modif_chim` de la NAMELIST `n1_chim` du fichier de données est `.TRUE.` cf. § 4.4 (Page 31).

Si on a codé `GARDE_XISH=.FALSE.` dans le fichier de données cf. § 4.8.1 (Page 41), le paramètre `add_Z`, qui agit sur les métaux dans leur ensemble, n'a pas d'incidence dans la détermination des rapports métal/Z. A l'opposé, une valeur non nulle de `add_C`, par exemple, aura une incidence sur les rapports métal/Z.

## 5.5 Personnalisation de la composition chimique du vent.

Dans le cas standard, avec la donnée `mdot` non nulle, cf. § 4.4 (Page 31), la composition chimique de la masse perdue ou gagnée par le vent est celle de la couche la plus externe du modèle. Il est possible de personnaliser cette composition chimique, afin qu'elle diffère de celle de la couche externe. Pour ce faire, placer dans l'environnement du calcul, un fichier de nom `mon_modele.vent` dans lequel seront indiquées les fractions de masse de chaque élément chimique dans l'apport ou la perte de masse. Si tous les modèles de l'environnement doivent être calculés avec ces modifications, donner au fichier précédent le nom de `vent`. Un exemple se trouve dans le sous-directory EXPLOIT, cf. § 4.17.4 (Page 56) :

```
&n1_vent
vt_H=0.7347, vt_He=0.2483, vt_Li=1.033e-08, vt_Be=1.72816e-10,
vt_B=4.859e-09, vt_C=2.899e-03, vt_N=8.493e-04, vt_O=7.885e-03,
```

```
vt_F=4.182e-07,vt_Ne=1.768e-03,vt_Na=3.501e-05,vt_Mg=6.736e-04,
vt_Al=6.078e-05,vt_Si=7.434e-04,vt_P=8.198e-06,vt_S=3.704e-04,
vt_Cl=4.924e-06,vt_Ar=7.315e-05,vt_K=3.845e-06,vt_Ca=6.541e-05,
vt_Sc=4.126e-08,vt_Ti=3.041e-06,vt_V=3.888e-07,vt_Cr=1.856e-05,
vt_Mn=1.357e-05,vt_Fe=1.287e-03,vt_Co=3.492e-06,vt_Ni=7.6092e-05
/
```

Les notations sont suggestives, ainsi `vt_Li=1.033e-08` est la fraction masse du lithium dans le vent. Une renormalisation étant effectuée lors de l'utilisation, il n'est pas nécessaire de normaliser les fractions de masse figurant dans ce fichier, certaines peuvent être nulles et même négatives. Suivant le signe de `MDOT`, la quantité de matière, apportée ou soustraite, est supposée provenir de la zone convective externe qui existe toujours. Le fichier `vent` peut être construit<sup>3</sup> par le programme `fichier_vent` du sous-directory `EXPLOIT`.

*Par sécurité*, ce fichier ne sera pris en compte, que si le paramètre `modif_chim` de la `NAMELIST n1_fdon` du fichier de données est `.TRUE.` cf. § 4.8 (Page 40).

## 5.6 Chute de planètes

Au cours d'une évolution `CESAM2k` permet de simuler une chute de planètes sur un intervalle de temps limité. Pour ce faire, placer dans l'environnement du calcul, un fichier de nom `mon_modele.planet` dans lequel seront indiquées les fractions de masse de chaque élément de la composition chimique des planètes, le nombre total de planètes qui sera reçu par l'étoile, l'intervalle de temps concerné par la chute et le profil de l'intensité de la chute. Si tous les modèles de l'environnement doivent être calculés avec ces modifications, donner au fichier précédent le nom de `planet`. Un exemple se trouve dans le sous-directory `EXPLOIT`, cf. § 4.17.4 (Page 56) :

```
&n1_planet
vt_H=0.7347,vt_He=0.2483,vt_Li=1.033e-08,vt_Be=1.72816e-10,
vt_B=4.859e-09,vt_C=2.899e-03,vt_N=8.493e-04,vt_O=7.885e-03,
vt_F=4.182e-07,vt_Ne=1.768e-03,vt_Na=3.501e-05,vt_Mg=6.736e-04,
vt_Al=6.078e-05,vt_Si=7.434e-04,vt_P=8.198e-06,vt_S=3.704e-04,
vt_Cl=4.924e-06,vt_Ar=7.315e-05,vt_K=3.845e-06,vt_Ca=6.541e-05,
vt_Sc=4.126e-08,vt_Ti=3.041e-06,vt_V=3.888e-07,vt_Cr=1.856e-05,
vt_Mn=1.357e-05,vt_Fe=1.287e-03,vt_Co=3.492e-06,vt_Ni=7.6092e-05,
ypl=0.d0,zpl=0.9d0,
n_planet=10,profil='rectangle',age_deb=5.d0,age_fin=10.d0,
r_giration=1.d0,t_giration=1.d0
/
```

Les notations sont suggestives. Avec `vt_H` et `vt_He` *non nuls*, des valeurs non nulles de `ypl` et/ou `zpl` permettent de modifier les fractions de masse de l'hydrogène  $X$ , de l'hélium  $Y$  et des métaux  $Z$  des planètes. Avec le fichier `planet` précédent, l'abondance en masse de l'hélium sera  $Y = 0$ , celle des métaux  $Z = 0.9$  et celle d'hydrogène  $X = 1.0 - 0.0 - 0.9 = 0.1$ . `age_deb` et `age_fin` sont respectivement les âges, en million d'années, du début et de la fin de la chute. `r_giration` est le rayon de giration en UA et `t_giration` le temps de giration, en années. `CESAM2k` estime la vitesse angulaire des planètes en admettant qu'ils décrivent un arc de  $\pi/2$  pendant

3. La composition chimique de l'exemple est celle de la mixture météoritique de Grevesse & Sauval (1998). Par ailleurs, il n'est pas nécessaire, ici, de respecter la 'case'.

un temps de giration. Une valeur négative du temps de giration, correspond à un apport négatif de moment cinétique, i.e. rétrograde. Avec une valeur nulle du temps de giration il n'y a pas d'apport de moment cinétique.

Dans la routine `planetoides.f`, `CESAM2k` propose 4 profils :

1. `rectangle` : L'intensité de la chute est constant.
2. `triangle` : Le profil de l'intensité de la chute est triangulaire, le maximum étant centré au milieu de l'intervalle de temps.
3. `parabole` : Le profil de l'intensité de la chute est parabolique, le maximum étant centré au milieu de l'intervalle de temps.
4. `gauss` : Le profil de l'intensité de la chute est une gaussienne limitée à l'intervalle de temps. Le maximum est centré au milieu de l'intervalle de temps. L'écart type est arbitrairement pris égal au tiers de l'intervalle de temps.

Cette option nécessite de coder `MODIF_CHIM=.TRUE.`, cf. § 4.8 (Page 40).

## 5.7 Personnalisation des paramètres de `free-eos`

Il est possible de personnaliser les paramètres de l'équation d'état `free-eos` en plaçant dans l'environnement un fichier de nom `opteos` avec les valeurs désirées pour les paramètres, cf. § 4.17.4 (Page 56) :

```
&nl_eos
ifoption=2,
ifmodified=1,
ifion=-1
/
```

L'implantation de cette équation d'état est décrite au § 3.2.5 (Page 23) et son exploitation dans `CESAM2k` au § 4.12 (Page 45). L'utilisation des paramètres est décrite dans les READMEs de la source <http://freeeos.sourceforge.net>, `free_eos-2.2.1.tar.gz`. Quelques indications sont données dans la routine `etat_irwin.f` du sous-directory `SOURCE` de `CESAM2k`.

## 5.8 Personnalisation du fichier Ascii

Pour créer un fichier de sortie en ASCII personnalisé, coder `nom_output='ascii'` dans le fichier de données et, dans l'environnement, placer un fichier de nom `sortie_ascii`<sup>4</sup> dans lequel sont définis les paramètres créant la structure souhaitée. Le codage, *délicat*, est décrit dans la routine `output.f`, les indices des variables sont identifiés dans la routine `ascii.f` du sous-directory `SOURCE`. Dans l'exemple :

```
12 3
1 2 3 4 5 6 9 10 11 12 13 15
1 2 3
.FALSE.
```

---

4. Dont un exemple se trouve dans le sous-directory `EXPLOIT`.

.TRUE.

.TRUE.

Fichier pour sortie ascii:

les quantités ont les significations suivantes :

- 12 3 → on écrira 12 variables globales et 3 variables locales,
- 1 2 3 4 5 6 9 10 11 12 13 15 → indices des 12 variables globales (mstar\*msol, rtot\*rsol, ltot\*lsol, z0, x0, alpha, X dans ZC, Y dans ZC, d2p, d2ro, age, vsal, w\_rot initial).
- 1 2 3 → indices des 3 variables locales : Rayon, log(M/Mtot), Température).
- .FALSE. → la masse sera écrite en fraction de la masse totale et non en DeX.
- .TRUE. → tabulation du centre à la surface.
- .TRUE. → on ajoute la composition chimique locale à la suite des variables locales.
- Fichier pour sortie ascii : texte d'entête du fichier de sortie.

Le nom générique du fichier de données est repris pour former celui du fichier de sortie en ASCII : avec `mon_modele.don` le nom du fichier de sortie est `mon_modele-ascii`. Avec le fichier cité comme exemple, on a obtenu :

```
Fichier pour sortie ascii: mon_modele-ascii
CESAM2k version 0.0.0.0 lagr colloc 1 2 np no diffus, 06 Juillet 2003 17h46
Physique utilisée: etat_eff, opa_yveline_lisse, conv_jmj, ppcno9, NACRE
solaire_gn, lim_atm, hopf, perte_ext, diffm_mp, diff_t_nu, ctes_94
 10 H1 He3 He4 C12 C13 N14 N15 O16 O17 Si28
    541      12      13      10      -1
1.989190000000E+33 6.010741880620E+10 2.478601588572E+33 2.000000000000E-02 7.000000000000E-01
1.800000000000E+00 7.000000000000E-01 2.800000000000E-01-4.241618755712E+01-2.519592625187E+01
0.000000000000E+00 0.000000000000E+00
0.000000000000E+00 0.000000000000E+00 1.359013041915E+07 7.000000000000E-01 8.826927693868E-05
2.799117307231E-01 3.425410508692E-03 4.128413464627E-05 1.059170894932E-03 4.168107978225E-06
9.641655715570E-03 3.903146720978E-06 5.824407491460E-03
1.302042603905E+09 1.175206179724E-08 1.353639070045E+07 7.000000000000E-01 8.826927693868E-05
2.799117307231E-01 3.425410508692E-03 4.128413464627E-05 1.059170894932E-03 4.168107978225E-06
9.641655715570E-03 3.903146720978E-06 5.824407491460E-03
.....
6.014328252162E+10 1.000000000164E+00 4.543950048106E+03 7.000000000000E-01 8.826927693868E-05
2.799117307231E-01 3.425410508692E-03 4.128413464627E-05 1.059170894932E-03 4.168107978225E-06
9.641655715570E-03 3.903146720978E-06 5.824407491460E-03
6.014636429604E+10 0.000000000000E+00 4.543482782113E+03 7.000000000000E-01 8.826927693868E-05
2.799117307231E-01 3.425410508692E-03 4.128413464627E-05 1.059170894932E-03 4.168107978225E-06
9.641655715570E-03 3.903146720978E-06 5.824407491460E-03
```

Comme pour les abondances initiales *cf.* § 5.2 (Page 60), il est possible de ne créer ce fichier de sortie personnalisé, que pour un modèle particulier, en donnant au fichier de sortie `sortie_ascii` le nom du modèle avec l'extension `.ascii`, exemple : `mon_modele.ascii`. S'il existe dans l'environnement, ce seront les paramètres de ce dernier fichier qui seront pris en compte pour *le* modèle de nom `mon_modele`. En son absence, CESAM2k cherchera à utiliser ceux du fichier `sortie_ascii`, à défaut de ce dernier, il n'y aura pas de sortie ASCII.

## 5.9 Réglages

**PB** Les arguments du fichier de réglage sont indiqués dans le fichier `arguments_des_reglages.f` des sous directories `SOURCE` et `EXPLOIT`.

Les réglages, prédéfinis dans la routine `cesam.f` en fonction du paramètre `PRECISION` du fichier de données `mon_modele.don`, peuvent être personnalisés en codant : `PRECISION='rg'`, et en disposant dans l'environnement un fichier de nom : `reglages`<sup>5</sup> comportant les réglages à utiliser. **Ces valeurs prévalent sur celles prédéfinies.** Exemple d'un tel fichier `reglages` :

```
&NL_RLG
m_qs=1,
m_ch=2,
m_rot=2,
m_tds=2,
m_ptm=2,
ordre=4,
precic=1.d-4,
precix=1.d-3,
precit=0.15d0,
psi0=0.1d0,
loc_zc=1.d-3,
dtmax=100.d0,
dtmin=1.d-10,
dt0=10.d0,
d_lum=0.1d0,
d_press=0.1d0,
d_ray=0.1d0,
d_temp=0.1d0,
age_car=0.1d0,
ini0=3,
n_atm=75,
kipp=.TRUE.,
en_m23=.FALSE.,
ctel=0.d0,
ctep=-1.d0,
ctem=15.d0,
cter=0.d0,
ctet=0.d0,
dn_fixe=0.05d0,
dpsi=0.05d0,
mu_saha=.TRUE.,
ajuste=.FALSE.,
lisse=.FALSE.,
npt_lisse=1,
q0=0.0d0,
l0=0,
new_bv=.TRUE.,
fcv=.FALSE.,
des_instb=.FALSE.,
```

---

5. Dont un exemple se trouve dans le sous-directory `EXPLOIT`.

```

no_discon=.FALSE.,
n_min_ZC=5,
l_fac=.TRUE.,
d_zc=5
/

```

Les significations des paramètres sont les suivantes :

**m\_qs=1** : Ordre des B-splines pour les variables quasi-statiques.

**m\_ch=2** : Ordre des B-splines pour l'interpolation de la composition chimique.

**m\_rot=2** : Ordre des B-splines pour l'interpolation du moment cinétique.

**m\_tds=2** : Ordre des B-splines pour interpolation de l'énergie gravipnique.

**m\_ptm=2** : Ordre des B-splines pour interpolation de la perte de masse.

**ordre=4** : Ordre du schéma d'intégration des réactions nucléaires. avec **rkimps**.

**precic=1.d-4** : Précision sur l'intégration temporelle de la composition chimique (réactions nucléaires)

**precix=1.d-3** : Précision sur les itérations Newton-Raphson des intégrations spatiales.

**precit=0.15d0** : Variation maximale/masse pour l'intégration temporelle de la composition chimique.

**psi0=0.1d0** : Constante de répartition à assurer.

**loc\_zc=5.d-3** : Précision de la localisation des limites zones radiatives / zones convectives.

**dtmax=100.d0** : Pas temporel maximum.

**dtmin=1.d-10** : Pas temporel minimum.

**dt0=10.d0** : Pas temporel initial.

**d\_lum=0.1d0** : Variation temporelle maximale en luminosité.

**d\_press=0.1d0** : Variation temporelle maximale en pression.

**d\_ray=0.1d0** : Variation temporelle maximale en rayon.

**d\_temp=0.1d0** : Variation temporelle maximale en température.

**age\_car=0.1d0** : Estimation du temps nécessaire à l'amortissement des conditions initiales physiquement incohérentes.

**ini0=3** : Nombre d'itérations Newton-Raphson avec réestimation de la composition chimique, de la vitesse angulaire et des limites zones radiatives / zones convectives.

**n\_atm=75** : Nombre de couches dans l'atmosphère restituée.

**kipp=.TRUE.** : Utilisation de l'approximation de Kippenhahn.

**en\_m23=.TRUE.** : Variables utilisées :  $\ln P$ .

**ctel=0.d0** : Facteur de répartition en luminosité.

**ctep=-1.d0** : Facteur de répartition en pression.

**ctem=15.d0** : Facteur de répartition. en masse.

- cter=0.d0** : Facteur de répartition. en rayon.
- ctet=0.d0** : Facteur de répartition. en température.
- dn\_fixe=0.05d0** : Taux limite de variation du nombre de couches nécessitant un réajustement de la grille fixe pour l'interpolation de la composition chimique.
- dpsi=0.05d0** : Taux limite de variation de la fonction de répartition nécessitant une modification du nombre de couches.
- mu\_saha=.TRUE.** : on supposera le milieu totalement ionisé pour le calcul du poids moléculaire moyen.
- ajuste=.FALSE.** : ajustement de T, X central, He core, Rext (sauf Teff).
- lisse=.FALSE.** : la composition chimique est lissée par diffusion turbulente.
- npt\_lisse=1** : nombre de points à droite et à gauche pour le lissage des discontinuités de composition chimique.
- q0=0.1d0** : Dans les fichiers ASCII de sortie, un point est placé à  $q0 > 0$  fois l'espacement entre les deux premiers points.
- l0=4** : Dans les fichiers ASCII de sortie, ajout de  $l0+1$  points entre `lim_ZC` et `lim_ZC+1`, de  $l0$  points entre `lim_ZC+1` et `lim_ZC+2`..... 1 point entre `lim_ZC+l0` et `lim_ZC+l0+1` et par symétrie entre `lim_ZC` et `lim_ZC-l0`.
- new\_bv=.TRUE.** : Calcul de la fréquence de Brunt-Väissälä utilisant  $\varphi = \frac{\partial \ln \rho}{\partial \ln \mu}$ .
- fcv=.FALSE.** : on ne forcera pas la convergence.
- des\_instb=.FALSE.** : la zone d'instabilité n'apparaîtra pas dans le graphe.
- no\_discon=.FALSE.** : on tiendra compte des discontinuités.
- n\_min\_ZC=5** : nombre minimum de couches dans une ZC. ,
- l\_fac=.TRUE.** : on rapprochera à mieux que  $d_{zc}\%$  chaque discontinuité d'un bord de couche.
- d\_zc=5** : taux de rapprochement des discontinuités d'un bord de couche.

L'ajustement des constantes de répartition *est délicat*. Les valeurs par défaut sont `ctel=0`, `ctep=-1`, `ctem=15`, `cter=0`, `ctet=0`. Prendre garde que `ctep` et `ctet` doivent être *négatives*, car elles affectent respectivement la pression et la température, quantités décroissantes du centre à la surface.

Comme pour les abondances initiales *cf.* §5.2 (Page 60), il est possible de caractériser ces réglages en donnant au fichier des réglages, le nom du modèle avec l'extension `.rg`, exemple : `mon_modele.rg`. S'il existe dans l'environnement, ce seront les paramètres de ce fichier qui seront pris en compte pour *tous les* modèles de nom `mon_modele`. En son absence, ce seront les quantités du fichier `reglages`, si aucun de ces fichiers n'est présent dans l'environnement le calcul est interrompu.

## 5.10 Limites des shells source

Dans le fichier de données, *cf.* §4.5 (Page 33), en ajoutant au nom du type de dessin (`nom_des`) l'extension `HHe`, exemples : (`des_mHHe`, `des_mCHHe`) `CESAM2k` constitue un fichier ASCII, de nom `mon_modele.mHHe`, contenant, pour chaque

modèle, le numéro du modèle, l'âge, les limites en masse des shells source H et He (FORMAT(i4,es22.15,2es10.3). Exemple :

```

0 0.0000000000000000E+00 0.000E+00 0.000E+00
1 1.457003778976599E-05 0.000E+00 0.000E+00
2 2.622606802157879E-05 0.000E+00 0.000E+00
3 3.555089220702902E-05 0.000E+00 0.000E+00
4 4.301075155538921E-05 0.000E+00 0.000E+00
5 5.034857965187844E-05 0.000E+00 0.000E+00
6 5.830431518424666E-05 0.000E+00 0.000E+00
.
.
.
438 1.273514666553580E+02 8.460E-01 4.876E-01
439 1.273587701858585E+02 8.461E-01 4.916E-01
440 1.273658006341733E+02 8.461E-01 4.983E-01
441 1.273725838461919E+02 8.463E-01 5.049E-01
442 1.273791284635452E+02 8.457E-01 5.088E-01
443 1.273854522544298E+02 8.459E-01 5.154E-01
444 1.273915631736387E+02 8.460E-01 5.193E-01
445 1.273974741112376E+02 8.462E-01 5.257E-01

```

## 5.11 Zoom

Il est possible de personnaliser le dessin 'en\_masse' du diagramme HR en y dessinant un rectangle cible à atteindre par le trajet évolutif, ainsi que de modifier l'échelle du diagramme, en centrant celui-ci autour de la valeur du point cible à des distances précisées en DeX de  $\log T_{\text{eff}}$  et  $\log L$ . On peut aussi tracer une cible de  $[\text{Fe}/\text{H}]$  à atteindre, dans le cadre du dessin des abondances en fonction de la masse. Pour ce faire, coder `nom_des='zoom'` dans le fichier de données et, dans l'environnement, placer un fichier de nom `mon_modele.zoom`, dans lequel sont introduites les données nécessaires. Exemple d'un tel fichier `mon_modele.zoom`<sup>6</sup> :

```

&NL_DES
teff_des=6530.,
dteff_des=50., -30.,
zoom_t=0.2, -0.1,
l_des=6.844,
dl_des=0.6, -0.4
zoom_l=0.1, -0.2,
fesh_des=0.05,
dfesh_des=0.03, -0.04,
logteff_max=4.6,
logteff_min=3.5,
logl_max=5.,
logl_min=2.5

```

6. Exemple se trouvant dans le sous-directory EXPLOIT.

/

- Il n'y a pas de personnalisation si `teff_des` et/ou `l_des` est négatif ou nul.
- Le zoom n'est effectif qu'avec `zoom_t` et `zoom_l` non nuls.
- Pour ne pas dessiner de cible en métallicité, coder `fesh_des=100`.
- Les valeurs des abscisses et ordonnées maximales *i.e.* `logteff_max` (min), `logl_max` (min) **prévalent** sur celles des cibles; dans l'exemple précédent, les échelles logarithmiques en température effective et en luminosité variront respectivement dans les intervalles [4.6, 3.5] et [2.5, 5.].
- Pour **seulement** adapter les échelles du diagramme HR coder, par exemple :

```
&NL_DES
teff_des=0.,
dteff_des=0., 0.,
zoom_t=0., 0.,
l_des=0.,
dl_des=0., 0.
zoom_l=0., 0.,
fesh_des=0.,
dfesh_des=0., 0.,
logteff_max=3.6,
logteff_min=2.5,
logl_max=2.,
logl_min=-2.5
/
```

- Pour **seulement** cadrer le diagramme HR autour de la cible, coder :

```
&NL_DES
teff_des=6530.,
dteff_des=50., -30.,
zoom_t=0.2, -0.1,
l_des=6.844,
dl_des=0.6, -0.4
zoom_l=0.1, -0.2,
fesh_des=100.,
dfesh_des=0., 0.,
logteff_max=0.,
logteff_min=0.,
logl_max=0.,
logl_min=0.
/
```

La signification des variables est la suivante :

**teff\_des=6530.** : Température effective du centre de la cible.

**dteff\_des=50., -30.** : Ecart en température effective.

**zoom\_t=0.2, -0.1** : Extension en température effective autour de la cible en DeX.

**l\_des=6.844** : Luminosité du centre de la cible.

**dl\_des=0.6, -0.4** : Ecart en luminosité.

**zoom\_l=0.1, -0.2** : Extension en luminosité autour de la cible en DeX.

**fesh\_des=0.05** : Cible à atteindre en métallicité.

**dfesh\_des=0.03, -0.04** : Ecart de la cible en métallicité.

**logteff\_max=4.6** :  $\log_{10}$  de l'abscisse maximale du diagramme HR.

**logteff\_min=3.5** :  $\log_{10}$  de l'abscisse minimale du diagramme HR.

**logl\_max=5.** :  $\log_{10}$  de l'ordonnée maximale du diagramme HR.

**logl\_min=2.5** :  $\log_{10}$  de l'ordonnée minimale du diagramme HR.

Il est possible de personnaliser davantage, en donnant au fichier `zoom` le nom attaché au modèle avec l'extension `.zoom`, exemple : `mon_modele.zoom`. Dès lors, les dispositions précédentes ne seront effectives *que pour* le modèle `mon_modele`. Avec l'option `zoom`, CESAM cherche à utiliser le fichier `mon_modele.zoom`, en son absence, le fichier `zoom` est utilisé.

## 5.12 Device

Pour adapter les dimensions des cadres et le nom du device à utiliser, placer dans l'environnement du calcul, un fichier de nom `device`, dans lequel sont introduites les données nécessaires. Exemple d'un tel fichier `device`<sup>7</sup> :

```
&NL_DEVICE
h=6.5,
dh=1.5,
ld=8.5,
dl=2.,
xleft=1.8,
ybot=1.4,
device='/xw'
/
```

La signification des variables est la suivante (unités en cm) :

**h=6.5** : Hauteur des cadres.

**dh=1.5** : Espace vertical entre les cadres.

**ld=8.5** : Largeur des cadres.

**dl=2.** : Espace horizontal entre les cadres.

**xleft=1.8** : Distance du dessin à la limite gauche de l'écran.

**ybot=1.4** : Distance du dessin à la limite inférieure de l'écran.

**device='/xw'** : Nom du device.

On peut aussi modifier directement les valeurs de ces paramètres dans le module `mod_donnees` du sous-directory `SOURCE`.

<sup>7</sup>. Exemple se trouvant dans le sous-directory `EXPLOIT`.

## 5.13 Limitation des écritures

Des informations permettant de suivre en détail le déroulement des calculs apparaissant "on line" sur le moniteur. Elles peuvent s'avérer non pertinentes pour une exploitation "batch" et encombrer inutilement. CESAM2*k* a la possibilité d'en détourner la plus grande partie vers des fichiers qui pourront, au besoin, lus "off line", *cf.* § 4.16.3 (Page 53). Quand la nature cette limitation de sorties n'est qu'occasionnelle, il suffit de disposer, dans l'environnement du calcul, le fichier suivant de nom `blabla` :

```
&NL_BLABLA  
baratine=.FALSE.  
/
```

On trouvera ce fichier dans le sous-directory `EXPLOIT`.

# Chapitre 6

## Extensions

### 6.1 Ajout d'une chaîne de réactions thermonucléaires

Pour implanter une nouvelle chaîne de réactions thermonucléaires, il est souvent nécessaire d'ajouter un ou plusieurs isotopes, et aussi une ou plusieurs réactions thermonucléaires.

– *Pour ajouter un isotope*<sup>1</sup> :

1. Dans le module `mod_nuc` augmenter le PARAMETER `niso_tot` d'une unité, Exemple : `niso_tot=28`.
2. Dans la routine `taux_nuc` :
  - (a) Dans les déclarations, ajouter l'excès de masse de l'isotope et son symbole, Exemple : `Fe56=-60.6054d0`.
  - (b) Ajouter la masse de l'isotope, (il est conseillé d'ajouter la masse de l'isotope, dans la routine de constantes physiques `ini_ctes`, son symbole dans les déclarations du module `mod_donnees` et dans l'instruction de restriction `ONLY` de la routine `taux_nuc`, Exemple : `nucleot(27)=afe56, afe56=55.847d0, afe56, af18...`).
  - (c) Ajouter la charge de l'isotope, Exemple : `zit(28)=11`.
  - (d) Ajouter le nom de l'isotope, Exemple : `'B11 '`. (ces chaînes de 4 caractères sont à cadrer à gauche sauf pour les symboles de 2 caractères qui sont centrés.)
3. Si besoin, dans la routine `abon_ini` :
  - (a) Ajouter un rapport isotopique pour ce nouvel isotope :
    - i. Introduire sa valeur dans la liste disposée à la fin de la routine.
    - ii. Indiquer son nom dans les NAMELISTs `nl_rap_iso` et `nl_modif_mix` dans la partie déclarative de la routine.
4. Si besoin, dans la routine `saha`, ajouter les potentiels d'ionisation pour ce nouvel élément

---

1. CESAM2k prévoit la possibilité d'utiliser des éléments chimiques de l'hydrogène ( $Z = 1$ ) jusqu'au nickel ( $Z = 28$ ). Pour introduire des éléments de masse supérieure, les disposer dans la routine `abon_ini` en appliquant une procédure voisine de celle décrite pour l'ajout d'un isotope.

– *Pour ajouter une réaction :*

1. Dans la routine `taux_nuc` :
  - (a) Compléter la liste de commentaires en y indiquant le symbole de la réaction et son indice, Exemple : `réaction 46 : B11(p,g)C12`.
  - (b) Initialiser le nom de la réaction, le défaut de masse, et les charges des noyaux concernés, Exemple : `nom_react(2)='H2(p,g)He3'`  
`nuc=H2+p-He3 ; qt(2)=nuc ; izzt(2,1)=1 ; izzt(2,2)=1`.
  - (c) Entrer le taux de la réaction<sup>2</sup> `rt(i)` en ln et ne pas oublier d'y inclure les 1! ou 2! ou 3! du dénominateur, Exemple : `réaction 8 : C12(p,g)N13(e+ nu)C13 z0=6, z1=1.....` en recalculant, au besoin, les coefficients à partir des  $S(0)$ ,  $S'(0)$ ,  $S''(0)$  (Lang, 1968, eq. 4-49).
2. Créer la routine de réactions thermonucléaires en s'inspirant d'une des routines existantes, `ppcno12BeBFe.f` par exemple.
3. Dans la routine `tabul_nuc` :
  - (a) Ajouter la nouvelle chaîne Exemple : `CASE('ppcno12')`
  - (b) Entrer les paramètres du calcul des taux, le nombre de réactions, les correspondances d'indices, les isotopes utilisés ..., Exemple : `nreact=30 ; ind(16)=31 ; WRITE(2,6) ; WRITE(*,6)....`
4. Dans le module `mod_nuc` :
  - (a) Augmenter d'une unité le `PARAMTER nreact_tot`, Exemple : `nreact_tot=46`
  - (b) Introduire par un `INCLUDE` le nom de la routine de la chaîne de réactions, Exemple : `INCLUDE 'ppcno9.f'`
5. Dans la routine générique `nuc` :
  - (a) Rendre active et faire reconnaître la nouvelle routine.
6. Tester les algorithmes ainsi modifiés en exploitant les programmes du sous-directory `TEST test_tabul_reac.f` pour vérifier les taux des réactions et `test_jacobien_reac_nuc.f` pour s'assurer de l'exactitude des dérivées.

## 6.2 Ajout d'une mixture initiale

Ainsi qu'il est décrit au § 5.2 (Page 60), il est possible de personnaliser les abondances initiales à l'aide d'un fichier `mixture`. Quand la mixture initiale est d'intérêt général<sup>3</sup> il est préférable et très simple, d'en ajouter la table des abondances dans la routine `abon_ini` de la source, en lui donnant un nom d'au plus 20 caractères. Pour refaire ensuite le module exécutable, il suffit d'appliquer le script `rempl2k_mod` au module `mod_nuc` : `rempl2k_mod mod_nuc`.

---

2. C'est pour des raisons historiques que ces calculs sont effectués en ln. Les raisons étaient d'éviter d'une part, les erreurs de troncature et d'autre part, les dépassements de capacité avec la limitation à  $10^{38}$ .

3. Pour faire profiter la communauté de votre travail, communiquer la routine ainsi transformée aux responsables de la mise à jour de *CESAM2k*.

# Chapitre 7

## Debug

*Quand je me regarde je me désole,  
Quand je me compare je me console.*

### 7.1 Debug

La structure de CESAM2k en modules permet une vérification efficace de la syntaxe par le compilateur mais aussi de limiter les recompilations aux modules dans lesquels des erreurs ont été signalées.

Quand une erreur est détectée en cours d'exécution CESAM2k édite un message facilitant sa localisation. **Exemple** : un message tel que "sortie depuis cesam 7" indique que l'arrêt a été provoqué à l'instruction `CALL sortie('cesam 7')` de la routine `cesam.f`. Il convient de s'y reporter pour analyser les raisons de la "sortie".

Pour les cas simples, la bibliothèque de debug `libcesam2k-dbg.a` et l'exécutable `cesam2k-dbg.out` ont été créés lors de la première phase de l'installation, cf. § 1.5 (Page 6). Pour une recherche plus approfondie des dispositifs de debug sont disposés dans les routines les plus sensibles.

**Exemple** : un bug est suspecté dans la routine `static.m13` du module `mod_static`.

1. Dans le directory `SOURCE`, inclure les corrections de debug dans les routines concernées.
2. Dans le programme `test_cesam.f` du directory de travail<sup>1</sup>, actualiser le chemin de la `SOURCE`<sup>2</sup> et décocher les "c" en colonne 1 des lignes correspondants aux modules supposés être concernés par le bug. Pour le présent exemple :

```
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_numerique.f'  
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_donnees.f'  
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_variables.f'  
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_etat_saha.f'
```

---

1. S'il n'y a pas été placé par le script `create_evol_space`, reprendre `test_cesam.f` du sous-directory `EXPLOIT`.

2. utiliser le chemin complet car ni `tilde` ni `$HOME` ne sont reconnus par F95.

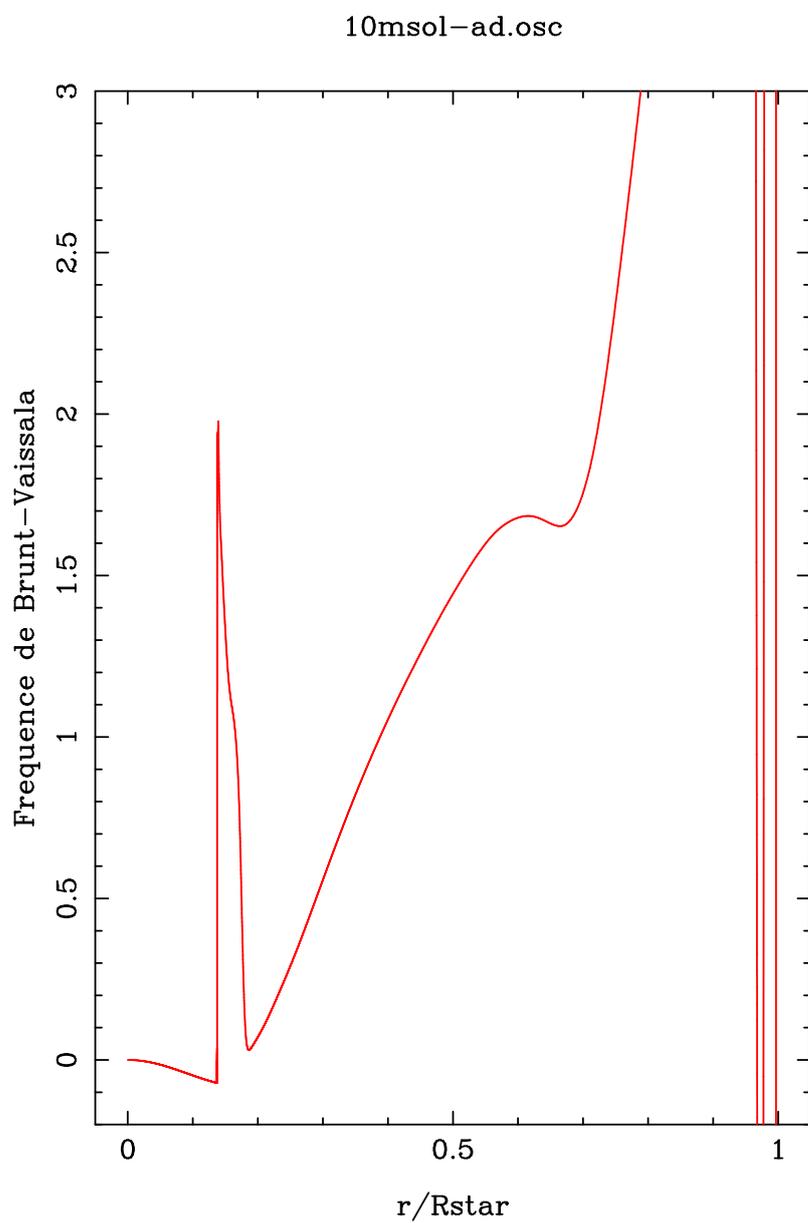


FIGURE 7.1 – Profil de la fréquence de Brunt-Väissälä d’un modèle de  $10M_{\odot}$  de séquence principale ( $X_c = 0.35$ ), calculé avec la procédure décrite § 16 (Page 80).

```

c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_etat.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_opa.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_conv.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_atm.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_nuc.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_bp_for_alecian.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_evol.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_static.f' <----- bug?
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_cesam.f'
c      INCLUDE '/home/pierre/CESAM2k/SOURCE/mod_exploit.f'

c*****

PROGRAM test_cesam

USE mod_cesam

IMPLICIT NONE

CALL cesam

STOP

END PROGRAM test_cesam

```

3. Créer localement l'exécutable de debug en saisissant au choix :  
`make -Bf Make_exe PRG=test_cesam` ou `exe2k-dbg test_cesam`
4. Exécuter la recherche des dysfonctionnements en lançant l'exécutable :  
`test_cesam.out`

Au link, le fichier `mod_static.mod`, créé localement, sera utilisé et non pas celui de la `SOURCE`. Lors de l'exécution, les instructions de debug, alors prises en compte, permettront de détecter le défaut.

Une fois la difficulté identifiée, dans la `SOURCE` corriger les routines concernées puis mettre à jour les bibliothèques et les exécutables conformément au `README_installation` de la `SOURCE`.

Localement, supprimer les fichiers créés localement pour le debug. Pour l'exemple, le module `mod_static.mod` ainsi que l'exécutable `test_cesam.out` désormais inutiles mais sources potentielles de dysfonctionnements.

## 7.2 Bugs connus

### 1. *Erreurs à la compilation.*

- (a) Lors de développements, une cause fréquente de difficulté est due aux erreurs de calcul des dérivées, par rapport aux variables principales, des coefficients des équations de la structure interne. Pour en faciliter la

détection, dans les routines `static_m13` et `static_m23`, la comparaison entre dérivées analytiques et dérivées numériques est possible sur des intervalles de points de collocation à définir en codant `deriv=.TRUE.`.

- (b) Dans des routines de provenance externe *e.g.* `etat_ceff`, `etat_eff`, `ferdir`, .. les variables ne sont pas déclarées, ou encore sont déclarées de façon implicite. Il y a risque d'erreur ou de "warning" lors de la compilation. **Remède** : y supprimer l'option `IMPLICIT NONE`.
  - (c) Avec le compilateur F95 de DIGITAL, on obtient un "*compilator internal error*" lors de la compilation du module `mod_etat`. La routine `etat_eff` est en cause. **Remède** : utiliser un degré d'optimisation au plus égal à 2.
  - (d) Avec un compilateur dont la version n'est pas à jour de la norme F95, (comme certaines versions du compilateur `ifc`) il y a refus d'effectuer une allocation dans une routine autre que celle où est déclarée l'allocation. **Remède** : remplacer dans les routines concernées, les instructions `ALLOCATABLE` par `POINTER` pour les tableaux incriminés. Ou encore, utiliser un compilateur plus récent.
2. **Mauvaise convergence**, le processus itératif "tourne en rond" : la convergence s'amorce, puis les corrections ne diminuent plus, ou oscillent autour d'une valeur fixe. **Cause possible** : interpolation non dérivable de l'opacité *e.g.* avec des opacités interpolées linéairement. **Remède** : si l'augmentation de l'ordre d'interpolation des opacités est la solution théorique, elle est souvent impraticable, et il y a risque d'oscillations à cause de pas d'interpolation trop grands, ou de raccords pas assez lisses entre tables ; pratiquement, il faut se contenter d'une précision moins grande pour passer le point difficile. Ce type de difficulté se produit assez souvent dans l'atmosphère, à cause du raccord entre tables d'opacités d'origines différentes. Dans certaines conditions, `CESAM2k` "*force la convergence*".
  3. **Variations temporelles trop importantes** de la pression, de la température, et de là, de l'entropie spécifique. On obtient le message du type : "`lnP varie trop..: diminution du pas temporel`". Le pas temporel est diminué sans que, pour autant, le problème disparaisse. **Cause possible** : discontinuité temporelle de la composition chimique avec un cœur convectif. **Remède** : parfois, bien que cela semble paradoxal et que l'on n'en comprenne pas la raison, il suffit de reprendre l'évolution avec un pas temporel plus grand pour que le point délicat soit passé ; on peut aussi tenter d'augmenter les paramètres `d_press` et/ou `d_temp`, des valeurs plus élevées en augmentant la robustesse dégradent la précision ; on peut éliminer ces contrôles en donnant une valeur très grande ( $10^{10}$  par exemple) aux paramètres ; ils sont initialisés dans la routine `cesam`, *cf.* § 5 (Page 59). On contourne parfois la difficulté en utilisant l'approximation de Kippenhahn *cf.* § 5.9 (Page 67). Pour ajuster ces paramètres il est pratique d'utiliser la possibilité de personnaliser les réglages de `CESAM2k` *cf.* § 5 (Page 59).
  4. **Divergence au premier pas temporel**. **Cause possible** : pas temporel initial trop élevé. **Remède** : diminuer le `dt0` initialisé dans la routine `cesam` suivant le type de précision requis. Comme ci-avant il est pratique d'utiliser un fichier de réglage.

5. **La localisation des limites zones radiatives / zones convectives** sur un point de grille ne fonctionne pas. **Cause possible** : les gradients  $\nabla_{\text{rad}}$  et  $\nabla_{\text{ad}}$  ne se croisent pas franchement, ou encore la limite se déplace notablement *e.g.* , lors de la disparition d'un cœur convectif. **Remède** : quand les gradients sont voisins, la limite zone radiative / zone convective est mal définie, y vouloir forcer un point de grille exactement est illusoire et inutile, CESAM2k s'accommode de la situation en imposant la position obtenue après le nombre d'itérations fixé dans la routine `cesam` suivant le type de précision requis. Pour ajuster ces paramètres, il est pratique d'utiliser la possibilité de personnaliser les réglages de CESAM2k, *cf.* § 5 (Page 59).
6. **Pour une évolution avec PMS**, en précision normale `np`, une fois le modèle initial obtenu, il n'y a pas convergence pour le premier pas temporel. **Cause possible** : La cause de cette difficulté mineure, n'a pas été trouvée. **Remède** : CESAM2k divise le pas temporel par 2 et la convergence est obtenue sans difficulté.
7. **Impossibilité de construire un modèle de séquence principale d'âge zéro.** **Cause possible** : utilisation d'une composition chimique trop éloignée de l'équilibre *e.g.* celle utilisée pour les modèles de pré-séquence principale avec les abondances cosmiques de H2 et Li7. **Remède** : adapter la composition chimique initiale. Pour ajuster ces paramètres, il est pratique d'utiliser la possibilité de personnaliser les abondances initiales de CESAM2k, *cf.* § 5 (Page 59).
8. **Pour des masses élevées**  $\sim 10M_{\odot}$  et plus, après la disparition de l'hydrogène central, des ZC apparaissent, disparaissent nuisant à la convergence. **Cause possible** : présence de zones semi-convectives. **Remède** : difficulté chronique, CESAM2k ne prévoit pas de traitement de la semi-convection.
9. **Convergence impossible** en utilisant le terme correctif de pression turbulente  $\frac{d \ln P_{\text{gaz}}}{d \ln P}$  *i.e.* avec une valeur strictement négative pour l'argument `cpturb` de la NAMELIST `NL_CONV`. **Cause possible** : Ce problème résulte vraisemblablement d'une incohérence de la MLT. **Remède** : néant en l'état actuel. **L'utilisation de la pression turbulents est écatée dans la présente version de CESAM2k**
10. **Le programme** `lit_osc` du sous-directory `EXPLOIT` de la source refuse de lire un fichier d'extension `.osc`. **Cause possible** : Pour une raison inconnue, la première ligne du fichier `mon_modele.osc` est "repliée" et le fichier est alors illisible. **Remède** : "déplier" cette ligne de façon à ce que la date se trouve à la fin de cette première ligne.
11. **Le compilateur refuse de compiler** une routine avec un niveau élevé d'optimisation. **Cause possible** : Caprice du compilateur. **Remède** : utiliser un niveau de compilation moins élevé.
12. **Très mauvaise convergence du calcul d'atmosphère I.** **Cause possible** : passage d'un point anguleux de la table d'opacité, ou modification de l'indice de grille du point d'indice  $n_{\star}$ <sup>3</sup>. **Remède** : souvent, après quelques

---

3. où  $T = T_{\text{eff}}$ ,  $R = R_{\star}$ ,  $M = M_{\star}$ .

pas temporels, on s'écarte des points anguleux, et ces difficultés disparaissent. Utiliser des fonctions d'interpolation dérivables pour les tables d'opacité.

13. **Très mauvaise convergence du calcul d'atmosphère II. Cause possible** : condition limite sur la pression qui ne peut être satisfaite au niveau de précision élevé requis par l'intégration de l'atmosphère. **Remède** : utiliser la condition limite sur la densité.
14. **L'évolution s'arrête "normalement"** avant que l'âge requis ne soit atteint. **Cause possible** : signe erroné (*respt.* valeur trop forte, faible) du paramètre `log_teff` (*respt.* `x_stop`, `t_stop`) de la NAMELIST `n1.temps` du fichier `mon_modele.don`. **Remède** : modifier ces paramètres suivant les indications du § 4.4 (Page 31) précédent.
15. **Mauvaise convergence de l'algorithme d'intégration temporelle pour  ${}^2\text{H}$** , se produit vers la fin de la pré-séquence principale, et parfois conjointement aux sorties de table d'opacité. **Cause possible** : oscillations au moment de la mise à l'équilibre du deutérium. **Remède** : CESAM2k "force la convergence" qui, souvent au niveau de quelques  $10^{-5}$ , n'est pas si mauvaise.
16. **Oscillations** de la quantité  $A = \frac{1}{\Gamma_1} \frac{\partial \ln P}{\partial \ln R} - \frac{\partial \ln \rho}{\partial \ln R}$  d'un fichier `mon_modele*.osc` cf. § A.1 (Page 83). Ces oscillations ont plusieurs causes. Beaucoup d'efforts aient été consacrés à cette difficulté. **Remède** : jusqu'à présent, le moins mauvais remède consiste à supprimer "à la main" les couches incriminées.

Si, comme pour le Soleil actuel, dans les zones mélangées la température reste inférieure à 0.5MK, température d'amorçage des réactions nucléaires, il n'y a pas de modification de composition chimique dans les zones mélangées, il est inutile de placer une discontinuité numérique aux limites ZR/ZC. **Remède** : s'il n'y a pas d'autre raison de modification de la composition chimique, utiliser l'option `no_discon=.TRUE..`

Ou encore, puisque la solution avec diffusion ne comporte pas de discontinuités, introduire un peu de diffusion turbulente par l'intermédiaire du fichier de données :

```

/
&NL_DIFF
DIFFUSION=.TRUE.,
NOM_DIFFM='diffm_0',
NOM_DIFFT='diff_t_nu',
D_TURB=10.d0,
RE_NU=1.d0,
NOM_FRAD='no_frad'
/

```

Le coût à payer sera un effort de calcul nettement plus important qu'avec `DIFFUSION=.FALSE..` La figure 7.1 (Page 76) représente le profil de la fréquence de Brunt-Väissälä obtenu pour un modèle de  $10M_{\odot}$  de séquence principale ( $X_c = 0.35$ ), calculé avec la procédure qui vient d'être décrite. Au cours de l'évolution le cœur convectif a régressé de  $0.22R_{\star}$  à  $0.14R_{\star}$ .

17. **Mauvaise convergence pour la restitution de l'atmosphère.** Ce problème se produit parfois lors de la reprise d'un modèle. La difficulté est due

au fait que l'atmosphère qui a été reprise en binaire n'est pas suffisamment proche de celle que CESAM2k désire calculer. **Remède** : supprimer le modèle d'atmosphère en binaire `mon_modele_B.atm`. Pour initialiser ou reprendre le calcul, CESAM2k utilisera l'atmosphère standard qu'il contient.

18. **Mauvaise convergence, pas temporel tendant vers 0**, et/ou oscillations de la vitesse angulaire au voisinage des limites zones radiatives / zones convectives. Les causes ne sont pas identifiées. Une conséquence est la génération d'instabilités de composition chimique, qui se transmettant à la densité, génèrent des coefficients très bruités pour les équations de la diffusion du moment cinétique. Une augmentation du coefficient de diffusion turbulente `d_turb`, cf. § 4.4 (Page 31), permet, parfois, de faire disparaître d'anomalie.
19. **Profil très bruité du taux de variation du poids moléculaire moyen** en particulier au voisinage des limites zones radiatives / zones convectives. **Cause possible** : instabilité vraisemblablement due au calcul de l'énergie gravitaionnelle avec un pas temporel très petit. **Remède** : comme décrit ci-avant, l'introduction de diffusion turbulente permet parfois de contourner la difficulté.



# Annexe A

## Liste des fichiers ASCII

### A.1 Fichiers pour oscillations

Les fichiers ASCII pour les oscillations ont une en-tête commune. Leur contenu diffère ensuite suivant leur type.

en-tête :

- Lignes 1-4 : identification et physique utilisée.
- Ligne 5 : nombre, noms des éléments chimiques utilisés.
- Ligne 6 : nombre de couches, de “global” (13), de variables, d’éléments chimiques (redondant), indice de la vitesse angulaire (s’il n’en est pas tenu compte, l’indice est -1).

Exemple :

```
Fichier pour inversion: test-inv.osc
CESAM2k version 0.0.0.0 lagr colloc 1 2 np no diffus, 31 Aout 2003 17h45
Physique utilisée: etat_eff, opa_int_zsx, conv_jmj, ppcno9, NACRE
solaire_gn, lim_atm, hopf, perte_ext, diffm_mp, difft_nu, ctes_94
 10 H1 He3 He4 C12 C13 N14 N15 O16 O17 Si28
      472      13      25      10      -1
1.989190000000E+33 6.959888774683E+10 3.846010241481E+33 1.723296157963E-02 7.033715921640E-01
1.601359931639E+00 7.033715881909E-01 2.793954502293E-01-1.234804262016E+02-1.250212463069E+02
4.650000000000E+03 0.000000000000E+00 0.000000000000E+00
6.965125316795E+10 1.125188830934E-10 4.712514552821E+03 1.082443642936E+03 3.550000000000E-09
2.317104979700E-04 3.846010241481E+33 5.992732100399E-03 0.000000000000E+00 1.664711685203E+00
3.986098681160E-01 1.004603449555E+00 1.630687926937E+08 1.300571860485E+00 2.500897748491E+03
0.000000000000E+00 1.729836584721E+00 8.429278614488E-01 0.000000000000E+00 0.000000000000E+00
.....
3.983397251906E-01 9.536638658353E-01 2.457927862174E+08 8.289918269170E-01 0.000000000000E+00
0.000000000000E+00-1.948967682005E+00 3.878822249046E-01 2.023289474023E+02 1.828866051074E+01
1.000000000000E+00 3.288637435509E-01-5.586998681649E-03 6.642708700530E-03-8.894901969547E-03
3.750487629847E-01 1.039826643345E-05 6.071493899575E-01 1.637161572818E-05 4.475726236722E-06
4.374234929779E-03 1.935496963607E-07 7.968745830613E-03 3.970500883875E-04 5.030377050940E-03
```

#### A.1.1 Fichier pour oscillations adiabatiques

glob: variables globales du fichier mon\_modele-ad.osc

```
glob(1)=mstar*msol
glob(2)=rtot*rsol
```

```

glob(3)=ltot*lsol
glob(4)=z0
glob(5)=x0
glob(6)=alpha
glob(7)=X dans ZC
glob(8)=Y dans ZC
glob(9)=d2p
glob(10)=d2ro
glob(11)=age
glob(12)=wrot vitesse de rotation globale
glob(13)=w_rot initial
glob(14)=g constante de la gravitation utilisée
glob(15)=msol masse solaire utilisée
glob(16)=rsol rayon solaire utilisé
glob(17)=lsol luminosité solaire utilisée

```

var: variables locales utilisées ; nvar=22 pour oscillations adiabatiques

```

var(1,i)=r*rsol
var(2,i)=log(m/mstar) -1.d38 au centre
var(3,i)=t
var(4,i)=Ptot
var(5,i)=ro
var(6,i)=gradient réel d ln T / d ln P
var(7,i)=l
var(8,i)=kap
var(9,i)=énergie thermo+gravifique
var(10,i)=grand Gamma1
var(11,i)=gradient adiabatique
var(12,i)=delta
var(13,i)=cp
var(14,i)=mu elec.
var(15,i)=vaissala, 0 au centre
var(16,i)=vitesse angulaire, radian/sec
var(17,i)=d ln kappa / d ln T
var(18,i)=d ln kappa / d ln ro
var(19,i)=d epsilon(nuc) / d ln T
var(20,i)=d epsilon(nuc) / d ln ro
var(21,i)=Ptot / Pgas (grad_mu sans pression turbulente)
var(22,i)=gradient radiatif

```

composition chimique

```

var(22+j,i)=xchim(j)*nucleo(j), j=1,nbelem

```

### A.1.2 Fichier pour oscillations non adiabatiques

glob: variables globales du fichier mon\_modele-nad.osc

```

var: variables
glob(1)=mstar*msol
glob(2)=rtot*rsol
glob(3)=ltot*lsol
glob(4)=z0
glob(5)=x0
glob(6)=alpha
glob(7)=X dans ZC
glob(8)=Y dans ZC
glob(9)=d2p
glob(10)=d2ro
glob(11)=age
glob(12)=wrot vitesse de rotation globale
glob(13)=w_rot initial
glob(14)=g constante de la gravitation utilisée
glob(15)=msol masse solaire utilisée
glob(16)=rsol rayon solaire utilisé
glob(17)=lsol luminosité solaire utilisée

```

var: variables locales utilisées ; nvar=44 pour oscillations non adiabatiques

```

var(1,i)=r*rsol
var(2,i)=log(m/mstar) -1.d38 au centre
var(3,i)=t
var(4,i)=Ptot
var(5,i)=ro
var(6,i)=gradient reel d ln T / d ln P
var(7,i)=l
var(8,i)=kap
var(9,i)=énergie thermo+gravifique
var(10,i)=grand Gamma1
var(11,i)=gradient adiabatique
var(12,i)=delta
var(13,i)=cp
var(14,i)=mu elec.
var(15,i)=vaissala, 0 au centre
var(16,i)=vitesse angulaire, radian/sec
var(17,i)=d ln kappa / d ln T
var(18,i)=d ln kappa / d ln ro
var(19,i)=d epsilon(nuc) / d ln T
var(20,i)=d epsilon(nuc) / d ln ro
var(21,i)=Ptot / Pgas (grad_mu sans pression turbulente)
var(22,i)=gradient radiatif

```

```

var(23,i)=d Gamma1 / d log P
var(24,i)=d Gamma1 / d log T
var(25,i)=d Gamma1 / dY = d Gamma1 / dZ
var(26,i)=dP / dro (TX)
var(27,i)=dP / dT (roX)
var(28,i)=dP / dX (Tro)
var(29,i)=du / dro (TX)
var(30,i)=du / dT (roX)
var(31,i)=du / dx(Tro)
var(32,i)=énergie interne
var(33,i)=d^2P / dro^2 (TX)
var(34,i)=d^2P / dro dT (X)
var(35,i)=d^2P / dT^2(roX)
var(36,i)=d^2U / dro^2 (TX)
var(37,i)=d^2U / dro dT (X)
var(38,i)=d^2U / dT^2 (X)
var(39,i)=dK / dX
var(40,i)=d^2K / dT^2
var(41,i)=d epsi / dX
var(42,i)=dX / dR
var(43,i)=J-B
var(44,i)=Edding. facteur

```

composition chimique

```
var(44+j,i)=xchim(j)*nucleo(j), j=1,nbelem
```

### A.1.3 Fichier pour inversions

glob: variables globales du fichier mon\_modele-inv.osc

```

glob(1)=mstar*msol
glob(2)=rtot*rsol
glob(3)=ltot*lsol
glob(4)=z0
glob(5)=x0
glob(6)=alpha
glob(7)=X dans ZC
glob(8)=Y dans ZC
glob(9)=d2p
glob(10)=d2ro
glob(11)=age
glob(12)=wrot vitesse de rotation globale
glob(13)=w_rot initial
glob(14)=g constante de la gravitation utilisée
glob(15)=msol masse solaire utilisée

```

```

glob(16)=rsol rayon solaire utilisé
glob(17)=lsol luminosité solaire utilisée

var: variables locales utilisées ; nvar=25 pour inversion

var(1,i)=r*rsol
var(2,i)=log(m/mstar) -1.d38 au centre
var(3,i)=t
var(4,i)=Ptot
var(5,i)=ro
var(6,i)=gradient reel d ln T / d ln P
var(7,i)=1
var(8,i)=kap
var(9,i)=énergie thermo+gravifique
var(10,i)=grand Gamma1
var(11,i)=gradient adiabatique
var(12,i)=delta
var(13,i)=cp
var(14,i)=mu elec.
var(15,i)=vaissala, 0 au centre
var(16,i)=vitesse angulaire, radian/sec
var(17,i)=d ln kappa / d ln T
var(18,i)=d ln kappa / d ln ro
var(19,i)=d epsilon(nuc) / d ln T
var(20,i)=d epsilon(nuc) / d ln ro
var(21,i)=Ptot / Pgas (grad_mu sans pression turbulente)
var(22,i)=gradient radiatif
var(23,i)=d Gamma1 / d log P
var(24,i)=d Gamma1 / d log T
var(25,i)=d Gamma1 / dY = d Gamma1 / dZ

composition chimique

var(25+j,i)=xchim(j)*nucleo(j), j=1,nbelem

```

#### A.1.4 Utilisation du nombre maximum de couches

Avec certains types de précision, *cf.* §4.4 (Page 31), il est possible de créer un fichier ASCII d'oscillation comportant au moins un nombre de couches fixé à l'avance; le nombre maximum de couches étant utilisé lors du calcul des derniers modèles. Les critères requis pour ce faire sont les suivants :

- Type de précision utilisé : 'sa', ou 'co'.
- **ET** réalisation d'un des critères suivants :
  - Age du modèle à calculer inférieur d'un million d'années de l'âge maximum à atteindre.
  - Numéro du modèle à calculer égal à NMAX\_MODELS - 1.
  - Température effective voisine de la limite demandée.

- Abondance centrale d’hydrogène voisine de la limite demandée.
- Extension du noyau d’hélium voisine de la limite demandée.

Une autre possibilité est de définir un nombre négatif de couches dans le fichier de données, Ex : NMAX=-1000, *cf.* § 4.5 (Page 33).

## A.2 Fichier pour diagramme HR

Le fichier pour tracé/exploitation du diagramme HR est systématiquement implémenté au cours de chaque exécution de CESAM2*k*. Il est créé lors de l’initialisation d’un modèle sur la séquence principale d’âge zéro homogène ou la pré-séquence principale homogène. Il est complété en cas de reprise d’un modèle existant.

Pour chaque modèle, sur une première ligne sont reportés :

1. l’âge en  $10^6$  ans,
2. le nombre d’éléments du vecteur de composition chimique,
3. le type de diffusion du moment cinétique (0 sans diffusion de  $\Omega$ ),
4. le nombre de limites zones radiatives / zones convectives
5. le numéro du modèle,
6. pour chaque limite zone radiative / zone convective la mention "T" ou "F" suivant qu’il s’agit, ou non, du début d’une zone convective ;
7. suivant le nombre de limites, une ou plusieurs lignes sont nécessaires pour indiquer :
  - (a)  $\log T_{\text{eff}}$ ,
  - (b)  $\log L/L_{\odot}$ ,
  - (c)  $\log R/R_{\odot}$
  - (d)  $M_{\star}/M_{\odot}$
  - (e) pour chaque limite,  $(M_{\star} - m)/M_{\odot}$ ,  $r/R_{\odot}$ ,  $r_{\text{ov}}/R_{\odot}$ ,  $\frac{\partial N^2}{\partial \ln R}$  ( $r_{\text{ov}}$  étant le rayon de la limite étendue par overshooting et  $N^2$  la fréquence de Brunt-Väissälä aux limites ZR/ZC, estimée numériquement si le suffixe "T" affecte le nom de la sortie ASCII, *cf.* § 4.5 (Page 33).

Dans les cas particuliers, on utilise les conventions suivantes :

- modèle totalement convectif : 1 (seule) limite "F" placée au centre ( $r/R_{\odot} = 0$ ),
- modèle totalement radiatif : 0 (aucune) limite "F" placée en -100 ( $r/R_{\odot} = -100$ ).
- Viennent ensuite les noms des éléments chimiques et leurs abondances, par unité de masse, au centre et à la surface. S’il n’est pas imposé d’overshoot, les valeurs  $r_{\text{ov}}$  des rayons des limites étendues sont fixées à  $-1.E10$ , de même pour les dérivées de la fréquence de Brunt-Väissälä.
- Sur la ligne suivante sont indiquées des valeurs approchées (*cf.* Schatzman & Praderie (1990), p.389, Eq. 100, 101, 102) de quantités sismologiques :  $\nu_0$ ,  $\partial\nu_{02}$ ,  $\partial\nu_{13}$ ,  $A = -\int 1/r dc/dr$  et la période en minutes :

$$P_0 = 2\pi^2 / \int_0^{R_{\star}} N/r dr.$$

– S'il est tenu compte de la diffusion du moment cinétique, une ligne supplémentaire, identifiée par `Wrot` indique les vitesses linéaire et angulaire de la couche externe. Exemple :

```

0.0000000000000000E+00 10 0 2 0 F T
3.749146E+00-1.723498E-01-6.098130E-02 1.000000E+00 9.053737E-01 1.262897E-01
-1.000000E+10 2.341106E+04 2.969686E-02 6.310311E-01-1.000000E+10-7.275880E+05
H1 6.97718E-01 6.97718E-01
He3 8.91487E-05 8.91487E-05
He4 2.82700E-01 2.82700E-01
C12 3.33842E-03 3.33842E-03
C13 4.02357E-05 4.02357E-05
N14 1.03227E-03 1.03227E-03
N15 4.06226E-06 4.06226E-06
O16 9.39680E-03 9.39680E-03
O17 3.80402E-06 3.80402E-06
Si28 5.67649E-03 5.67649E-03
1.752E+02 2.288E+01 3.813E+01 7.627E+01 1.025E+02
<-- nu0, dnu02, dnu13, A, P0
Wrot 5.00050E+01 4.92437E-05 <-- qu'avec diffusion du moment cinétique
1.0000000000000000E+01 10 2 1 F T
3.750235E+00-1.660591E-01-5.995672E-02 1.000000E+00 9.485921E-01 1.002733E-01
-1.000000E+10 7.346418E+04 2.900062E-02 6.330032E-01-1.000000E+10-1.096457E+06
H1 6.97097E-01 6.97795E-01
He3 6.62006E-05 8.91324E-05
He4 2.83225E-01 2.82627E-01
C12 2.45937E-03 3.33777E-03
C13 4.11117E-04 4.02273E-05
N14 1.66274E-03 1.03208E-03
N15 2.18466E-07 4.06144E-06
O16 9.39741E-03 9.39506E-03
O17 3.87613E-06 3.80329E-06
Si28 5.67687E-03 5.67553E-03
1.752E+02 2.288E+01 3.813E+01 7.627E+01 9.784E+01
<-- nu0, dnu02, dnu13, A, P0
Wrot 5.21773E+01 5.03611E-05 <-- qu'avec diffusion du moment cinétique
.....

4.6850000000000000E+03 10 1 31 T
3.761741E+00-3.270666E-06 2.093004E-06 1.000000E+00 2.259322E-02 7.173108E-01
-1.000000E+10-1.782865E+06
H1 3.22007E-01 7.30094E-01
He3 6.46882E-06 8.19412E-05
He4 6.57151E-01 2.51936E-01
C12 2.05912E-05 3.04480E-03
C13 5.63346E-06 3.64768E-05
N14 5.13761E-03 9.44078E-04
N15 2.25834E-07 3.69734E-06
O16 8.99719E-03 8.61328E-03
O17 7.99800E-04 3.47279E-06

```

```

Si28 5.87454E-03 5.24189E-03
 1.510E+02 1.390E+01 2.316E+01 4.632E+01 3.660E+01
 <-- nu0, dnu02, dnu13, A, P0
Wrot 1.94190E+02 8.24763E-05 <-- qu'avec diffusion du moment cinétique

```

Les fichiers pour diagramme HR sont lus par la routine `lit.hr` du module `mod_exploit` et exploités par le programme `des2k_hr.f`.

### A.2.1 Fichier simplifié pour diagramme HR

Ce fichier est créé en lieu et place du fichier HR décrit ci-dessus. Une ligne par modèle comporte l'âge,  $\log T_{\text{eff}}$ ,  $\log L/L_{\odot}$ ,  $\log R/R_{\odot}$ , le numéro du modèle. Exemple :

```

0.0000000000000000E+00 3.620205E+00 1.873363E+00 1.221085E+00 0
9.786241004082362E-04 3.623663E+00 1.810559E+00 1.182668E+00 1
2.152973020898119E-03 3.626855E+00 1.750447E+00 1.146141E+00 2
3.562191725485980E-03 3.629826E+00 1.692302E+00 1.111047E+00 3
5.253254170991412E-03 3.632618E+00 1.635643E+00 1.077063E+00 4
7.282529105597930E-03 3.635253E+00 1.580105E+00 1.043958E+00 5
9.717659027125751E-03 3.637745E+00 1.525402E+00 1.011563E+00 6
.....
.....

```

Ces fichiers sont lisibles par la routine `lit_lr.f` du module `mod_exploit` et exploités par le programme `des2k_lr.f`.

## A.3 Fichier ASCII des variables de la diffusion du moment cinétique

**PB** Cette fonction n'est pas active dans la présente version de `CESAM2k`.

Le fichier ASCII des variables de la diffusion du moment cinétique est créé dans l'environnement du calcul suivant les circonstances décrites au § 4.11 (Page 44). Il peut être exploité par le programme de dessin `des2k_rot` du sous-directory `EXPLOIT`, cf. § 1.3 (Page 4).

en-tête :

- Lignes 1-4 : identification du modèle et physique utilisée.
- Ligne 5 : noms des variables dans l'ordre d'écriture.
- Ligne 6 : nombre, noms des éléments chimiques utilisés.
- Ligne 7 : nombre de couches, nombre total de variables, nombre de variables, d'éléments chimiques (redondant), identificateur de formalisme de diffusion de moment cinétique (3 pour Talon & Zhan 1997, 4 pour Mathis & Zahn 2004), indice de  $^4\text{He}$ , numéro du modèle, indice de  $^7\text{Li}$ .
- Ligne suivantes : `FORMAT(5es19.12)` variables et composition chimique.

Le modèle est listé du centre vers la surface. Exemple :

## A.4. FICHER ASCII DES VARIABLES DE LA ZÔNE D'INSTABILITÉ DES CÉPHÉIDES91

```
Fichier pour la diffusion du moment cinétique: 1.0_coeff_rota.dat
CESAM2k version V2.2.0 lagr colloc 2 3 pr no diffus, 18 Mai 2006 11h59
Physique utilisée: etat_eff, opa_yveline, conv_jmj, ppcno9, ctes_94, NACRE, diff_tz97
solaire_gn, lim_atm, hopf, pertm_ext, pertw_0, diffm_mp, diff_t_nu, diffw_mpz, diff_tz97
Variables : R, M, Omega, U, Theta, Psi, Lambda, Flux, Deff, Dh, Dv, T, ro, grad_mu, Xchim
10 H1 He3 He4 C12 C13 N14 N15 O16 O17 Si28
      601      24      14      10      3      3      42      0
0.000000000000E+00 0.000000000000E+00 2.492796313373E-06 1.222009447047E-15-2.215093172271E-11
4.146564017724E-13 2.345111261140E-11-9.012551719834E-35 2.621399230830E+00 3.397109673134E+00
2.621399230830E+00 1.340098717163E+07 8.330731227990E+01 1.426753865039E-02 6.871377248045E-01
5.018205056432E-05 2.922381643859E-01 1.473316033117E-05 4.025153911526E-06 5.082420872692E-03
2.299827930187E-07 9.638362729473E-03 6.971562756039E-06 5.827185297122E-03
1.644625353770E-02 2.636507234666E-04 2.492796313373E-06 1.222009447047E-15-2.215093172271E-11
4.146564017724E-13 2.345111261140E-11-9.012551719834E-35 2.621399230830E+00 3.397109673134E+00
2.621399230830E+00 1.340098717163E+07 8.330731227990E+01 1.426753865039E-02 6.871377248045E-01
5.018205056432E-05 2.922381643859E-01 1.473316033117E-05 4.025153911526E-06 5.082420872692E-03
2.299827930187E-07 9.638362729473E-03 6.971562756039E-06 5.827185297122E-03
2.464852077382E-02 8.840621368772E-04 2.492616591912E-06 3.215689566074E-15-4.144404321209E-11
1.675631823126E-12 4.481166434756E-11-5.704344611195E-34 2.614052191228E+00 3.746479760886E+01
2.614052191228E+00 1.334650928015E+07 8.276105949309E+01 8.706470878900E-03 6.872913782734E-01
5.143754364837E-05 2.920832735684E-01 1.464390266467E-05 4.002793748349E-06 5.082554714526E-03
2.301323508453E-07 9.638504967128E-03 6.795246250354E-06 5.827178857934E-03

.....
.....

8.794865539306E-01 9.99999998874E-01 2.361680091400E-06-5.768386555867E-05 2.367155329527-282
-6.926405852191-289 3.012743146671-282-1.572130233500E-28 1.000000000000E+13 1.000000000000E+13
1.000000000000E+13 9.146301617794E+03 4.727352051188E-07 0.000000000000E+00 6.999999992655E-01
8.827001090114E-05 2.799117307236E-01 3.424868374763E-03 4.127760067609E-05 1.059003261745E-03
4.167448298824E-06 9.640129747024E-03 3.902528976547E-06 5.826651038552E-03
8.795031344698E-01 1.000000000000E+00 2.361680091400E-06-5.770215872484E-05 6.993868019057-283
-2.308801950730-289 1.022180710478-282-1.515562617419E-28 1.000000000000E+13 1.000000000000E+13
1.000000000000E+13 8.883218491357E+03 4.555466484484E-07 0.000000000000E+00 6.999999992655E-01
8.827001090114E-05 2.799117307236E-01 3.424868374763E-03 4.127760067609E-05 1.059003261745E-03
4.167448298824E-06 9.640129747024E-03 3.902528976547E-06 5.826651038552E-03
```

## A.4 Fichier ASCII des variables de la zone d'instabilité des Céphéïdes

Le fichier ASCII des variables caractéristiques des modèles de Céphéïdes lors de la traversée de la zone d'instabilité est formé avec l'option PRECISION='ce' du fichier de données `mon_modele .don`, cf. § 4.5 (Page 33). Les quantités enregistrées sont : numéro du modèle(i4), âge(My,es22.15), luminosité(lsol,es10.3), masse(Msol,es10.3), periode(jour,es10.3), rayon(Rsol,es10.3) température effective(K,es10.3). Exemple :

```
356 2.466547805411627E+01 1.259E+04 1.000E+01 1.516E+01 1.084E+02 9.107E+03
357 2.466890876246931E+01 1.254E+04 1.000E+01 1.573E+01 1.111E+02 9.037E+03
.....
```

## A.5 Fichier ASCII d'extension \_TR.csv

Pour faciliter l'exploitation IDL, à l'initiative de T.Guillot, la création du fichier ASCII `mon_modele_TR.csv` a été implémentée, cf. § 4.5 (Page 34). La première ligne permet l'identification des quantités extraites. Exemple :

```
model_num,age_Ma,Mstar_sun,Rstar_sun,Teff_K,lum_sun,Pc_cgs,Tc_K,Roc_cgs,C_tr,Mmt_In,lim,lconv1,lconv2,lconv3,lconv4,l
zc2,r_zc3,r_zc4,r_zc5,m_zc1,m_zc2,m_zc3,m_zc4,m_zc5,ro_zc1,ro_zc2,ro_zc3,ro_zc4,ro_zc5,dv_zc1,dv_zc2,dv_zc3,dv_zc4,dv
0,0.,1.00000,0.878318,5726.22,0.744397,0.154563E+18,0.137357E+08,81.6554,-0.111885E-14,0.130735,2,1,0,0,0,0.143355,
0.,0.981023806234E-01,0.979120175916,0.,0.,0.,61.6498,0.247404,0.,0.,0.,0.281031E-05,0.772181E-05,0.,0.,0.
1,5.00000000000,1.00000,0.879827,5732.36,0.750164,0.156738E+18,0.137284E+08,82.8474,-0.107368E-14,0.129992,2,1,0,0,0,
741824,0.,0.,0.,0.778790202533E-01,0.979041639075,0.,0.,0.,65.3007,0.246481,0.,0.,0.,0.207130E-05,0.840763E-05,0.,0.,
```

```

2,11.0000000000,1.00000,0.880609,5728.61,0.749535,0.156270E+18,0.136800E+08,82.9099,-0.735205E-15,0.130228,2,1,0,0,0,0,0.113
745243,0.,0.,0.,0.525455362860E-01,0.979576762828,0.,0.,0.,69.3036,0.240813,0.,0.,0.,0.135260E-05,0.774815E-05,0.,0.,0.
3,18.2000000000,1.00000,0.882366,5727.15,0.751761,0.156266E+18,0.136408E+08,83.1676,-0.107784E-14,0.130149,2,1,0,0,0,0,0.106
742974,0.,0.,0.,0.439925754963E-01,0.979087194457,0.,0.,0.,70.9699,0.244267,0.,0.,0.,0.123966E-05,0.797560E-05,0.,0.,0.
.....
.....

```

## A.6 Fichier ASCII d'extension `_LB.csv`

De façon similaire, suivant les desiderata de L.Bigot, la création du fichier ASCII `mon_modele_LB.csv` a été implémentée, *cf.* § 4.5 (Page 34). La première ligne permet l'identification des quantités extraites. Exemple :

```

model_num,age_Ma,R_sun,Teff_K,lum_sun,A,dnu02,dnu13,Nu0,PO,X_c,Y,Z/X
0,0.,0.878035,5576.44,0.669084,74.3417,22.3025,37.1708,167.957,102.849,0.709000,0.273000,0.253879E-01
1,5.0000000000,0.879009,5578.57,0.671596,74.4110,22.3233,37.2055,167.683,97.8700,0.708696,0.273000,0.253879E-01
2,11.0000000000,0.879844,5572.51,0.669952,74.0933,22.2280,37.0467,167.437,96.2205,0.708306,0.273000,0.253879E-01
3,18.2000000000,0.881721,5569.55,0.671382,73.8484,22.1545,36.9242,166.909,94.6852,0.707823,0.273000,0.253879E-01
4,26.8400000000,0.883938,5570.68,0.675311,73.6800,22.1040,36.8400,166.293,91.6004,0.707211,0.273000,0.253879E-01
5,37.2080000000,0.885809,5573.39,0.679497,73.4052,22.0216,36.7026,165.777,86.0907,0.706421,0.273000,0.253879E-01
.....
.....

```

# Bibliographie

- Adelberger, E., et al. : Solar Fusion Rates. *Rev. Mod. Phys.* **70**, 1265–1291 (1998)
- Anders, E., Grevesse, N. : Abundances of the elements - Meteoritic and solar. *Geochimica et Cosmochimica Acta* **53**, 197–214 (1989)
- Angulo, C., Arnould, M., Rayet, M., and the NACRE collaboration : A compilation of charged-particle induced thermonuclear reaction rates. *Nuclear Physics A* **656**, 3–183 (1999)
- Asplund, M., CNO in the universe, Proceedings of a conference held in Saint-Luc, Valais, Switzerland, 10-14 September 2002. Edited by Corinne Charbonnel, Daniel Schaerer, and Georges Meynet. *ASP Conference Series*, Vol. 304. San Francisco, CA : Astronomical Society of the Pacific, 2003., p.275
- Asplund, M., Grevesse, N., Sauval, A.J. 2005, in *Cosmic Abundances as Records of Stellar Evolution and Nucleosynthesis*, ed. T.G. Barnes III, & F.N. Bash, *ASP Conf. Ser.*, 336, 25
- Asplund, M., Grevesse, N., Sauval, Scott, P., *ARAA* **47**, 481, 2009
- Basu, S., Antia, H.M. : Helioseismology and solar abundances, *Phys. Rep.* **457**, 217–283 (2008)
- Böhm-Vitense, E. : Über die Wasserstoffkonvektionszone in Sternen verschiedener Effektivtemperaturen und Leuchtkräfte. *Zs. f. Astrophys.* **46**, 108 (1958)
- Canuto, V.M., Mazitelli, I. : Stellar turbulent convection - A new model and applications. *ApJ* **370**, 295–311 (1991)
- Castro, M., Vauclair, S., Richard, O., *A&A* **463**, 755–758, (2007)
- Caughlan, G.R., Fowler, W.A. : Thermonuclear Reaction Rates V. *Atomic Data and Nuclear Data Tables* **40**, 284–334 (1988)
- Christensen-Dalsgaard, J. : Computational procedures for GONG solar model project, *Astronomisk Institut, Aarhus Universitet* (1988)
- Cox, J.P., Giuli, R.T. : *Principles of Stellar Structure*, vol I & II. Gordon & Breach, New-York, (1968)
- De Boor, C. : *A Practical Guide to Splines* Springer, third ed., Berlin (1978)

- Eggleton, P.P., Faulkner, J., Flannery, B.P. : An Approximate Equation of State for Stellar Material, *A&A* **23**, 325 (1973)
- Gabriel M. : Influence of heavy element and rotationally induced diffusions on the solar models. *A&A* **327**, 771–778 (1997)
- Grevesse, N., Noels, A. : Cosmic Abundances of the Elements. In : Prantzos, E. Langioni-flam, M. Casse (eds.) *Origin and Evolution of the Elements*. Cambridge Univ. Press, 14, (1993)
- Grevesse, N., Sauval, A.J. : Standard Solar Composition. *Space Science Revue*, **85**, 161-174 (1998)
- Houdek, G., & Rogl, J. : On the accuracy of opacity interpolation schemes. *Bulletin of the Astronomical Society of India* **24**, 317 (1996)
- Iben, I., *ApJ* **196**, 525, 1975
- Iglesias, C.A., & Rogers F.J. : Opacities for the solar radiative interior. *ApJ* **371**, 408–417 (1991)
- Kippenhahn, R., Weigert, A. : *Stellar Structure and Evolution*, Springer-Verlag Berlin (1991)
- Lang, K.R. : *Astrophysical formulae*, Springer-Verlag Berlin (1968)
- Mathis, S., Zahn, J.P. : Transport and mixing in the radiation zones of rotating stars. I. Hydrodynamical processes, *A&A* **425**, 229–242 (2004)
- Mathis, S., Palacios, A., Zahn, J.P. : On shear-induced turbulence in rotating stars, *A&A* **425**, 243–247 (2004)
- Mihalas D., Dappen, W., Hummer, D.G. : The equation of state for stellar envelopes. II - Algorithm and selected results. *ApJ* **331**, 815–825 (1988)
- Mitler, H.E. : Thermonuclear ion-electron screening at all densities. I - Static solution. *ApJ* **212**, 513–532 (1997)
- Ogata, K., Kan, M., Kamimura, K., astro-ph arXiv :0905.0007v
- Palacios, A., Talon, S., Charbonnel, C., Forestini, M. : Rotational mixing in low mass stars : I Effects of the  $\mu$  gradient in main sequence and sub giant Pop I stars. *A&A* **399** 603–617 (2003)
- Schatzman, E., Praderie, F. : *Les Etoiles*, Ed. InterEditions/Editions du CNRS (1990)
- Talon, S., Zahn, J.P., Maeder, A., Meynet, G. : Rotational mixing in early type stars : the main sequence evolution of a  $9M_{\odot}$  star. *A&A* **322**, 209-217 (1997)
- Waldron, W.L. : Stellar winds : Observational evidence for a hot-cool star connection. In NASA. Goddard Space Flight Center *The Origin of Nonradiative Heating/Momentum in Hot Stars* *A.Ap* **229**, 95-100 (1985)