

Utilisation et Description du Code d'évolution stellaire CESAM2K

Pierre Morel

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

Version 3.11.0, 6 Février 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

I	IMPLANTATION et EXPLOITATION	1
1	La version 2000 de CESAM	1
1.1	La restructuration CESAM4/5 vers CESAM2 <i>k</i>	1
1.2	CESAM2 <i>k</i>	2
1.2.1	PGPLOT	2
1.3	Contenu de la distribution	3
1.4	Installation	3
1.5	Adaptation de la source	4
1.6	Création du module exécutable	5
1.6.1	Utilisation d'un MAKEFILE	5
1.6.2	Fonctions des modules	6
2	Implantation	9
2.1	Premier test de fonctionnement	9
2.2	Mise en forme des données physiques	13
2.2.1	Opacités "Yveline"	13
2.2.2	Opacités pour $Z > 0.1$	14
2.2.3	Opacités HOUDEK	15
2.2.3.1	Implantation simplifiée de B.Pichon	16
2.2.4	Equation d'état OPAL	16
2.2.4.1	OPAL 1991	16
2.2.4.2	OPAL 2001	18
2.2.4.3	OPAL 2005	19
2.2.5	Equation d'état MHD	19
2.2.6	Equation d'état free_eos	20
2.2.7	Equation d'état etat_saha	21
3	Exploitation	25
3.1	Conseils pour l'exploitation	25
3.2	Création d'un espace de travail	25
3.3	Le fichier de données : mon_modele.don	26
3.4	Exemple de fichier de données	26
3.5	Arguments de la NAMELIST NL_CESAM	29
3.6	Arguments de la NAMELIST NL_MASS	33
3.7	Arguments de la NAMELIST NL_EVOL	33
3.7.1	Caractérisation des types de modèles	34
3.8	Arguments de la NAMELIST NL_CHIM	35

3.8.1	Alternative : conservation métal/H ou métal/Z	36
3.9	Arguments de la NAMELIST NL_CONV	37
3.10	Arguments de la NAMELIST NL_DIFF	38
3.11	Arguments de la NAMELIST NL_ROT	39
3.12	Arguments de la NAMELIST NL_ETAT	40
3.13	Arguments de la NAMELIST NL_OPA	41
3.14	Arguments de la NAMELIST NL_NUC	42
3.15	Arguments de la NAMELIST NL_ATM	43
3.16	Exploitation sous UNIX & LINUX	44
3.16.1	Fichiers ASCII et binaires d'initialisation	45
3.16.2	Exemple d'évolution avec PMS	45
3.16.3	Limitation des sorties	48
3.17	Outils astrophysiques	48
3.18	Programmes d'exploitation	50
3.19	Scripts d'exploitation	50
4	Personnalisation	53
4.1	Langue	53
4.2	Mixture	54
4.3	Rapports isotopiques	55
4.4	Personnalisation des rapports d'abondances	56
4.5	Personnalisation de la composition chimique du vent.	56
4.6	Chute de planétoïdes	57
4.7	Personnalisation des paramètres de free-eos	58
4.8	Personnalisation du fichier Ascii	58
4.9	Réglages	60
4.10	Limites des shells source	62
4.11	Zoom	63
4.12	Device	65
4.13	Limitation des écritures	65
5	Extensions	67
5.1	Ajout d'une chaîne de réactions thermonucléaires	67
5.2	Ajout d'une mixture initiale	68
6	Debug	69
6.1	Debug	69
6.2	Bugs connus	70
II	DESCRIPTION DE CESAM2k	75
7	Numérisation de la physique	77
7.1	Intégration par splines-collocation	77
7.1.1	Les B-splines normalisées	78
7.1.2	Résolution d'un problème différentiel	78
7.1.3	L'architecture de CESAM2k	79
7.1.4	Collocation pour un système non linéaire	80

7.1.5	Choix de la base de la collocation	80
7.2	Equations de la structure interne	82
7.2.1	Discontinuités de la densité	83
7.2.2	Variables d'intégration	84
7.2.3	Adaptation de la discrétisation	86
7.2.4	Modification du nombre total de couches	87
7.2.5	Nœud sur une limite zone radiative / zone convective	89
7.3	Restitution de l'atmosphère	91
7.3.1	L'approximation monocouche	91
7.3.2	Reconstitution d'une atmosphère	92
7.3.3	Artifices numériques utilisés pour le raccord de ∇	94
7.3.4	Loi $T(\tau)$ purement radiative	95
7.3.5	Lois $T(\tau)$ avec convection	96
7.3.6	Résolution numérique	97
7.4	Evolution temporelle de l'énergie gravifique	99
7.4.1	L'approximation de Kippenhahn	99
7.4.2	Discrétisation	100
7.4.3	Initialisation	100
7.5	Evolution de la composition chimique sans diffusion	101
7.5.1	Problème raide	102
7.5.2	Résumé des contraintes	103
7.5.3	Les formules IRK Lobatto IIIC.	104
7.5.4	Mélange des éléments chimiques sans diffusion	104
7.5.5	Conservation du nombre de nucléons	106
7.5.6	Conservation des baryons et de la charge	107
7.5.7	Normalisation de la somme des abondances	107
7.5.8	Estimation de la précision de l'intégration	108
7.6	Evolution du moment cinétique sans diffusion	108
7.7	Evolution temporelle avec diffusion	110
7.7.1	Intégration par éléments finis	110
7.7.2	Bases de B-splines pour Petrov-Galerkin	112
7.7.3	Présence de discontinuités	112
7.8	Diffusion des éléments chimiques	113
7.8.1	Condition limite externe	115
7.8.2	Chutes de planétoïdes	116
7.8.3	Notations	116
7.8.4	Charge moyenne des ions	118
7.8.5	Equation de diffusion des espèces chimiques	120
7.8.6	Jacobien.	127
7.8.7	Accélérations radiatives	128
7.9	Diffusion du moment cinétique	129
7.9.1	Changement de variable $M \rightarrow \nu = \left(\frac{M}{M_{\odot}}\right)^{\frac{2}{3}}$	129
7.9.2	Quelques notations	130
7.9.3	Expressions de H_P , H_T , ∇_{μ} , χ etc... et dérivées	130
7.9.4	Les coefficients de diffusion D_h , D_v et D_{eff}	132
7.9.4.1	Formalisme de Mathis, Palacios & Zahn	132

7.9.4.2	Formalisme simplifié de Castro, Vauclair & Richard .	132
7.9.5	Formalisme de Talon et al. (1997)	133
7.9.5.1	Les variables	133
7.9.5.2	Transport du moment cinétique	133
7.9.5.3	Expressions initiales de \check{E}_Ω et \check{E}_μ	133
7.9.5.4	Expression vérifiée par E_Ω	134
7.9.5.5	Expression vérifiée par E_μ	135
7.9.5.6	Expression vérifiée par $U = y_2$	135
7.9.5.7	Expression vérifiée par $\Theta = y_3$	136
7.9.5.8	Fluctuation du poids moléculaire $\Lambda = y_4$	137
7.9.5.9	Expression vérifiée par $\Psi = y_5$	137
7.9.6	Formalisme de Mathis & Zahn (2004)	137
7.9.6.1	Les variables	137
7.9.6.2	Transport du moment cinétique	138
7.9.6.3	Vitesse de circulation méridienne U (Equ. B4)	138
7.9.6.3.1	Expressions dérivées de A (Equ. B5)	139
7.9.6.3.2	Equation 7.83 (Page 138)	139
7.9.6.3.3	Equation 7.84 (Page 138)	139
7.9.6.3.4	Equation 7.85 (Page 139)	140
7.9.6.3.5	Equation 7.86 (Page 139)	140
7.9.6.3.6	Equation 7.87 (Page 139)	140
7.9.6.3.7	Equation 7.88 (Page 139)	141
7.9.6.4	Regroupement	141
7.9.6.5	Relation barocline (Equ. B6)	143
7.9.6.6	Fluctuations du poids moléculaire (Equ. B7)	143
7.9.6.7	Equation de Poisson (Equ. B8)	143
7.9.7	Les conditions physiques dans les zones mélangées	144
7.9.8	Les conditions physiques aux limites	145
7.9.9	Les conditions initiales	146
7.9.10	Pertes / gains de moment cinétique	147
7.10	La convection	148
7.10.1	Critères de convection	148
7.10.2	Calcul du gradient convectif	149
7.10.3	Echelle de temps de la convection	151
7.10.4	Pression turbulente	151
7.10.5	Localisation des limites des zones convectives	152
7.10.6	Extension des zones convectives	152
7.10.7	Mélange convectif	153
7.10.8	Retrait d'un coeur convectif	153
7.10.9	Semi-convection	154
7.10.10	Estimation de la fréquence de Brunt-Väissälä	154
7.11	Les réactions thermonucléaires	158
7.11.1	Abondances initiales	158
7.11.2	Cycle PP simplifié	159
7.11.3	Exemple de réseau nucléaire : cycles PP, CNO et 3α	160
7.11.4	Éléments à l'équilibre	161
7.11.5	Effet d'écran	161

7.11.6	Energie thermonucléaire et neutrinos	161
7.11.7	Equations d'évolution	162
7.12	Equation d'état gaz parfait	164
8	Les routines de physique	167
8.1	Routines génériques	167
8.2	Unités	168
8.3	Routine <code>abon_ini</code>	169
8.4	Routine <code>add_ascii</code>	169
8.5	Routine <code>alecian1</code>	170
8.6	Routine <code>ascii</code>	171
8.7	Routine générique <code>atm</code>	172
8.8	Routine subordonnée <code>base_chim</code>	173
8.9	Routine subordonnée <code>base_rota</code>	173
8.10	Routine <code>cesam</code>	173
8.11	Programmes <code>cesam2k</code> , <code>cesam2k_dbg</code>	175
8.12	Routine <code>chim_gram</code>	175
8.13	Routine générique <code>coeff_rota</code>	175
8.14	Routines <code>coeff_rota3/4</code>	176
8.15	Routine <code>coll_atm</code>	176
8.16	Routine <code>coll_qs</code>	177
8.17	Routine <code>collision</code>	177
8.18	Routine générique <code>conv</code>	178
8.19	Routine <code>conv_a0</code>	179
8.20	Routine <code>conv_cm</code>	179
8.21	Routines <code>conv_cgm_reza</code> , <code>conv_cm_reza</code>	179
8.22	Routine <code>conv_jmj</code>	179
8.23	Routine <code>coulomb</code>	180
8.24	Routines <code>ctes_85</code> , <code>ctes_94</code> , <code>ctes_94m</code>	180
8.25	Routine générique <code>des</code>	180
8.26	Routines <code>des_m</code> , <code>des_r</code>	181
8.27	Routine <code>df_rotx</code>	181
8.28	Fonction <code>dgrad</code>	182
8.29	Routine générique <code>diffm</code>	182
8.30	Routine <code>diffm_br</code>	183
8.31	Routine <code>diffm_mp</code>	184
8.32	Routine générique <code>difft</code>	184
8.33	Routine <code>difft_gab</code>	184
8.34	Routine <code>difft_nu</code>	185
8.35	Routine <code>difft_sun</code>	185
8.36	Routine <code>diffus</code>	186
8.37	Routine générique <code>diffw</code>	187
8.38	Routine <code>diffw_mpz/p03</code>	187
8.39	Routine <code>dnunl</code>	187
8.40	Routine subordonnée <code>ecrit_ascii</code>	188
8.41	Routine subordonnée <code>ecrit_rota</code>	189
8.42	Routine <code>edding</code>	189

8.43	Routine eq_atm	189
8.44	Routine eq_diff_chim	190
8.45	Routine eq_diff_poisson	191
8.46	Routine eq_diff_rota3/4	191
8.47	Routine générique etat	192
8.48	Routine etat_ceff	193
8.49	Routine etat_eff	193
8.50	Routine etat_gong1	193
8.51	Routine etat_gong2	193
8.52	Routine etat_mhd	194
8.53	Routines etat_opal, etat_opalX, etat_opalZ	194
8.54	Routine evol	195
8.55	Routine générique f_rad	196
8.56	Routine subordonnée fcmax	197
8.57	Routine from_alecian	197
8.58	Routine hopf	197
8.59	Routine iben	197
8.60	Routine générique ini_ctes	198
8.61	Routine initialise_rota	198
8.62	Fonction initialise_u	198
8.63	Fonction initialise_w	199
8.64	Routine inter	199
8.65	Routine inter_atm	200
8.66	Routines k5750, k5777	200
8.67	Routine kappa_cond	200
8.68	Routine lim_atm	201
8.69	Routines lim_gong1, lim_tau1	201
8.70	Routine lim_zc	202
8.71	Routine list	203
8.72	Routine lit_binaire	204
8.73	Routine lit_hr	205
8.74	Routine lit_nl	205
8.75	Fonction logique lmix	206
8.76	Routine modif_mix	206
8.77	Routine générique nuc	206
8.78	Routine générique opa	208
8.79	Routine opa_gong	209
8.80	Routine opa_houdek9	209
8.81	Routine opa_int_zsx	210
8.82	Routine opa_opal2	210
8.83	Routine opa_opalC0	211
8.84	Routine opa_yveline	212
8.85	Routine opa_yveline_lisse	212
8.86	Routine osc_adia, osc_invers, osc_noad	213
8.87	Routine générique output	213
8.88	Routine générique pertm	214
8.89	Routine pertm_ext, pertm_msol	214

8.90	Routine <code>pertm_tot</code>	215
8.91	Routine <code>pertem_waldron</code>	215
8.92	Routine générique <code>pertw</code>	215
8.93	Routine <code>pertw_loc</code>	216
8.94	Routine <code>pertw_ptm</code>	216
8.95	Routine <code>pertw_sch</code>	216
8.96	Routine <code>planetoides</code>	217
8.97	Routine <code>poisson_initial</code>	218
8.98	Routine <code>pp1</code>	218
8.99	Routine <code>pp3</code>	218
8.100	Routine <code>ppcno10</code>	218
8.101	Routine <code>ppcno10BeBFe</code>	219
8.102	Routine <code>ppcno10Fe</code>	219
8.103	Routine <code>ppcno10K</code>	219
8.104	Routine <code>ppcno11</code>	220
8.105	Routine <code>ppcno12</code>	220
8.106	Routine <code>ppcno12Be</code>	220
8.107	Routine <code>ppcno12BeBFe</code>	221
8.108	Routine <code>ppcno12Li</code>	221
8.109	Routine <code>ppcno3a12Ne</code>	221
8.110	Routine <code>ppcno3a9</code>	221
8.111	Routine <code>ppcno3aco</code>	222
8.112	Routine <code>ppcno9</code>	222
8.113	Routine <code>ppcno9Fe</code>	222
8.114	Routine <code>print_ctes</code>	222
8.115	Routine <code>read_ascii</code>	223
8.116	Routine <code>resout</code>	223
8.117	Routine <code>resout_chim</code>	225
8.118	Routine <code>resout_rota</code>	226
8.119	Routine <code>resout_rota3/4</code>	226
8.120	Routine <code>rkimps</code>	227
8.121	Routines <code>roger00</code> , <code>roger02</code> , <code>roger05</code> , <code>roger10a</code>	229
8.122	Routine <code>rq_reac</code>	229
8.123	Routine <code>saha</code>	230
8.124	Routine <code>sortie</code>	231
8.125	Routine générique <code>static</code>	231
8.126	Routines <code>static_m</code> , <code>static_r</code>	232
8.127	Routine <code>tabul_nuc</code>	233
8.128	Routine <code>taueff</code>	234
8.129	Routine <code>taux_nuc</code>	234
8.130	Routine générique <code>tdetau</code>	235
8.131	Routine <code>thermo</code>	236
8.132	Routine <code>thermo_atm</code>	237
8.133	Routine <code>trho</code>	238
8.134	Routine <code>update</code>	238
8.135	Routine <code>vent</code>	239
8.136	Routine <code>write_nl</code>	239

8.137	Package z14xcotrin21	240
9	Les routines numériques	241
9.1	Routines numériques et assimilées	241
9.1.1	Routine arb_rom	241
9.1.2	Routine boite	241
9.1.3	Routine box	241
9.1.4	Routine delete_doubles	241
9.1.5	Routine difdiv	242
9.1.6	Routine fermi_dirac	242
9.1.7	Routine gauss_band	242
9.1.8	Routine horner	242
9.1.9	Routine intgauss	242
9.1.10	Routine matinv	242
9.1.11	Routine max_local	242
9.1.12	Routine min_max	242
9.1.13	Routine min_max_cond	242
9.1.14	Routine neville	243
9.1.15	Routine newton	243
9.1.16	Routine pause	243
9.1.17	Routine pgplot_factice	243
9.1.18	Routine plot_rota	243
9.1.19	Routine polyder	243
9.1.20	Routine shell	243
9.1.21	Routine zoning	243
9.2	Routines spécifiques aux B-splines	243
9.2.1	Routine bsp1ddn	244
9.2.2	Routine bsp1dn	244
9.2.3	Routine bsp_dis	244
9.2.4	Routine bsp_gal	244
9.2.5	Routine bval0	244
9.2.6	Routine bval1	244
9.2.7	Routine bvald	244
9.2.8	Routine coll	244
9.2.9	Fonction colpnt	244
9.2.10	Routine left_right	245
9.2.11	Routine linf	245
9.2.12	Routine newspl	245
9.2.13	Routine newspl_gal	245
9.2.14	Routine noedif	245
9.2.15	Routine noein	245
9.2.16	Routine noeu_dis	245
9.2.17	Routine noeud	245
9.2.18	Routine schu58_n	245
9.2.19	Routine sum_n	246

III	APPENDICES	247
A	Installation sous LINUX	249
A.1	Création de la bibliothèque avec <code>ifc</code>	249
A.2	Installation de PGPLOT	250
A.2.1	Installation intégrale avec <code>ifc</code>	250
A.2.2	Installations sans larme	251
A.2.3	CESAM2 <i>k</i> sans PGPLOT	251
A.3	Exemple de MAKEFILE	252
B	Données diverses	261
B.1	Liste du tableau <code>glob</code>	261
B.2	Liste du tableau <code>var</code>	261
B.3	Liste du tableau <code>list_cesam</code>	263
C	Liste des fichiers binaires	265
D	Liste des fichiers ASCII	267
D.1	Fichiers pour oscillations	267
D.1.1	Fichier pour oscillations adiabatiques	267
D.1.2	Fichier pour oscillations non adiabatiques	269
D.1.3	Fichier pour inversions	270
D.1.4	Utilisation du nombre maximum de couches	271
D.2	Fichier pour diagramme HR	272
D.2.1	Fichier simplifié pour diagramme HR	274
D.3	Fichier ASCII des variables de la diffusion du moment cinétique . . .	274
D.4	Fichier ASCII des variables de la zone d'instabilité des Céphéïdes . .	275
D.5	Fichier ASCII d'extension <code>_TR.csv</code>	276
D.6	Fichier ASCII d'extension <code>_LB.csv</code>	276
E	Liste détaillée des modules	277
E.1	Module <code>mod_kind</code>	277
E.2	Module <code>mod_numerique</code>	277
E.3	Module <code>mod_donnees</code>	278
E.4	Module <code>mod_variables</code>	286
E.5	Module <code>mod_etat</code>	290
E.6	Module <code>mod_opa</code>	290
E.7	Module <code>mod_conv</code>	290
E.8	Module <code>mod_atm</code>	291
E.9	Module <code>mod_nuc</code>	292
E.10	Module <code>mod_bp_for_alecian</code>	294
E.11	Module <code>mod_evol</code>	294
E.12	Module <code>mod_static</code>	295
E.13	Module <code>mod_cesam</code>	295
E.14	Module <code>mod_exploit</code>	296

F	Contenu des sous-directory	297
F.1	Sous-directory SOURCE	297
F.2	Sous-directory EXPLOIT	306
F.3	Sous-directory TESTS	308
F.4	Sous-directory SCRIPTS	309
F.5	Sous-directory SUN_STAR_DATA	310

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.	7
2.1	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 § 3.4 (Page 26).	23
6.1	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 72).	70
7.1	B-splines normalisées d'ordre $m=2, 3$ & 4 pour $n=5$ points de grille équidistants.	78
7.2	B-splines normalisées d'ordre $s=2$ (gauche) et $s=3$ (droite) calculées avec la base de C. de Boor pour $n=3$ points de grille équidistants. Les vecteurs nodaux sont schématisés par des croix disposées sous chaque graphe.	78

7.3	Deux itérations ont été nécessaires pour positionner un point de grille sur une limite zone radiative / zone convective. Au cours du processus itératif, le gradient radiatif ∇_{rad} a légèrement diminué, et la limite s'est déplacée vers la droite, alors que le gradient adiabatique ∇_{ad} restait inchangé. Les points de grille se sont successivement déplacés de leurs positions initiales, carrés vides, vers leurs positions finales, cercles vides puis cercles pleins.	89
8.1	Environnement de la routine <code>evol</code> de gestion de l'évolution de la composition chimique et du moment cinétique.	194
8.2	Organigramme de la routine <code>lim_zc</code> de détermination des limites zones radiatives / zones convectives.	202
8.3	Environnement de la routine de calcul des réactions thermonucléaires. La routine <code>nuc</code> est appelée de <code>cesam</code> pour l'initialisation des abondances. Lors d'une évolution, elle est appelée i) de <code>eq_dif_chim</code> ou de <code>rkimps</code> pour le calcul de l'évolution des abondances, suivant qu'il y ait ou non suivi de la diffusion des éléments chimiques, ii) de <code>thermo</code> pour le calcul de l'énergie thermonucléaire libérée, iii) de <code>list</code> pour le calcul des flux de neutrinos. Au premier appel, il y a initialisation des abondances initiales et tabulation des réactions nucléaires suivant le cycle et la compilation des réactions à utiliser, <i>cf.</i> §3.4 (Page 26). Lors des appels suivants, il y a calcul des variations des abondances ou/et de l'énergie thermonucléaire libérée, ou encore des flux de neutrinos.	207
8.4	Organigramme de la routine de la gestion de la résolution des équations de la structure interne <code>resout</code>	223

Liste des tableaux

3.1	Valeurs des paramètres numériques de la precision par défaut (pr/PR) et celles qui en diffèrent suivant les divers types de précision. 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 Céphéïdes 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 § 4.9 (Page 60) et lignes 805-1237 dans la routine cesam du sous-directory SOURCE . L'utilisation des minucules/MAJUSCULES (Exemple : pr , PR) du type de précision définit le type de variables quasi-statiques à utiliser, <i>cf.</i> § 3.5 (Page 29). Une astérisque * signale les paramètres qui sont modifiés lorsque le modèle prend en compte la diffusion microscopique.	32
3.2	Valeurs des paramètres numériques (suite).	33
7.1	Coefficients de Eq. 7.15 (Page 104) pour les formules IRK Lobatto IIIC d'ordre $p = 1, 2, 4$ avec $s = 1, 2, 3$ étapes. On utilise la notation usuelle, les c_i , $i = 1, \dots, s$ sont éléments de la première colonne, les b_i ceux de la dernière ligne, les autres coefficients sont les a_{ij} de la matrice caractérisant la formule de Runge-Kutta.	105

Index

pertm_tot.f, 213
2d-2.pms, 304
5d-4.pms, 304
8d-5.pms, 304

sa, 30

A, 29
A-stable, 101
aal27, 276
ab, 290
ab11, 276
ab_ini, 276
ab_min, 276
abe7, 276
abe9, 276
abon_ini.f, 167, 292, 295
abon_m, 276
abon_rela, 290
abondance d'hélium, 52
abondances, 52, 100, 105
abondances initiales, 33, 52, 156
ac12, 276
ac13, 276
accélération centrifuge, 91, 93,
199
accélération totale, 126
accélération radiatives, 126
adaptation de la grille, 83
add_Al, 52
add_Ar, 52
add_ascii.f, 167, 294, 295
add_B, 52
add_Be, 52
add_C, 52
add_Ca, 52
add_Cl, 52
add_Co, 52
add_Cr, 52
add_F, 52
add_Fe, 52
add_K, 52
add_Li, 52
add_Mg, 52
add_Mn, 52
add_N, 52
add_Na, 52
add_Ne, 52
add_Ni, 52
add_O, 52
add_P, 52
add_S, 52
add_Sc, 52
add_Si, 52
add_Ti, 52
add_V, 52
add_Z, 52
Adelb, 40
af18, 277
af19, 277
afe56, 276
age, 286
age_deb, 291
age_fin, 291
agemax, 31, 277
ah, 277
ah2, 277
ahe3, 277
ahe4, 277
aide mémoire, 3
ajout de mixture, 64
ajout isotope, 63
ajuste, 57, 281
alecian1.f, 35, 168, 293, 295
alecian2.f, 35
ali6, 277
ali7, 277
all_adia, 27
all_ascii, 28

- all_invers, 27
- all_iter, 38
- all_mod, 38
- all_nadia, 28
- all_output, 281
- all_rep, 281
- ALLOCATABLE, 3, 247
- alpha, 35, 277
- amg23, 277
- amg24, 277
- amg25, 277
- amg26, 277
- amu, 277
- an, 277
- an13, 277
- an14, 277
- an15, 277
- ana23, 277
- ane20, 277
- ane21, 277
- ane22, 277
- anglais, 49
- ao16, 277
- ao17, 277
- ao18, 277
- ap, 277
- ap31, 277
- approximation de Kippenhahn, 97
- approximation monocouche, 91, 199
- ar, 290
- aradia, 277
- arb_rom.f, 239, 275
- arb_rom.fasc, 295
- arrêt de l'évolution, 69, 91, 93, 199
- arret, 31, 282
- as32, 277
- ascii.f, 54, 169, 294, 295
- ascii2bin_opa.f, 13
- asi8, 277
- atm.f, 169, 290, 295
- atmosphère, 69, 90, 198
- AV, 28
- av, 30
- B-splines, 74
- baratine, 45, 61
- base, 78
- base_chim.f, 170, 295
- base_rota.f, 171, 295
- bashrc, 4
- be7sbe9, 291
- be7sz, 291
- blabla, 45, 61, 304
- blas, 20
- boite.f, 239, 275, 295
- box.f, 239, 275, 295
- bp, 284
- bp_atm, 289
- bp_atm_t, 289
- bp_t, 284
- Brunt-Väissälä, 69, 152
- bspiddn.f, 242, 275, 295
- bsp1dn.f, 242, 275, 295
- bsp_dis.f, 242, 275, 295
- bsp_gal.f, 242
- bugs, 66
- Burgers, 118
- bval0.f, 242, 275, 295
- bval1.f, 242, 275, 295
- bvald.f, 242, 275, 296
- c, 291
- C-shell, 3
- c13sc12, 291
- c_iben, 286
- calib2k_.f, 304
- calib2k_pms, 48, 307
- calib2k_pms.f, 47
- calib2k_zams, 48, 307
- calib2k_zams.f, 47, 304
- calibration solaire, 34
- Cau-Fow, 40
- CE, 28
- cephede, 29, 273
- cesam.f, 171, 294, 296
- cesam2k_dbg.f, 296
- cesam2k.f, 172, 296
- cesam2k.out, 2, 9, 42
- cesam2k_dbg.f, 172
- cesamT_dbg.f, 296
- cesamT.f, 296
- charges, 167
- chim, 284

- chim_gram.f, 172, 288, 296
 chim_t, 284
 clight, 277
 co, 30
 coca, 32
 coeff_rota.f, 173, 293, 296, 305
 coeff_rota3.f, 293, 296
 coeff_rota3/4.f, 173
 coeff_rota4.f, 293, 296
 coeff_vth.f, 296
 coeur convectif, 151
 cohe, 32
 col_atm.f, 296
 col_qs.f, 296
 coll.f, 242, 275, 296
 coll_atm.f, 173, 290
 coll_qs.f, 174, 293
 collision.f, 175, 293, 296
 colpnt, 275
 colpnt.f, 242, 296
 compilateur ifc, 247, 248
 compilateur INTEL, 248
 compilateurs, 2
 compilation, 3
 compile2k, 48, 307
 compile2k-dbg, 48, 307
 compile2k-dbg_list, 48, 307
 compile2k_list, 48, 307
 conditions limites, 80, 85, 114,
 229
 cone, 32
 cons_glob_mnt_cin, 37
 cons_loc_mnt_cin, 37
 conseils, 23
 conservation, 104
 conservation du moment cinétique,
 106
 constantes physiques, 166
 contraintes, 101
 conv, 175
 conv.f, 288, 296
 conv_a0.f, 147, 176, 288, 296
 conv_cgm, 41
 conv_cgm_reza.f, 148, 176, 288,
 296
 conv_cm.f, 148, 176, 288, 296
 conv_cm_reza.f, 148, 176, 288, 296
 conv_cml.f, 288, 296
 conv_jmj.f, 147, 177, 288, 297
 convd, 292
 convection, 175
 convergence, 100
 convergence forcée, 69
 convf, 292
 coox, 32
 coulomb.f, 177, 293, 297
 cpturb, 35, 277
 create_evol_space, 23
 cshrc, 3, 249
 ctel, 57, 277
 ctem, 57, 277
 ctep, 57, 277
 cter, 57, 277
 ctes, 177
 ctes_31, 27
 ctes_85.f, 27, 177, 283, 294, 297
 ctes_94.f, 27, 177, 283, 294, 297
 ctes_94m.f, 27, 177, 283, 294, 297
 ctes_ba, 27
 ctes_gaia, 27
 ctet, 57, 278
 cycle CNO, 157
 cycle PP, 157
 cycle3 α , 157
 définition de la ZAMS, 42
 d_conv, 276
 d_grav, 57, 68, 278
 d_turb, 35, 278
 d_conv, 111
 Debian, 249
 delete_doubles.f, 239, 275, 297
 delfim, 289
 delfip, 289
 dell, 13
 des.f, 3, 177, 294, 297
 des2k_abon.f, 47, 304
 des2k_abontc.f, 47, 304
 des2k_abonts.f, 47, 305
 des2k_bin.f, 47, 305
 des2k_dhve.f, 47
 des2k_diff_osc, 305
 des2k_diff_spl, 305
 des2k_diff_spl.f, 47

- des2k_grad, 305
- des2k_grad.f, 47
- des2k_hr.f, 47, 305
- des2k_opa.f, 47, 305
- des2k_osc.f, 47, 305
- des2k_rot.f, 305
- des2k_rot_ext.f, 305
- des2k_vaiss.f, 305
- des2k_ZC.f, 47, 304
- des_Krot, 280
- des_m.f, 4, 27, 178, 294, 297
- des_mC, 27
- des_r.f, 3, 4, 27, 178, 294, 297
- dessin, 13
- device, 60, 283, 305
- df_rotx, 167
- df_rotx.f, 178, 288, 297
- dgrad.f, 179, 293, 297
- dh, 280
- difdiv.f, 240, 275, 297
- diff_mz04, 37
- diff_tz97, 37
- difficultés, 66
- diffm.f, 179, 293, 297
- diffm_0, 35
- diffm_br.f, 35, 180, 293, 297
- diffm_mp.f, 35, 180, 293, 297
- diff_t.f, 181, 293, 297
- diff_t_gab.f, 35, 181, 293, 297
- diff_t_nu.f, 35, 182, 293, 297
- diff_t_nut.f, 35
- diff_t_smc.f, 35, 297
- diff_t_sun.f, 182, 293, 297
- diffus.f, 183, 293, 297
- diffusion, 35, 142, 281
- diffusion du moment cinétique, 187, 188
- diffusion microscopique, 35
- diffusion microscopique, 102, 111, 186
- diffusion turbulente, 35
- diffusivité horizontale, 130
- diffusivité verticale, 130
- diffw, 294
- diffw.f, 183, 293, 298
- diffw_0.f, 36
- diffw_mpz.f, 36, 293, 294, 298
- diffw_mpz/p03.f, 184
- diffw_p03.f, 36, 293, 294, 298
- diffw_toul.f, 37
- dim_ch, 286
- dim_qs, 286
- dim_rot, 286
- discontinuité de la densité, 78, 152
- discontinuités fossiles, 151
- divergence, 68
- dl, 280
- dlpp_atm, 289
- dn_fixe, 199, 278
- dnunl.f, 184, 294, 298
- données, 2, 13
- dp, 275
- dpsi, 57, 278
- dr_zc, 292
- dt0, 278
- dt_planet, 291
- dtlist, 31, 278
- dtmax, 57, 278
- dtmin, 57, 276
- dws, 36
- dx_tams, 276
- echarg, 278
- ecran faible, 159
- ecrit_ascii.f, 184, 298
- ecrit_rota.f, 185, 293, 298
- edding.f, 41, 94, 186, 290, 298
- effet d'écran, 159
- éléments à l'équilibre, 158
- elem, 291
- element trace, 180
- elements finis, 108
- en_masse, 57, 281
- endevol, 38
- end_mod, 38
- energie cinétique, 106
- energie gravifique, 98
- energie thermonucléaire, 159
- enhan_al, 33
- enhan_cha, 33
- enhan_w, 33
- eps, 292
- eq_atm.f, 186, 290, 298

- eq_diff_chim.f, 186, 293, 298
- eq_diff_poisson.f, 187, 293
- eq_diff_rota.f, 188
- eq_diff_rota3.f, 293, 298
- eq_diff_rota4.f, 293, 298
- eq_ini_rota4.f, 298
- equation d'état, 39
- equation d'état etat_gp, 161
- equations d'évolution, 80, 85, 160
- equations de Burgers, 180
- equations de la structure interne, 80
- espace de travail, 23
- eta.f, 298
- etat, 188
- etat.f, 188, 288
- etat_ceff.f, 189, 288, 298
- etat_eff.f, 39, 189, 288, 298
- etat_gong1.f, 39, 190, 288, 298
- etat_gong2.f, 39, 190, 288, 298
- etat_irwin.f, 20, 38, 54
- etat_irwin_explik, 38
- etat_mhd.f, 19, 39, 190, 288, 298, 308
- etat_opal.f, 16, 18, 39, 190, 288, 298
- etat_opalX.f, 18, 39, 190, 288, 298
- etat_opalZ.f, 18, 39, 190, 288, 298
- eve, 278
- evol.f, 191, 293, 298
- evol2k_pms, 48, 307
- evol2k_zams, 48, 307
- evolution temporelle, 99
- exe2k, 3, 48, 307
- exe2k_dbg, 48, 307
- EXPLOIT, 2
- extension .hom, 43
- extension .mix, 50
- extension .pms, 43
- extension .zams, 43
- f037_2k.f, 47, 305
- f_eos, 38, 283
- f_opa, 39, 283
- f_rad.f, 193, 293, 298
- fac, 293
- facteurs de répartition, 86
- fcmax, 194
- fcmax.f, 299
- fcv, 58
- fermi_dirac.f, 240, 275, 299
- fesh_des, 280
- fesh_sol, 278
- fichier de données, 24
- fichier de sorties ASCII, 27
- fichier mon_modele_B.pms, 195
- fichier_vent.f, 47, 305
- fichiers d'initialisation, 42
- fichiers d'oscillations, 265
- fonction d'espacement, 86
- fonction de répartition, 86
- forces radiatives, 126
- FORTTRAN95, 2
- free_eos, 20
- from_alecian.f, 194, 299
- g, 278
- gain de masse, 52
- garde_xish, 33, 34, 281
- gauss_band.f, 240, 275, 299
- genere_bases.f, 275
- genere_cesam2k, 3, 48, 307
- genere_cesam2k_dbg, 48, 308
- glob, 259
- gmsol, 278
- Gnome, 249
- grad_ovi, 281
- grad_ovs, 281
- granr, 278
- gravité, 126
- grille équidistante, 84
- grille adaptative, 86
- grille_fixe, 33, 191, 281
- h, 280
- h2sh1, 291
- He, 28
- he3she4, 291
- he3she4z, 291
- He_ajuste, 281
- he_core, 31, 278
- HHe, 27, 58

- dhe_core, 278
 hopf.f, 41, 94, 194, 290, 299
 horner.f, 240, 275, 299
 hpl, 278
 hsra.f, 41, 299

 i3al, 291
 i_ex, 280
 Iben, 99
 iben.f, 195, 292, 299
 id_conv, 286
 idis, 292
 if_conv, 286
 ifc, 248
 ife56, 280
 ihe4, 166, 280
 iLi7, 280
 indices spécifiques, 166
 ini0, 57, 280
 ini_ctes.f, 166, 195, 283, 294, 299
 initialise_poisson.f, 293
 initialise_rota.f, 195, 293, 299
 initialise_rota4.f, 299
 initialise_u.f, 195, 293
 initialise_w.f, 196, 293
 intégrales de collision, 175
 intégration numérique, 73
 integrales, 299
 inter.f, 196, 288, 299
 inter_atm.f, 197, 294, 299
 intgauss.f, 240, 275, 299
 Ipg, 280
 Ipgt, 289
 IRK Lobatto IIIC, 102
 iter_qs, 58
 izz, 291

 jacob_rota.f, 299
 jacobien, 161
 jlim, 286
 jlim_t, 286
 journal, 2, 299
 jours, 36
 jpz, 35, 150, 282

 k5750.f, 41, 95, 197, 290, 299
 k5777.f, 41, 95, 197, 290, 299

 kappa_cond, 288
 kappa_cond.f, 197, 299
 kbol, 278
 kind, 6
 kipp, 57, 282
 kms/s, 36
 knot, 286
 knot_ptm, 286
 knot_t, 286
 knot_tds, 287
 knot_tds_t, 287
 knot_temp, 291
 knotc, 286
 knotc_t, 286
 knotr, 286
 knotr_t, 286
 Krot, 280
 Kubuntu, 249

 L, 29
 10, 58
 l_demi, 57
 l_des, 280
 langue, 49, 282, 305
 lapack, 20
 lbol0, 278
 lconv, 287
 lconv_t, 287
 ld, 280
 Ledoux, 35, 146, 179, 199
 ledoux, 282
 left_right.f, 243, 299
 lhe_stop, 287
 li6sli7, 291
 li_ini, 278
 lib_del, 308
 lib_del_repl, 48, 308
 lib_repl, 48, 308
 libcesam2k.a, 5
 lim, 287
 lim_atm.f, 41, 198, 290, 299
 lim_gong1.f, 41, 198, 290, 300
 lim_gong2.f, 300
 lim_jpz, 282
 lim_ro, 41, 282
 lim_t, 287
 lim_tau1.f, 41, 198, 290

- lim_zc.f, 199, 293, 300
- limitation des écritures, 61
- limitation des sorties, 45, 61
- limite externe, 170
- limites zones radiatives / zones convectives, 68
- linf.f, 243, 275, 300
- link, 15
- LINUX, 2, 3, 42
- lisse, 57, 282
- list.f, 201, 294, 300
- list_cesam, 5, 260, 300
- lit_binaire.f, 202, 294, 300
- lit_hr.f, 202, 294, 300
- lit_nl.f, 203, 283, 294, 300
- lit_osc.f, 68
- LM, 28
- lm, 28, 30
- lmix.f, 203, 293, 300
- ln10, 278
- ln_Tli, 278
- lnt_stop, 278
- loc_zc, 57, 278
- log_teff, 31, 278
- logl_max, 280
- logl_min, 280
- logteff_max, 280
- logteff_min, 280
- lsol, 278
- lt_stop, 287
- ltaue, 289
- ltauf, 289
- lw_perte, 293
- lx_stop, 288

- m, 290
- m010.zams, 305
- m020.zams, 305
- m050.zams, 305
- m23, 285
- m23_t, 285
- mélange convectif, 151
- météorites, 33
- métal/H, 34
- métal/Z, 34
- méthode de collocation, 76
- méthode de Galerkin, 76
- méthode de résolution, 73
- m_atm, 289
- m_ch, 57, 280
- m_ptm, 57, 280
- m_qs, 57, 280
- m_rot, 57, 281
- m_tds, 57, 281
- m_temp, 290
- m_zc, 285
- m_zc_t, 285
- majuscules/minuscules, 29
- makefile, 249, 307, 308
- Mandrake, 249
- MARCS, 41
- MARCS.f, 41
- marcs.f, 300
- masses atomiques, 167
- matinv.f, 240, 275, 300
- max_local.f, 240, 275, 300
- mc, 284
- mc_fixe, 284
- mc_t, 284
- mct, 284
- mct_t, 284
- mdot, 29, 278
- me, 279
- methode, 283
- mg25smg24, 291
- mg26smg25, 291
- min_max.f, 240, 275, 294, 300
- min_max_cond.f, 240, 294, 300
- mitler, 39, 159, 282
- mix, 293
- mixture, 50, 64, 305
- mod_atm.f, 6, 289, 300
- mod_bp_for_alecian.f, 292, 300
- mod_cesam.f, 3, 6, 293, 300
- mod_conv.f, 6, 288, 300
- mod_donnees.f, 6, 166, 276, 300
- mod_etat.f, 6, 288, 300
- mod_evol.f, 6, 292, 300
- mod_exploit.f, 6, 294, 300
- mod_kind.f, 6, 275, 300
- mod_nuc.f, 6, 290, 300
- mod_numerique.f, 6, 275, 301
- mod_opa.f, 6, 288, 301
- mod_static.f, 6, 293, 301

mod_variables.f, 6, 284, 301
modèle de PMS, 195
modèle de ZAMS, 68
model_num, 287
modif_chim, 33, 282
modif_mix, 33, 305
modif_mix.f, 204, 301
modif_xish, 52
moment cinétique, 106
moment d'inertie, 271
mon_modele-ad.osc, 27, 265
mon_modele-ascii, 27
mon_modele-inv.osc, 27, 268
mon_modele-nad.osc, 27, 267
mon_modele.ceph, 29, 273
mon_modele.don, 2, 24, 305
mon_modele.HR, 270
mon_modele.LR, 272
mon_modele.modif_mix, 305
mon_modele.rap_iso, 306
mon_modele.vent, 306
mrot, 284
mrot_t, 284
mrott, 284
mrott_t, 284
msol, 279
mstar, 286
mstar_t, 286
mterre, 279
mtot, 29, 279
mu_saha, 57, 282
mvt_dis, 282
mw_tot, 286
mw_tot_t, 286
mzc_ext, 291

n15sn14, 291
n23_atm, 289
n_atm, 57, 281
n_ch, 287
n_ch_t, 287
n_max, 26, 281
n_min, 276
n_mix, 293
n_ptm, 287
n_qs, 287
n_qs_t, 287

n_rot, 287
n_rot_t, 287
n_tds, 287
n_tds_t, 287
n_temp, 291
nœud sur une limite ZR/ZC, 87
NACRE, 40
NACRE+, 40
NACRU, 40
nb_max_modeles, 31, 281
nb_modeles, 287
nc_fixe, 287
nchim, 281
ndis, 292
ne, 281
ne22sne20, 291
ne_atm, 289
nelem_ini, 290
neutrinos, 159
neville.f, 241, 275, 301
new_bv, 58, 153
newspl.f, 243, 275, 301
newspl_gal.f, 243, 301
newton.f, 241, 275, 301
niso_tot, 290
NL_ATM, 24, 41, 97
NL_CESAM, 24, 26
NL_CHIM, 24, 33, 156
NL_CONV, 24, 35, 68, 147, 150
NL_DIFF, 24
NL_ETAT, 24, 38
NL_EVOL, 31
NL_MASS, 24, 29
NL_MODIF_MIX, 52
NL_NOMS, 27
NL_NUC, 24, 39, 159
NL_OPA, 24, 39
NL_RLG, 56
NL_ROT, 24, 36
NL_TEMPS, 24, 69
no_croiss, 275
no_des, 27, 38, 178
no_dis, 57
no_frad, 35
no_output, 27
noedif, 243
noedif.f, 243, 275, 301

- noein.f, 243, 275, 301
- noeu_dis.f, 243, 275, 301
- noeud.f, 243, 275, 301
- nom_abon, 24, 33, 283
- nom_atm, 24, 41, 282, 290
- nom_chemin, 24, 26, 283
- nom_conv, 24, 283
- nom_ctes, 24, 27, 283
- nom_des, 24, 283
- nom_des_rot, 37
- nom_diffm, 24, 35, 283
- nom_difft, 24, 35, 283
- nom_diffw, 38, 283
- nom_elem, 282
- nom_etat, 24, 39, 283
- nom_fich2, 283
- nom_frad, 35, 283
- nom_nuc, 24, 39, 283
- nom_nuc_cpl, 39, 40, 283
- nom_opa, 24, 39, 283
- nom_opa_cond, 39
- nom_output, 27, 283
- nom_perte, 24
- nom_pertm, 29, 283
- nom_pertw, 38, 283
- nom_qs, 293
- nom_rot, 282
- nom_tdetau, 24, 41, 283
- nom_thw, 37, 283
- nom_xheavy, 282
- nombre électrons libres, 158
- nombre de couches, 269
- nombre total de couches, 85
- normalisation, 106
- notations, 81
- NOTICE, 2, 3
- NP, 28
- np, 28, 68
- nreac, 291
- nreac_tot, 290
- nrl, 281
- nrot, 281
- nu_min, 292
- nuc.f, 204, 292, 301
- nuc_gong.f, 157
- nucleo, 276
- numéro de version, 2
- numérotation des modèles, 27, 32
- nuzc_ext, 291
- nzc, 292
- o17so16, 291
- o18so16, 291
- object module, 5
- old_ptm, 284
- opa.f, 206, 288, 301
- opa_compton.f, 288, 301
- opa_cond_iben, 39
- opa_cond_mestel, 39
- opa_gong.f, 39, 207, 288, 301
- opa_houdek.f, 309
- opa_houdek9.f, 15, 39, 207, 288, 301
- opa_int_zsx.f, 39, 208, 288, 301
- opa_opal2.f, 208, 288, 301
- opa_opal2_cno, 39
- opa_opal2_co, 39
- opa_opalC0.f, 39, 209, 288, 301
- opa_yveline.f, 39, 210, 288, 301, 308
- opa_yveline_jorgen.f, 39
- opa_yveline_lisse.f, 39, 211, 288, 301
- opacité, 13, 39, 206
- opacité conductive, 39
- opalX_ascii_bin.f, 18
- opalZ_ascii_bin.f, 18
- optimisation, 69
- option mixture, 50
- ord_qs, 281
- ordre, 57, 281
- ordre des éléments, 157
- osc_adia.f, 27, 211, 294, 301
- osc_invers.f, 27, 211, 294
- osc_invers.f], 302
- osc_nadia.f, 27, 294, 302
- osc_noad.f, 211
- output.f, 211, 294, 302
- overshoot, 150
- ovshti, 35, 150, 279
- ovshts, 35, 150, 279
- p_atm, 289
- p_pertw, 38, 279
- pas temporel, 101

- pause.f, 241, 275, 302
- personnalisation, 49
- perte de masse, 52
- pertm.f, 212, 293
- pertm.f], 302
- pertm_ext.f, 29, 212, 293, 302
- pertm_msol.f, 29, 212, 293, 302
- pertm_tot.f, 29, 293, 302
- pertm_waldron.f, 29, 213, 293, 302
- pertw.f, 213, 293, 302
- pertw_0, 38
- pertw_loc.f, 38, 214, 293, 302
- pertw_ptm.f, 38, 214, 293, 302
- pertw_sch.f, 38, 214, 293, 302
- Petrov-Galerkin, 110
- PGPLOT, 2, 3, 248, 249, 307
- pgplot_factice, 249, 302
- pgplot_factice.f, 241
- phi, 276
- pi, 279
- piau1/3, 41
- planétoïdes, 53, 114
- planet, 306
- planetoides.f, 215, 292, 302
- plot_reac_nuc.f, 306
- plot_rota.f, 241, 302
- PMS, 43, 99, 195
- pnzc, 276
- point de collocation, 80
- POINTER, 3, 247
- poisson_initial.f, 215
- polyder.f, 241, 275, 302
- post, 32
- pp1.f, 39, 216, 292, 302
- pp3.f, 39, 216, 292, 302
- ppcno10.f, 39, 216, 292, 302
- ppcno10BeBFe.f, 39, 216, 292, 302
- ppcno10Fe.f, 39, 217, 292, 302
- ppcno10K.f, 39, 217, 292, 302
- ppcno11.f, 39, 217, 292, 303
- ppcno12.f, 39, 218, 292, 303
- ppcno12Be.f, 39, 218, 292, 303
- ppcno12BeBFe.f, 39, 218, 292, 303
- ppcno12Li.f, 39, 218, 292, 303
- ppcno3a12Ne.f, 39, 219, 292, 303
- ppcno3a9.f, 39, 219, 292, 303
- ppcno3aco.f, 39, 219, 292, 303
- ppcno3acos.f, 39
- ppcno9.f, 39, 219, 292, 303
- ppcno9Fe.f, 39, 220, 292, 303
- PR, 28
- pr, 28, 30
- précision de l'intégration, 106
- precision, 28, 29, 282
- precit, 57, 279
- precix, 57, 279
- pression turbulente, 68, 80, 149
- print_ctes.f, 220, 303
- problème raide, 100
- psi0, 57, 286
- pt_atm, 289
- pturb, 282
- q, 285
- q0, 57, 290
- q-t, 285
- qt, 285
- qt-t, 285
- r2, 285
- r2_t, 285
- réactions thermonucléaires, 39, 63, 157
- réglages, 55
- résolution de problèmes différentiels, 75
- r_atm, 289
- r_ov, 285
- r_ov_t, 285
- r_qs, 276
- R_stop, 31
- r_zc, 285
- r_zc_conv, 285
- r_zc_t, 285
- raccord des ∇ , 93
- rad, 290
- rad/s, 36
- rap_iso, 306
- rapport des abondances, 52
- rappports isotopiques, 51, 167
- rayon bolométrique, 92
- rayon observable, 91
- re_nu, 35, 279
- read_ascii.f, 220, 294, 303

- reconstitution de l'atmosphère,
 - 91, 95
- RedHat, 249
- reglages, 306
- reml2k_mod, 48, 308
- reml2k_mod-dbg, 48, 308
- renormalisation, 191
- rep_atm, 282
- resout.f, 222, 293, 303
- resout_chim.f, 223, 293, 303
- resout_rota.f, 224, 293, 303
- resout_rota3.f, 224, 293
- resout_rota4.f, 224, 293
- restitution de l'atmosphère, 41
- restructuration, 1
- rg, 28, 30, 56
- rkimps.f, 225, 226, 293, 303
- ro_test, 57, 279
- roger.f, 303
- roger00.f, 41, 95, 226, 290, 308
- roger02.f, 41, 95, 226, 290
- roger05.f, 41, 95, 226, 290
- roger10a.f, 41, 95, 226, 290
- rot_0, 37
- rot_cte, 37
- rot_min, 276
- rot_solid, 282
- rota, 284
- rota_t, 284
- routine de dessin, 27
- routine de sorties ASCII, 27
- routines dépendantes, 165
- routines de convection, 175
- routines de physique, 165
- routines disponibles, 26
- routines génériques, 165
- RPM, 249, 307
- rq_reac.f, 227, 292, 303
- rsol, 279
- rstar, 286

- SA, 28
- sa, 28
- saha.f, 116, 227, 288, 303
- schu58_n.f, 243, 275, 303
- Schwarzschild, 146, 179, 199
- SCRIPTS, 2

- secon6, 279
- semi convection, 68
- semi-convection, 152
- semi_convection, 35
- shell.f, 241, 275, 303
- shells source, 58
- sigma, 279
- singularité des gradients, 82
- solaire_ag3_03, 33
- solaire_ag3_05, 33
- solaire_gn, 33
- solaire_gs, 33
- sortie.f, 228, 288
- sortie_ascii, 306
- sorties ASCII, 27
- SOURCE, 2
- source, 283
- SP, 28
- sp, 28, 30, 275
- stabilité, 100
- static.f, 229, 293, 303
- static_m.f, 229, 293, 303
- static_r.f, 229, 293, 303
- suffix V, 35
- suffixe de convection S, 35
- suffixe de précision A, 32
- suffixe de precision G, 29
- suffixe de precision N, 29
- suffixe de precision O, 29
- suffixes de dessin C, HHe & S, 27
- suffixes de output A, T & L, 28
- suffixes de précision A, L & M,
 - 28
- sum_n.f, 244, 275, 303
- SUN_STAR_DATA, 2, 13
- superconvergence, 78
- SuSE, 249

- t_ajuste, 282
- t_atm, 289
- t_gab, 292
- t_inf, 279
- t_stop, 31, 279
- t_sup, 279, 291
- tab_vth.f, 304
- tabul_nuc.f, 166, 167, 231, 292,
 - 304

- tau, 289
- tau_max, 41, 279
- tau_min, 289
- taueff.f, 231, 290, 304
- taux_nu.f, 292
- taux_nuc.f, 232, 304
- taux_reac, 290
- tdetau.f, 232, 290, 304
- tds, 284
- TdS varie trop, 66
- tds_t, 284
- teff_des, 280
- temp, 290
- temps de convection, 149
- test_acc_rad.f, 306
- test_atm.f, 306
- test_bsp_dis.f, 306
- test_cesam.f, 306
- test_coef_diff.f, 306
- test_coeff_rota.f, 306
- test_collision.f, 306
- test_colloc.f, 306
- test_convection.f, 306
- test_data_ceff.f, 306
- test_data_eff.f, 306
- test_der_rotx.f, 306
- test_etat/2.f, 306
- test_jacobien_reac_nuc.f, 306
- test_lim_ZC.f, 306
- test_mu_mol.f, 306
- test_nl.f, 306
- test_opa.f, 306
- test_opa_opal2.f, 307
- test_opacite.f, 307
- test_read_osc.f, 307
- test_rkimps.f, 307
- test_saha.f, 307
- test_tabul_reac.f, 307
- test_tdetau.f, 307
- test_thermo.f, 307
- test_thermo_atm.f, 307
- TESTS, 2
- thermo.f, 233, 293, 304
- thermo_atm.f, 234, 290, 304
- thw, 283
- tot_conv, 288
- tot_rad, 288
- trho.f, 235, 290, 304
- ttemp, 290
- turbulence rotationnelle, 130
- ua, 279
- un_eps, 292
- unit, 36, 282
- unités, 165
- UNIX, 2, 3, 42
- update.f, 235, 293, 304
- var, 259
- variable indépendante, 80
- variable lagrangienne, 80
- variables, 78
- variables d'intégration, 81
- vecteur nodal, 79
- vent.f, 52, 114, 226, 236, 292, 304, 306
- version, 276
- vitesse de convection, 149
- W_FORM, 144
- w_form, 279
- w_rot, 36, 279
- WINDOWS, 2
- write.nl.f, 236, 294, 304
- wrot, 286
- wrot_t, 286
- x0, 33, 279
- x_ajuste, 282
- x_atm, 289
- x_atm_t, 289
- x_mix, 292
- x_planet, 285
- x_ptm, 285
- x_stop, 31, 279
- x_tams, 276
- x_tds, 285
- x_tds_t, 285
- xcoll, 293
- xl, 285
- xleft, 280
- xt_atm, 289
- xt_ptm, 285
- xt_tds, 285
- xt_tds_t, 285

xvent, 276

y0, 33, 279

y_age, 280

ybot, 280

z0, 279

z14xcotrin21.f, 237, 304

ZAMS, 9

zams, 32

zi, 276

zone de vent, 114

zoning.f, 241, 275, 304

zoom, 59, 306

zsx0, 33, 279

zsx_sol, 279

Première partie

**IMPLANTATION et
EXPLOITATION**

Chapitre 1

La version 2000 de CESAM

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

Le Tartuffe.

1.1 La restructuration CESAM4/5 vers CESAM2*k*

En vue d'extensions futures, une restructuration¹ de CESAM est apparue nécessaire. Elle est articulée autour d'une réécriture du code en FORTRAN95 qui permet le regroupement des routines en MODULEs². Elle concerne les points suivants :

- dépeussierage et fiabilisation des algorithmes,
- suppression de variables et de routines obsolètes,
- amélioration de la lisibilité, en particulier par l'utilisation de tableaux à plusieurs dimensions qui, grâce aux MODULEs, sont gérés de façon plus fiable et surtout plus simple par FORTRAN95 que par FORTRAN77,
- exploitation sous forme "générique" des routines de physique, afin d'éviter les arborescences d'EXTERNALs et ainsi de faciliter et fiabiliser la mise en œuvre des extensions et des modifications ultérieures,
- utilisation de facilités de programmation offertes par FORTRAN95 par exemple : allocation dynamique des tableaux, boucles infinies, opérations sur les tableaux, gestion des chaînes de caractères, etc...
- CESAM2*k* est un abrégé de CESAM2000.
- introduction des variables d'espace $\ln P$, $\ln T$, R , L , $M^{\frac{1}{3}}$ assurant la possibilité de modèles ayant une luminosité localement négative résultant de pertes d'énergie importantes par neutrinos.

On trouvera § 1.1 (Page 7) un organigramme général du code.

1. Ce paragraphe concerne principalement les utilisateurs habitués aux versions précédentes de CESAM.

2. Un module est une sorte de "super COMMON". Un des intérêts de cette structure réside dans le fait que le compilateur vérifie la concordance des types, (`real`, `integer`...), du nombre, des dimensions des arguments des listes d'appel des routines entre le programme appelant et la routine appelée.

1.2 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 §2.1 (Page 9). Il n'est donc pas nécessaire de créer un exécutable adapté à chaque utilisation³. Il est toutefois possible de modifier la plupart des options, numériques ou physiques retenues, grâce à des fichiers dits de "réglage" décrits au §4 (Page 53). Ils permettent d'adapter CESAM2*k* aux moyens de calcul dont on dispose et, dans une large mesure, aux particularités physiques des modèles à calculer.

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) :

- sous LINUX : `ifc`, `ifort` de INTEL, Lahey-Fujitsu `lf95`,
- sous UNIX : compilateur UNIX des machines ALPHA, `ifc`, `ifort`
- 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.

1.2.1 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 centrale⁴, 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. 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 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.

3. Sauf, bien sûr, pour des modifications, debug, *etc...*

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

1.2.2 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` effectue l'installation sans difficulté.

Il est aussi possible d'éviter l'utilisation de dessin "on line", *cf.* § A.2.3 (Page 251).

1.3 Contenu de la distribution

Le package CESAM2k⁵ contient les sous-directory :

- `EXPLOIT` : ensemble de programmes et d'exemples de fichiers pour l'exploitation.
- `NOTICE` : sources `LATEX` et fichiers postscript de la notice d'utilisation et de l'aide-mémoire.
- `SCRIPTS` : ensemble de procédures UNIX/LINUX pour l'installation et l'exploitation avec les shells `csh` et `bash`.
- `SOURCE` : ensemble des routines, modules et programmes de la source et des `makefiles` d'installation.
- `SUN_STAR_DATA` : ensemble de données physiques, essentiellement des packages, des programmes pour l'exploitation de tables d'EOS et d'opacités.
- `TESTS` : divers programmes de test⁶.

1.4 Installation

On décrit l'installation sous UNIX/LINUX. Il n'est pas exclus 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.

PB Avec un compilateur dont la version n'est pas à jour de la norme F95, (comme des versions anciennes du compilateur d'INTEL⁷) il y aura refus d'effectuer une allocation dans une routine autre que celle où est déclarée l'allocation. Il y a lieu de remplacer dans les routines concernées, les instructions `ALLOCATABLE` par `POINTER` pour les tableaux incriminés *cf.* § A.1 (Page 249).

Installation :

- Détarer la distribution avec, par exemple :


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

5. 2k pour "2000".

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

7. Contrairement aux idées reçues, ce compilateur n'est pas gratuit, *cf.* le contrat de licence.

Le sous-directory CESAM2k est créé⁸. Il contient 6 sous-directory tarés et compressés :

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

Il est recommandé de détarer ces directory au fur et à mesure des besoins. Cette disposition permet de personnaliser les endroits du système d'exploitation les mieux adaptés pour placer ces sous-directory et ainsi de faciliter les mises à jour qui, la plupart du temps, n'affectent que le seul sous-directory SOURCE.

- Les deux sous-directory SCRIPT_BASH et SCRIPT_CSH du directory SCRIPT, contiennent des scripts d'exploitation qu'il convient d'adapter au shell et au compilateur F95 utilisés. Les scripts des procédures de compilation sont respectivement `compile2k` et `exe2k`. Indiquer aussi les chemins de la source et des bibliothèques de données physiques qui seront utilisées. Rendre les procédures exécutables : `chmod 700 *`.
- Pour une utilisation avec le shell `csh` ou `tcsh`, dans le fichier `.cshrc` ajouter les chemins de la bibliothèque et des exécutables, par exemple :

```
set mypath=(.... ~/CESAM2k/SOURCE ~/CESAM2k/EXPLOIT .....)
set path=(. ~ ~/SCRIPTS $path $mypath) .....
```

 Avec le shell `bash`, dans le fichier `.bashrc`, ces instructions deviennent :

```
PATH="$PATH:~/CESAM2k/SCRIPTS:~/CESAM2k/EXPLOIT:~/CESAM2k/SOURCE"
export PATH
```
- Si ce n'est déjà fait, installer le logiciel de dessin PGPLOT⁹, et indiquer le chemin de sa bibliothèque dans le fichier `.cshrc` :

```
LD_LIBRARY_PATH **** :/usr/lib/pgplot
```

 et définir la variable d'environnement¹⁰ :

```
setenv PGPLOT_DIR /usr/lib/pgplot.
```

 On donne au § A.2.1 (Page 250) un supplément d'informations concernant l'installation de ce logiciel.
 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 est peut être incomplète.

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

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

10. Une de ces instructions est peut être redondante. L'avis d'experts est accepté.

1.5 Adaptation de la source

Avant de créer le module exécutable, il convient d'adapter certains paramètres libres de la source à l'installation dont on dispose.

- 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`, 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.* § 4.12 (Page 65).
- La source contient plusieurs `Makefile` dans lesquels sont définis les paramètres de compilation pour 2 compilateurs avec et sans debug. Ces makefiles sont dérivés d'une version initiale créée par Th. Corbard. La procédure de construction de la bibliothèque `libcesam2k.a`, et de l'exécutable `cesam2k.out` qui est commune aux différents makefile est contenue dans le fichier `core.mak` introduit par un `include` dans le makefile au cours de son exécution. Suivant le compilateur utilisé, le codage à utiliser, le niveau d'optimisation souhaité, il convient d'adapter la formulation des règles du Makefile, exemples :
 - règle de compilation `IFORT = lf95`,
 - règle d'extension `T2k = 2k`
 - règle de debug `DBG = $(T2K)-dbg`
 - règle de link `FLAG2K = -c -O3`

1.6 Création du module exécutable

Se placer dans le sous-directory `SOURCE`, soit :

- saisir `make -Bf Make2k`, puis `cesam2k.out`,
- lancer la procédure¹¹ `genere_cesam2k`.

Après la compilation, on obtient la liste des "object module" de la bibliothèque `libcesam2k.a` et le menu d'entrée de CESAM :

```
mod_kind.o
mod_numerique.o
mod_donnees.o
mod_variables.o
mod_etat.o
mod_opa.o
mod_conv.o
mod_atm.o
mod_nuc.o
mod_bp_for_alecian.o
mod_evol.o
mod_static.o
```

11. Dans la suite le symbole `<==` indique une instruction à saisir.

```

mod_cesam.o
mod_exploit.o
Encountered 0 errors, 0 warnings in file cesam2k.f.

```

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
 ARRET

Suivant l'installation et les paramètres de compilation retenus, la création de la bibliothèque et du module exécutable nécessitent de l'ordre de 1 à 3 minutes.

On peut créer un exécutable de debug¹² avec `make -Bf Make2k-dbg` ou la procédure `genere_cesam2k-dbg` du sous-directory `SCRIPTS`.

1.6.1 Fonctions des modules

Les modules de `CESAM2k` ont les fonctionnalités suivantes :

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

12. L'argument `-B` est nécessaire si les dates de modification des routines correspondent à une compilation avec une optimisation différente.

- 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.
- Module `mod_exploit` : regroupe des routines concernées par l'exploitation des résultats.

On trouvera §1.1 (Page 7) un organigramme général du code. Chaque routine `PRIVATE` et/ou `PUBLIC` est introduite sous la forme d'`INCLUDE` dans son module d'appartenance, le nom de ce dernier indiqué dans des commentaires au début de chaque routine.

ORGANIGRAMME GÉNÉRAL

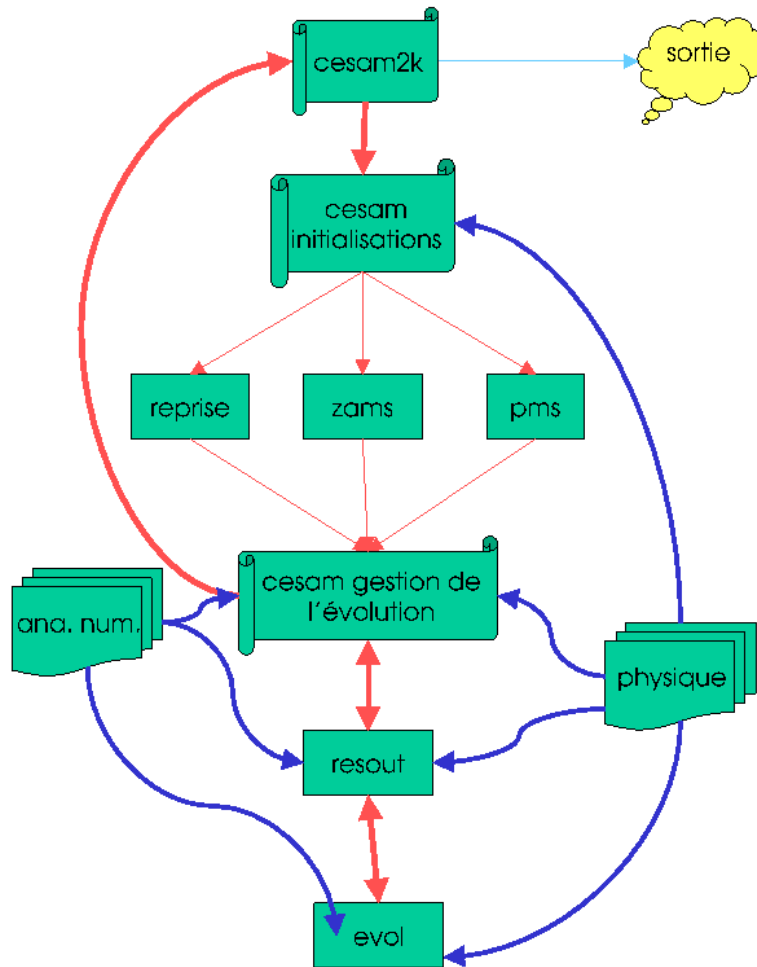


FIGURE 1.1 – Schéma de l'organigramme général de CESAM2k. Le programme **cesam2k**, compilé une fois pour toutes, fait appel à la routine **cesam** qui constitue en fait, le programme principal. Dans **cesam**, après des initialisations, on sépare le traitement des 3 possibilités : i) poursuite d'une 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.

Chapitre 2

Implantation

2.1 Premier test de fonctionnement

Toutes les données physiques utilisées dans ce test de fonctionnement sont implantées dans la SOURCE.

Saisir "create_evol_space TEST" et se placer dans le sous-directory TEST créé.

Alternative : copier les fichiers mon_modele.don et m010.zams du sous-directory EXPLOIT dans le sous-directory TESTS dans lequel on se placera.

- Remplacer "des_m" par "no_des" si PGPLOT n'est pas opérationnel.
- Adapter "nom_chemin" au chemin du sous-directory SUN_STAR_DATA de votre installation.
- Établir les instructions suivantes :

```
&NL_CESAM
NOM_CHEMIN='~/SUN_STAR_DATA/',    <==== à adapter
NOM_CTES='ctes_94',
NOM_DES='des_m',    <==== à adapter
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='difft_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.
/

```

Lancer 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, modele_euler

```

```
mon_modele <==
```

```
identificateur des fichiers du modèle : mon_modele
```

```
*****
```

```
MODELE DE STRUCTURE INTERNE calculé par CESAM2k version : V3.7.24
```

Début du calcul le : 16 Décembre 2012 à 09h37

NAMELISTS du fichier: mon_modele.don

&NL_CESAM NOM_CHEMIN=~ /SUN_STAR_DATA/

.....

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

âge= 5.000E+03, LogTeff= 3.761E+00, LogL/Lsol=-2.454E-02, LogR/Rsol=-1.147E-02
 Log g= 4.461E+00, Pc= 2.309E+17, Tc= 1.545E+07, Roc= 1.504E+02, Xc= 3.435E-01
 ePP/eNUC= 99%, eCNO/eNUC= 1%, e3a+C+O/eNUC= 0%, eGRAV/eNUC= 0%, Yc= 6.359E-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= 4.084E-02day)

Fin d'évolution avec CESAM2k version : V3.11.2

Sortie car agemax atteint

.....

nom du fichier du modèle d'atmosphère en binaire : mon_modele_B.atm

nom du fichier ASCII zones de pulsation des Céphéides : mon_modele.ceph

nom du fichier ASCII H et He shell sources : mon_modele.mHHe

Le code d'évolution stellaire CESAM a été élaboré

dans le cadre du Groupement de Recherche Structure Interne

des Etoiles et des Planètes Géantes. Si son utilisation

vous a donné satisfaction, le but poursuivi par tous

ceux qui y ont contribué aura été atteint.

P.Morel, ON. Décembre 1989, CESAM1

P.Morel, OCA. Octobre 1991, CESAM2

P.Morel, OCA. Avril 1993, CESAM3

P.Morel, OCA. Décembre 1997, CESAM4

P.Morel, OCA. Décembre 2002, CESAM5

P.Morel, B.Pichon OCA. Septembre 2003, CESAM2k

P.Morel, OCA, Y.Lebreton, MJo Goupil OBSPM, Février 2004, Version anglaise

P.Morel, OCA, A.Moya, OBSPM, Mars 2005, diffusion du moment cinétique

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.Renault, 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'évolution à partir de la ZAMS. Au § 3.16.2 (Page 45) 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'adapter les dimensions des cadres dans les routines `des_m.f` et `des_r.f` du sous directory `SOURCE`, cf. § 1.5 (Page 4). On recherchera les valeurs optimales en utilisant le fichier "device" qui permet de personnaliser le dessin, cf. § 4.12 (Page 65).

2.2 Mise en forme des données physiques tabulées

L'opacité et l'équation d'état ne sont, le plus souvent, connues que sous 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 dépend du matériel utilisé. Il y a lieu de les créer au fur et à mesure des besoins. On décrit ci-après la mise en forme de ces données regroupées dans le sous-directory `SUN_STAR_DATA` dans lequel on se placera.¹

PB En raison de l'infini diversité des possibilités, il est incontournable de devoir effectuer des calculs avec des tables d'opacité ou d'équation d'état, dont les abondances en éléments lourds Z seront différentes de celles souhaitées pour le modèle.

2.2.1 Opacités `opa_yveline.f`, `opa_yveline_lisse.f`

Ces deux routines utilisent les mêmes données.

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

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

Effectuer la transformation ASCII ==> binaire, en exécutant³ le programme `ascii2bin_opa` :

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

3. Avec F95 il est pratique d'utiliser les procédures du sous-directory `SCRIPTS`.

```
exe2k ascii2bin.opa
```

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 <== à adapter!
```

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.

2.2.2 Opacités opa_opal2.f

Les opacités OPAL de type 1 n'existent que pour une abondance des métaux inférieure à $Z < 0.1$. Le package `z14xcotrin21` d'Arnold I. Boothroyd qui, gérant les opacités OPAL de type 2, permet de tenir compte de la nucléosynthèse des éléments lourds. Bien que son implantation dans `CESAM2k` ne recouvre pas toutes les situations envisagées, son *USAGE est DELICAT*. Lors de la première utilisation, le système peut demander de décompresser des routines du sous-directory `SUN_STAR_DATA`.

2.2.3 Opacités opa_houdek12.f

Dans cette version de CESAM2*k*, 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 et 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>

Après détarage et décompression : tar -xzvf opint_v12.tar.gz le module obtenu gh_int.v12 a été placé⁴ dans le sous-directory SUN_STAR_DATA sous le nom H12.

Les éléments de ce module dans lesquels il a été nécessaire d'intervenir sont :

doc --> notes_v12.pdf, lib, Makefile, OPINPATH

On trouve dans notes_v12.pdf les directives pour la compilation et l'exploitation de ce module.

lib contient l'ensemble des routines nécessaires à la construction des tables et à leur exploitation.⁵

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 makefile⁶. 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
```

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

4. en remplacement de l'ancien module HOUDEK

5. 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

6. 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.

donc sage de 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 exécutable 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. Ces fichiers seront utilisés par la routine `opa_houdek12.f` lors des appels à l'opacité. Il est nécessaire de compléter les chemins de ces fichiers. Pour ce faire, remplacer les `"/` par le chemin du fichier H12 par exemple :

```
/home/pierre/SUN_STAR_DATA/H12/op05.GN93/little_endian/opalxe.bin
```

Pour construire les tables d'interpolation pour d'autres associations de données d'opacité il est nécessaire d'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. Il est possible d'indiquer ces tables dans la saisie de l'appel à la construction du Makefile ainsi qu'il est indiqué dans le mode d'emploi cf. `notes_v12.pdf`.

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

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 <==
```

2.2.5 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. § 3.8 (Page 35).

- Avec $Z \neq 0$, se placer dans le sous-directory `SUN_STAR_DATA`, et y créer le fichier d'équation d'état correspondant à la métallicité requise en 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.

2.2.5.1 Equation d'état etat_opal5.f

Equation d'état similaire aux précédentes. La construction de la table pour une valeur de métallicité Z donnée est effectuée dans le sous directory (*à décompresser*) OPAL2005_tables.gz du directory SUN_STAR_DATA. La table est créée par les deux programmes ZFS_interp_EOS5 et opal5Z_ascii_bin dont on formera les exécutable. 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 adaptée à CESAM2k par V.A. Baturin & A.V. Oreshina du Sternberg Astronomical Institute, Lomonosov Moscow State.

2.2.6 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 SUN_STAR_DATA sous les noms mhd1.tab.gz ... mhd8.tab.gz. Se placer dans le sous-directory SUN_STAR_DATA, décompresser ces tables en ASCII : `gzip mhd* <==` et la source du programme fmttob : `gzip fmttob.f.gz`. Transformer les tables en binaire en exécutant `fmttob <==`

```

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

```

```
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 <==
```

2.2.7 Equation d'état `etat_irwin.f`

Equation d'état "analytique" de Irwin. Suivant le système d'exploitation utilisé il est possible de l'implanter de différentes façons dans *CESAM2k*. Le mode d'emploi qui suit est essentiellement dû à Y.Lebreton.

Un préalable est de vérifier si les bibliothèques d'algèbre linéaire `lapack` et `blas` sont déjà implantées. Sinon, sous LINUX, les installer avec `apt-get install lapack3-dev blas3-dev` ou utiliser un processus équivalent.

Implémentation de `free_eos` :

1. Du site <http://freeeos.sourceforge.net>, télécharger le fichier source de l'EOS : `free_eos-2.2.1.tar.gz`, éventuellement une version plus récente. La façon la plus directe d'implanter `free-eos` dans *CESAM2k* est de :
 - (a) Détarer le fichier : `tar -xzvf free_eos-2.2.1.tar.gz`,
 - (b) se placer dans le fichier `free_eos-2.2.1/src`,
 - (c) copier la bibliothèque `libeos.a` dans le directory `SUN_STAR_DATA` :
`cp libeos.a /SUN_STAR_DATA/libeos.a`.
2. Si `libeos.a` n'est pas reconnue on peut :
 - (a) Créer cette bibliothèque en suivant la procédure des `MAKEFILEs` décrite dans le `README` du fichier `free_eos-2.2.1`.
 - (b) Créer la bibliothèque en utilisant le compilateur à disposition :

- Dans le sous-directory `SUN_STAR_DATA` se placer dans `ETAT` après l’avoir détaré `tar -xzvf ETAT.tar.gz`.
- 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`.

Un autre type d’implantation 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. § 3.19 (Page 50), et/ou dans les Makefiles indiquer le chemin des bibliothèques `liblapack`, `libblas`, `libeos` :

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

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 § 4.7 (Page 58) et, pour son exploitation, § 3.12 (Page 40).

2.2.8 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.tar.gz` du directory `SUN_STAR_DATA` 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).

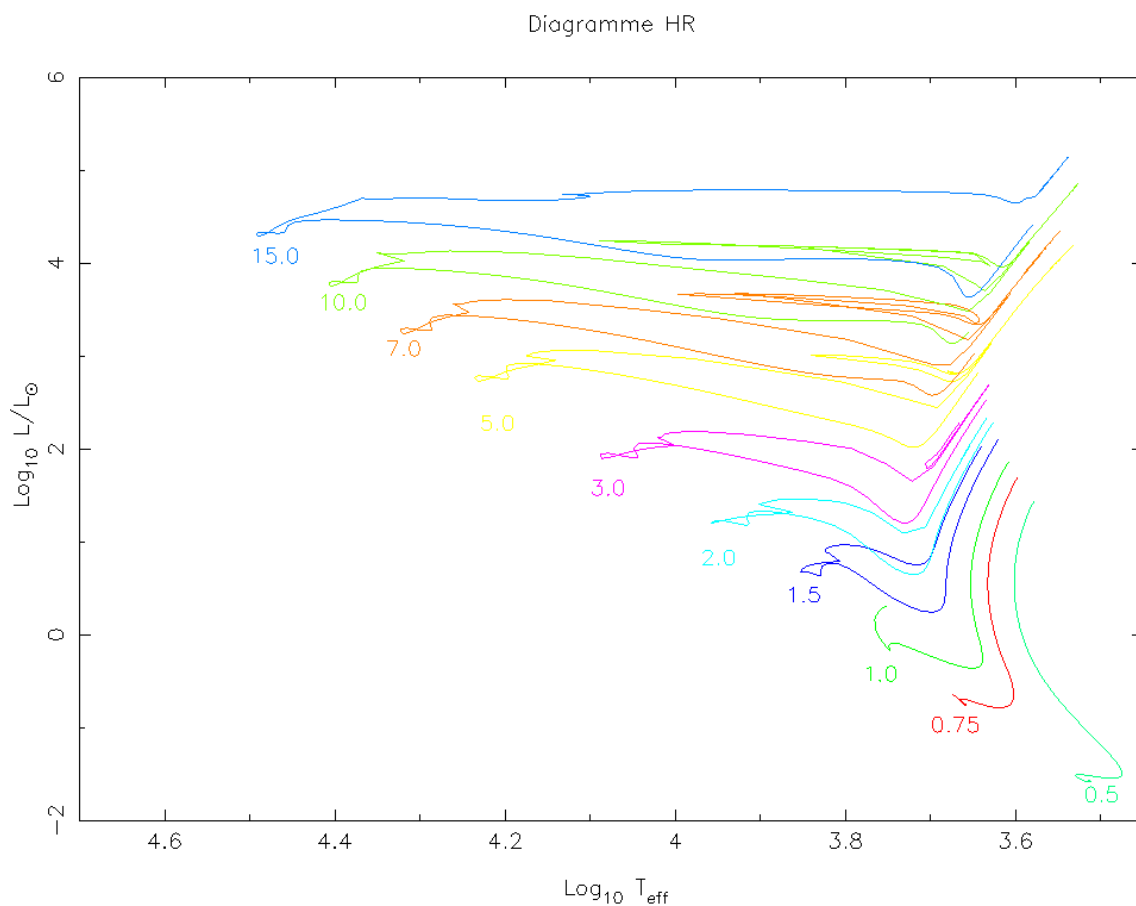


FIGURE 2.1 – Tracés évolutifs incluant la pré-séquence principale pour des étoiles de $0.5M_{\odot}$ à $15M_{\odot}$. Le flash de l'hélium a été dépassé pour les masses supérieures à $3M_{\odot}$. Pour les modèles de $3M_{\odot}$ et de $5M_{\odot}$ les calculs ont été arrêtés après la disparition du 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 § 3.4 (Page 26).

Chapitre 3

Exploitation

Va despacio, estoy de prisa.

Proverbe Mexicain.

(Je vais doucement car je suis pressé)

3.1 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 des routines pour plus d'informations.

3.2 Création d'un espace de travail

Pour chaque 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 de w-virginis, on se placera dans un directory qu'on nommera, pour l'exemple, W_VIR.

1. Saisir `create_evol_space W_VIR`, les données générales (modèles initiaux etc..) sont alors copiés de `CESAM2k/EXPLOIT` dans l'environnement.
2. Se placer dans le directory `W_VIR` : `cd W_VIR`.
3. 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.f` de la SOURCE les arguments reconnus par la présente version `CESAM2k`. Ils sont détaillés ci-après.
4. 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. § 4.12 (Page 65). Une fois les réglages satisfaisants, les reproduire dans le fichier `device` du directory `CESAM2k/EXPLOIT` afin de les récupérer à chaque fois qu'un nouvel espace de travail est créé.

3.3 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.

3.4 Exemple de fichier de données

Cet exemple est volontairement différent de celui utilisé au § 2.1 (Page 9).

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='~/SUN_STAR_DATA/',
NOM_CTES='ctes_94',
NOM_DES='des_m',
NOM_OUTPUT='no_output',
N_MAX=6000,
PRECISION='pr'
/
&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=700,
HE_CORE=-0.1d0,
R_STOP=0.9d0,
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='difft_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_yveline',
F_OPA='opa_yveline.bin','H12/OPINTPATH',6*' '
NOM_OPA_COND=' '
/
&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"¹ permettant différentes actions, Exemple : avec le suffixe M de l'item PRECISION de la NAMELIST NL_CESAM le calcul sera effectué avec un nombre de couches fixé à la valeur maximale N_MAX indiquée dans le fichier de données. Exemple : PRECISION='prM'

S'il y a reprise ou poursuite d'une évolution, CESAM2*k* accepte que des suffixes apparaissent/disparaissent des NAMELISTs, leurs actions seront alors prises en compte/ignorées dans l'évolution ultérieure.

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.

1. Toujours en "upper case".

3.5 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.f` des sous-directories 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/SUN_STAR_DATA/'`.
- `NOM_CTES` : indique le nom de la routine d'initialisation des principales constantes de physique à utiliser² :
 - `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 § 4.11 (Page 63), pratique lorsqu'on désire atteindre des conditions particulières, par exemple une localisation dans le diagramme HR.

Suffixes :

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. § 4.10 (Page 62). Exemples : `des_mHHe`, `des_rHHe`.

- `no_des` : sans dessin *on line*, à utiliser si on ne dispose pas du logiciel PGPLOT, voir § 1.4 (Page 3),
- `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,

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

- `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 `mon_modele-ascii` décrite au § 4.8 (Page 58), de nom
- `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 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. § D.5 (Page 276), cf. § D.6 (Page 276) **Les suffixes A & T sont incompatibles.**

R Le suffixe R (Ex : `no_outputR`) permet de créer un fichier simplifié pour diagramme HR cf. § D.2 (Page 272); il ne contient que l’âge, L et T_{eff} . Ce fichier a pour extension `.LR`, Ex : `mon_modele.LR`, cf. § D.2.1 (Page 274) .

- `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 3.1 (Page 32), 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 : 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. § 3.7 (Page 33).

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. § 3.5 (Page 29).

Les types de précision implantés dans **CESAM2k** sont ³ :

- '**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/He**' précisions pour des évolutions vers les stades avancés. '**ce/CE**' (Céphéïdes) 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.cephe**, cf. § D.4 (Page 275).

Suffixe :

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

- '**he/He**' (flash de l'hélium) identique à '**av/AV**', le pas temporel minimal pouvant descendre au niveau de l'heure.
- '**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 § 4.9 (Page 60) 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. § 4.9 (Page 60).

3.6 Arguments de la NAMELIST NL_MASS

- **MTOT=1.d0** : masse totale initiale, en M_{\odot} ,
- **NOM_PERTM** : nom de la routine de perte de masse :
 - **permt_dominik** : perte de masse selon l'article "stationary dust-driven winds", de Dominik & al. 1990 A&A, 240, 365
 - **permt_ext** : perte/gain de masse externe linéaire en fonction du temps.
 - **permt_msol** : perte/gain de masse externe linéaire en fonction du temps, la masse de l'étoile restant au moins égale à $1 M_{\odot}$,
 - **permt_tot** : perte/gain de masse externe, linéaire en fonction du temps et tenant compte de la perte de masse due aux réactions thermonucléaires ($E = mc^2$),

3. Les "/" indiquent une alternative.

TABLE 3.1 – 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 Céphéïdes 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 § 4.9 (Page 60) 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.* § 3.5 (Page 29). 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.

	<code>pr/PR</code>	<code>sp/SP</code>	<code>lm/LM</code>	<code>av/AV</code>	<code>np/NP</code>	<code>co/CO</code>	<code>sa/SA</code>
<code>m_qs</code>	2				1		
<code>m_ch*</code>	2	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>	0.1	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 3.2 – 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	.TRUE.		.FALSE.		.FALSE.		

- `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).
- `MDOT=1.d-14` : taux de perte/gain de masse, en M_{\odot}/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 plus externe du modèle. Voir § 4.5 (Page 56) comment personnaliser, si besoin, cette composition chimique.

3.7 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'⁴), 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`

4. 'tams' est accepté.

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 $\log_{10} T_{\text{eff}}$ est 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 9999. 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. § 3.5 (Page 29), le pas temporel est ajusté de façon à obtenir les valeurs requises.

3.7.1 Caractérisation des types de modèles

- `'zams'` : Un modèle décrivant la pré-séquence principale est identifié comme modèle de ZAMS dès que la génération d'énergie d'origine nucléaire surpasse celui d'origine graviphique.
- `'post'` : 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.
- `'cohe'` : Un modèle décrivant la post-séquence principale est identifié comme modèle avec combustion d'hélium dès que la température centrale dépasse 1.10^8K .
- `'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 7.10^8K .
- `'cone'` : Un modèle décrivant la séquence de combustion de carbone est identifié comme modèle avec combustion du néon dès que la température centrale dépasse 1.10^9K .
- `'coox'` : Un modèle décrivant la séquence de combustion du néon est identifié comme modèle avec combustion d'oxygène dès que la température centrale dépasse 1.910^9K .

3.8 Arguments de la NAMELIST NL_CHIM

- GRILLE_FIXE=.TRUE. : on utilisera une grille fixe pour la composition chimique et la diffusion du moment cinétique⁵,
- NOM_ABON : nom de la mixture initiale⁶ :
 - enhan_al : abondances α -enhanced de Allard,
 - enhan_cha : abondances α -enhanced de Chaboyer,
 - enhan_w : abondances α -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. (2009),
 - meteorites_agss : abondances météoritiques de Asplund et al. (2009)
 - mixture : forme personnalisée de la compilation d’abondances décrite au §4.2 (Page 54),
 - 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. (2009)
 - solaire_agss : abondances météoritiques de Asplund et al. (2009).
 Ces abondances sont initialisées dans la routine `abon_ini`.
- MODIF_CHIM=.FALSE. : si, dans l’environnement, existent des fichiers des types `modif_mix`, `rap_iso`, `planet` ou `vent` qui permettent, respectivement, de modifier la composition chimique initiale, *cf.* §4.4 (Page 56), les rapports isotopiques, *cf.* §4.3 (Page 55), la composition chimique des planétoïdes, *cf.* §4.6 (Page 57) ou celle du vent, *cf.* §4.5 (Page 56), CESAM2k invitera à supprimer ces fichiers avant d’effectuer les calculs. 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. Il convient de coder `MODIF_CHIM=.TRUE.` pour tenir compte des dispositions offertes par la présence de ces fichiers.
- 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,
 - si $ZSX0 \leq 0$, l’abondance initiale d’hydrogène est initialisée à $X0$, $Z = 1 - X - Y$ 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 $X0$ est ignorée.

La valeur initiale de X utilisée dans le calcul est fractionnée en les divers isotopes retenus pour décrire l’évolution de l’hydrogène.

5. Inactivée dans la présente version de CESAM2k.

6. *cf.* équation d’état OPAL : WEB site http://www-phys.llnl.gov/V_Div/OPAL/opal.html.

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, CESAM2*k* offre divers outils pour réaliser cet ajustement :

- Conserver le rapport Z/X de la mixture initiale, au lieu de ZSX0, objet du § 3.8.1 (Page 36),
- utiliser une mixture originale, *cf.* § 4.2 (Page 54),
- modifier les rapports isotopiques, *cf.* § 4.3 (Page 55),
- modifier les rapports d'abondances de la mixture initiale, *cf.* § 4.4 (Page 56).

3.8.1 Alternative : conservation métal/H ou métal/Z

On désigne par *mixture de base* la mixture proposée par CESAM2*k*, Exemple `solaire.gn`, ou celle fournie par l'utilisateur *cf.* § 4.2 (Page 54). Elle inclut éventuellement les aménagements décrits aux *cf.* § 4.4 (Page 56), ou encore *cf.* § 4.3 (Page 55).

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.* Y_0 .

`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)⁷.

Ainsi, en codant⁸ :

- `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 Y_0 et de ZSX0 correspondent **exactement** à celles de la mixture 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 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, les rapports métal/Z de la mixture de base et le Y_0 du fichier de données.

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 CESAM2*k* correspondent bien à ce que l'on désire.

7. Calcul effectué dans la routine `abon.ini.f`.

8. De meilleures dénominations seraient, par exemple : `GARDE_XISH_MIX`, `GARDE_XISH_MIX.INI`, `GARDE_ZSX` ou encore `GARDE_ZSX_MIX.INI`, *etc...*

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

3.9 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 δ ,
- conv_jmj : convection MLT avec $l = \alpha H_p$ (Böhm-Vitense, 1958).

Suffixe :

S : CESAM2*k* ne cherchera pas à placer les limites ZR/ZC sur des points de grille. Exemple : conv_jmjS. Option par défaut pour les stades avancés.

- alpha=1.7d0 : longueur de mélange,
- ovshts=0.05d0 : coefficient d'overshoot supérieur, avec une valeur positive il y a 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. § 3.10 (Page 38).

PB L'utilisation d'un paramètre de pression turbulente non nul est délicate.

3.10 Arguments de la NAMELIST NL_DIFF

- DIFFUSION=.TRUE. : on tiendra compte de la diffusion microscopique des éléments chimiques, éventuellement de la diffusion du moment cinétique,
- 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 :
 - `diff_t_nu` : calcul des coefficients de diffusion turbulente incluant la diffusivité radiative.
 - `diff_t_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).
 - `diff_t_gab` : calcul des coefficients de diffusion turbulente selon une prescription simple de M.Gabriel.
 - `diff_t_smc` : calcul des coefficients de diffusion turbulente incluant la semi-convection et la diffusivité radiative.
 - `diff_t_sun` : coefficients de diffusion turbulente, sous la zone convective-solaire suivant Gabriel (1997).

Quand le suffixe "V" est accolé à la fin du nom de la routine de diffusion turbulente à utiliser, Exemple : `diff_t_nuV`, le coefficient de diffusion permettant le mélange convectif est pris égal à $1/3V_{conv}L_{ml}$.

- `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,
 - `alecian1` : calcul des accélérations radiatives suivant le premier formalisme de G.Alécian,
 - `alecian2` : calcul des accélérations radiatives suivant le second formalisme de G.Alécian⁹,
 - `no_frad` : les accélérations radiatives seront ignorées.

3.11 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.

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

- 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 : ($D_{\text{eff}} = 300$, $D_h = 1\,000\,000$, $D_v = 250$).
 - 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 Ω^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 cinétique¹⁰.
- 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.* § D.3 (Page 274). 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

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

modèle. Tous les fichiers ainsi créés sont conservés.

- `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.

3.12 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 § 2.2.5 (Page 19).
 - `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`, est la plus rapide et la moins précise. Cette équation d'état utilise la bibliothèque dont la création est décrite § 2.2.6 (Page 20). *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 § 2.2.4 (Page 16).
 - `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. § 2.2.4 (Page 16). 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. § 2.2.4 (Page 16).

- `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. § 2.2.4.3 (Page 19). 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. § 2.2.7 (Page 21)
- `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`.

3.13 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_houdek9` : opacités de Houdek version 9, (OPAL+Alexander), interpolation par rational B-spline; utilise les tables en binaire construites dans le sous-directory `SUN_STAR_DATA/V9`, voir § 2.2.3 (Page 15). Les opacités Houdek contiennent 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. § 2.2.2 (Page 14)
 - `opa_opalCO` : 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 § 2.2.1 (Page 13).
 - `opa_yveline_jorgen` : identique à la précédente pour les opacités Asplund et al. (2009) 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 § 2.2.1 (Page 13).
- `F_OPA='opa_yveline.bin', '/HOUDEK/OPINTPATH_AX95',6*' '` : noms des fichiers d'opacité (au plus 8).
- `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)
 - `' '` : On utilise pas d'opacité conductive

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 selon Cox & Giuli (1968).

3.14 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 α , 9 éléments, H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,80MK]..
 - ppcno3a12Ne : réactions PP+CNO+3 α +carbone, 12 éléments dont Ne22, H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,800MK].
 - ppcno3aco : réactions PP+CNO+3 α +carbone+néon+oxygène, 17 éléments H2, Li7, Be7 à l'équilibre, intervalle de tabulation :[1MK,3GK].
 - ppcno3acos : réactions PP+CNO+3 α +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 3α selon Ogata et al. (2009).
- **NACRU** : Compilation de NACRE updatée.
- **MITLER=.TRUE.** : effet d'écran selon Mitler (1997).

3.15 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 tables¹¹ 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 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. Domaine couvert $T_{\text{eff}} / \log g$:

T_{eff}	$\log g$	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

11. Codifiées MARCS_Zsx.yy, Exemple : MARCS_Z-2.00.

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

3.16 Exploitation sous UNIX & LINUX

Sous UNIX/LINUX, il est conseillé¹² 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 ξ Hya ($3M_{\odot}$) de la ZAMS jusqu'à 400 My : Créer le sous-directory `KSI_HYA` en saisissant `create_evol_space KSI_HYA`. Une autre possibilité est de :

- Créer un sous-directory `KSI_HYA` (ou saisir `create_evol_space KSI_HYA`)
- Copier les fichiers `m020.zams`¹³ et `mon_modele.don` du sous-directory `EXPLOIT`.

Puis :

- Renommer le fichier de données `ksi_hya.don`.
- Adapter la physique et les paramètres de ce fichier au problème de ξ Hya.

Lancer ensuite deux calculs successifs d'un modèle préliminaire de ZAMS (`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 (`PRECISION='pr'`) en utilisant le modèle binaire `ksi_hya_B.hom` obtenu à l'étape précédente.

Conserver le fichier `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 seront initialisés le calcul des modèles des exploitations.

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.

3.16.1 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

12. Voir aussi § 3.2 (Page 25).

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

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. § 2.1 (Page 9). 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 égale celle d'origine gravitationnelle, la composition chimique n'est plus homogène alors qu'elle l'est dans 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 d'extensions `.pms` et `.zams` créés lors initialisations ou des évolutions avec PMS peuvent servir à des initialisations ou à des reprises ultérieures ; il en est de même des binaires d'extension `.hom` créés à l'issue d'initialisations sur la ZAMS. De façon générale ces fichiers binaires servent à optimiser les initialisations.

3.16.2 Exemple d'évolution avec PMS

On se place dans le directory TEST créé au § 2.1 (Page 9).
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:

```

3.16.3 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 les 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. §3.19 (Page 50). 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 §4.13 (Page 65).

3.17 Outils astrophysiques

La mise en œuvre de `CESAM2k` nécessite l'utilisation d'outils astrophysique, comme par exemple pour déduire la valeur rapport $[Z/X]$ d'une mixture, ou encore de transformer une donnée photométrique et une parallaxe en luminosité. Un certain nombre de ces outils sont regroupés dans le sous-directory `UTILS_ASTRO` contenant les programmes d'exploitation de bibliothèques de routines. Ces outils, dont une description est donnée dans les sources, sont les suivants :

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 ν_{\max}

calc_dzdy : $\Delta Z/\Delta Y$ à partir de Y et $[\text{Fe}/\text{H}]$ initial et final

calc_dtl : calcul de $\Delta \log_{10} T_{\text{eff}}$ et ΔL

calc_fesh : calcul de $[\text{Fe}/\text{H}]$

calc_fesh_zsx : $\text{Fe}/\text{H} \iff \text{Z}/\text{X}$

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_lr_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_lg_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_{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** : $[\text{Fe}/\text{H}]$ initial et X pour avoir fesh0 a la fin d'une évolution
- calc_rltgm** : Détermination de 2 quantités parmi T_{eff} , $\log_{10} g$, M/M_{\odot} , R/R_{\odot} , L/L_{\odot} à partir des 3 autres
- calc_r_theta** : diamètre + π en mas $\implies R/R_{\odot}$
- calc_som_mag** : pour une binaire, calcul de la magnitude de A+B
- calc_teff** : T_{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 $[\text{Xi}/\text{H}]$
- 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

3.18 Programmes d'exploitation

Des 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.
- **f037_2k** : Interpolation d'un modèle en des points de masse ou de rayon fixés.

- `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.

3.19 Scripts d'exploitation

Les scripts UNIX suivants des sous-directories `SCRIPTS/BASH` et `SCRIPTS/CSH`¹⁴ facilitent l'installation et l'exploitation¹⁵ de `CESAM2k` :

- `calib2k_pms` : procédure de calibration d'un modèle solaire avec PMS.
- `calib2k_zams` : procédure de calibration d'un modèle solaire initialisé sur la ZAMS homogène.
- `compile2k` : script de compilation avec optimisation.
- `compile2k-dbg` : script de compilation pour debug.
- `compile2k-dbg.list` : script de compilation pour debug pour un ensemble de routines d'une liste.
- `compile2k.list` : script de compilation avec optimisation pour un ensemble de routines d'une liste, avec création ou mise à jour d'une bibliothèque.
- `evol2k_pms` : procédure pour l'évolution à partir de la PMS homogène.
- `evol2k_zams` : procédure pour l'évolution à partir de la ZAMS homogène.
- `exe2k` : procédure d'exécution d'un programme avec optimisation
- `exe2k-dbg` : procédure d'exécution d'un programme avec debug.
- `genere_cesam2k-dbg` : script créant la bibliothèque et l'exécutable `cesam2k-dbg.out` de debug,
- `genere_cesam2k` : script créant la bibliothèque et l'exécutable `cesam2k.out` d'exploitation,
- `lib_del_repl` : script permettant de supprimer, puis de remplacer, une routine dans une bibliothèque.
- `repl2k_mod` : script permettant de compiler, remplacer dans la bibliothèque, un ou plusieurs modules et de créer le module exécutable `cesam2k.out`.
- `repl2k_mod-dbg` : script permettant de compiler, remplacer dans la bibliothèque, un ou plusieurs modules, et de créer le module de debug exécutable `cesam2k-dbg.out`.
- `makefile` : `makefile` permettant de générer l'exécutable `cesam2k.out`.

14. La tendance actuelle étant de privilégier le shell `bash` les scripts `csh` disponibles sont moins nombreux.

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

Chapitre 4

Personnalisation

Bien que l'exécutible 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. CESAM2*k* 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 français¹.
- 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 boites 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 `nl_chim` du fichier de données est `.TRUE.` cf. § 3.8 (Page 35).

4.1 Langue

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

1. Seulement en anglais dans la présente version.

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 un fichier de nom `langue` dans lequel on codera :

```

&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.

4.2 Mixture

Pour utiliser une mixture initiale différente² de celles implémentées, coder `nom_abon='mixture'` dans le fichier de données et, dans l'environnement, placer un fichier de nom `mixture` contenant la mixture désirée en Dex. `CESAM2k` *refuse* toute mixture dont la normalisation est différente de `H=12`. L'exemple suivant se trouve dans le sous-directory `EXPLOIT` :

2. Voir aussi § 5.2 (Page 68).


```

&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. § 3.8 (Page 35).

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.

4.3 Rapports isotopiques

Pour utiliser des rapports isotopiques différents de ceux que CESAM prend par défaut, 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`. 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 ^3He , `be7sz` est l'abondance, quasi nulle, du ^7Be 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. § 3.8 (Page 35).

4.4 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 :

```
&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. § 3.4 (Page 26).

Si on a codé `GARDE_XISH=.FALSE.` dans le fichier de données cf. § 3.8.1 (Page 36), 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.

4.5 Personnalisation de la composition chimique du vent.

Dans le cas standard, avec la donnée `mdot` non nulle, cf. § 3.4 (Page 26), 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 :

```
&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 construit³ 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` `nl_fdon` du fichier de données est `.TRUE`. cf. § 3.8 (Page 35).

4.6 Chute de planétoïdes

Au cours d'une évolution `CESAM2k` permet de simuler une chute de planétoïdes 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 planétoïdes, le nombre total de planétoïdes 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` :

```
&nl_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 planétoïdes. 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 planétoïdes en admettant qu'ils décrivent un arc de $\pi/2$ pendant 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

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'.

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. § 3.8 (Page 35).

4.7 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 :

```
&n1_eos
ifoption=2,
ifmodified=1,
ifion=-1
/
```

L'implantation de cette équation d'état est décrite au § 2.2.6 (Page 20) et son exploitation dans `CESAM2k` au § 3.12 (Page 40). 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`.

4.8 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`⁴ 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.
.TRUE.
.TRUE.
Fichier pour sortie ascii:
```

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

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
1.9891900000000E+33 6.010741880620E+10 2.478601588572E+33 2.000000000000E-02 7.000000000000E-01
1.8000000000000E+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.* § 4.2 (Page 54), 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.

4.9 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`⁵ comportant les réglages à utiliser. **Ces valeurs prévalent sur celles**

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

prédéfinies. Exemple d'un tel fichier réglages :

```
&NL_RLG
m_qs=2
m_ch=2,
m_rot=2,
m_tds=2,
m_ptm=2,
ordre=6,
precic=1.d-4,
precix=1.d-3,
precit=1.d0,
psi0=0.03d0,
loc_zc=1.d-3,
dtmax=0.2d0,
dtmin=1.d-10,
dt0=0.01d0,
d_lum=0.3d0,
d_press=0.2d0,
d_ray=0.2d0,
d_temp=0.2d0,
age_car=0.1d0,
ini0=4,
n_atm=75,
kipp=.TRUE.,
en_m23=.FALSE.,
ctel=0.0d0,
ctep=-1.d0,
ctem=15.d0,
cter=0.d0,
ctet=-1.d0,
dn_fixe=0.05d0,
dpsi=0.03d0,
mu_saha=.TRUE.,
ajuste=.FALSE.,
lisse=.FALSE.,
npt_lisse=1,
l_demi=.TRUE.,
q0=0.d0,
l0=0,
new_bv=.TRUE.,
fcv=.TRUE.,
des_instb=.TRUE.,
no_discon=.FALSE.,
n_min_ZC=5,
iter_qs=0, 0, 0, 0, 0, 0, 7
l_actu= F, F, F, F
```

/

Les significations des paramètres sont les suivantes :

m_qs=2 : Ordre des B-splines pour les variables quasi-statiques.

m_ch=3 : Ordre des B-splines pour l'interpolation de la composition chimique.

m_rot=3 : Ordre des B-splines pour l'interpolation du moment cinétique.

m_tds=2 : Ordre des B-splines pour interpolation de l'énergie graviphique.

m_ptm=2 : Ordre des B-splines pour interpolation de la perte de masse.

ordre=1 : Ordre du schéma d'intégration des réactions nucléaires. avec **rkimps**.

precix=1.d-4 : Précision sur les itérations Newton-Raphson des intégrations spatiales.

precit=0.1d0 : Variation maximale/masse pour l'intégration temporelle de la composition chimique.

ro_test=0.1d0 : Test de variation de l'énergie graviphique si **ro** > **ro_test**.

psi0=0.05d0 : Constante de répartition à assurer.

d_grav=0.5d0 : Variation temporelle maximale de l'énergie graviphique.

loc_zc=5.d-4 : Précision de la localisation des limites zones radiatives / zones convectives.

dtmax=200.d0 : Pas temporel maximum.

dtmin=1.d-8 : Pas temporel minimum.

ini0=5 : 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_masse=.TRUE. : Variables quasi-statiques lagrangiennes.

ctel, ctep, ctem, cter, ctet : Facteurs de répartition.

no_dis=.TRUE. : ajustement de la composition chimique suivant les déplacements des discontinuités.

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=0.02 : Taux limite de variation de la fonction de répartition nécessitant une modification du nombre de couches.

mu_saha=.FALSE. : 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.

l_demi=.FALSE. : un pas sur deux, la composition chimique est définie au milieu des intervalles de discrétisation de l'équilibre quasi-statique.

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 $\varphi = \frac{\partial \ln \rho}{\partial \ln \mu}$.

dlntc=0.07 : coefficient limitant aux environ de 7% la variation relative de la température centrale sur un pas temporel.

fcv=.TRUE. : si besoin, on forcera la convergence.

iter_qs=0, 0, 3, 0, 0, 0, 7 : Permet de ne pas tenir compte d'une précision médiocre sur certaines variables quasi-statiques.

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.* §4.2 (Page 54), il est possible de plus personnaliser ces réglages en donnant au fichier des réglages, celui 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 *le* modèle de nom **mon_modele**. En son absence, ce seront les quantités du fichier **reglages**, s'il n'existe pas, le calcul ne sera pas effectué.

4.10 Limites des shells source

Dans le fichier de données, *cf.* §3.5 (Page 29), 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
```

4.11 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`⁶ :

```
&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
/
```

- 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.,
```

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

```

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.

4.12 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`⁷ :

```
&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`.

4.13 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. *CESAM2k* a la possibilité d'en détourner la plus grande partie vers des fichiers qui pourront, au besoin, lus "off line", cf. § 3.16.3 (Page 48). 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`.

7. Exemple se trouvant dans le sous-directory `EXPLOIT`.

Chapitre 5

Extensions

5.1 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*¹ :

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 réaction² `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 : `nreac=30 ; ind(16)=31 ; WRITE(2,6) ; WRITE(*,6)....`
4. Dans le module `mod_nuc` :
 - (a) Augmenter d'une unité le `PARAMTER nreac_tot`, Exemple : `nreac_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.

5.2 Ajout d'une mixture initiale

Ainsi qu'il est décrit au § 4.2 (Page 54), il est possible de personnaliser les abondances initiales à l'aide d'un fichier `mixture`. Quand la mixture initiale est d'intérêt général³ 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 6

Debug

6.1 Debug

La structure de CESAM2*k* en modules permet une vérification efficace de la syntaxe par le compilateur. Le revers de la médaille est que la mise au point et/ou le debug nécessitent de recompiler tout le module auquel appartient la routine dans laquelle le bug est recherché. Lors de l'installation, il est utile de créer, conjointement à la bibliothèque d'exploitation `libcesam2k.a` une bibliothèque de debug `libcesam2k-dbg.a`. Cette bibliothèque est créée automatiquement lors de l'exécution du script `genere_cesam2k-dbg` du sous-directory `SCRIPTS`. Pour ce faire, utiliser les options de debug du compilateur ; la procédure `compile2k-dbg` du sous-directory `SCRIPTS` en est un exemple. Le temps nécessaire à la construction de la bibliothèque de debug est significativement plus court que celui de la construction de la bibliothèque d'exploitation, facteur 5 à 10 par exemple, facteur qui se retrouve, mais dans l'autre sens, lors d'une exécution.

Exemple : un bug a été détecté dans la routine `des_m` du module `mod_cesam`¹. On introduit des instructions de debug dans `des_m`. Dans le sous-directory `TESTS` on exécute le programme `test_cesam` dans lequel on inclut le module `mod_cesam` :

```
INCLUDE './SOURCE/mod_cesam.f'
```

```
C*****
```

```
PROGRAM test_cesam
```

```
USE mod_cesam
```

```
IMPLICIT NONE
```

```
CALL cesam
```

```
STOP
```

1. Les *bons* compilateurs indiquent le nom des routines où ils détectent un bug.

FIGURE 6.1 – 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 72).

END PROGRAM test_cesam

Au link, le fichier `mod_cesam.mod` créé dans le sous-directory TESTS sera utilisé et non pas celui du sous-directory SOURCE. Lors de l’exécution, les instructions de debug, alors prises en compte, permettront de détecter le défaut. Une fois la routine `des_m` corrigée, dans le sous-directory SOURCE, l’exécution du script : `mod_repl mod_cesam`, remplacera le module `mod_cesam` dans la bibliothèque et recréera l’exécutable `cesam2k.out`. Ainsi, aucun fichier du sous-directory SOURCE n’a été déplacé.

6.2 Bugs connus

1. Erreurs à la compilation.
 - (a) 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** : supprimer l’option `IMPLICIT NONE`.
 - (b) 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.
 - (c) 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.
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. Mauvais fonctionnement du contrôle de la variation temporelle de l’entropie spécifique (TdS), on obtient le message : “TdS varie trop... diminution du pas temporel”, le pas temporel diminue 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 de modifier `d_grav`, valeur du paramètre de contrôle de la variation admise du TdS ; augmenter sa valeur, diminue la précision mais augmente la robustesse ; on peut éliminer le contrôle en donnant une valeur très grande (10^{10} à ce paramètre de contrôle ; il est initialisé dans la routine `cesam`, cf. § 8.10 (Page 173). On contourne parfois la difficulté en utilisant l'approximation de Kippenhahn cf. § 7.4.1 (Page 99). Pour ajuster ces paramètres il est pratique d'utiliser la possibilité de personnaliser les réglages de `CESAM2k` cf. § 4 (Page 53).

4. 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.
5. Divergence au premier pas temporel. **Cause possible** : pas temporel trop élevé. **Remède** : diminuer le `dt0` initialisé 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. § 4 (Page 53).
6. La localisation des limites zones radiatives / zones convectives sur un point de grille ne fonctionne pas. **Cause possible** : les gradients ∇_{rad} et ∇_{ad} ne se croisent pas franchement, ou encore la limite se déplace notablement *i.e.* , 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. § 4 (Page 53).
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. § 4 (Page 53).
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.
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. **Cause possible** : passage d'un point anguleux de la table d'opacité, ou modification de l'indice de grille du point d'indice n_* . **Remède** : souvent, après quelques 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. **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 `nl_temps` du fichier `mon_modele.don`. **Remède** : modifier ces paramètres suivant les indications du § 3.4 (Page 26) précédent.
15. Mauvaise convergence de l'algorithme d'intégration temporelle pour ^2H , se produit vers la fin de la pré-séquence principale, se produit 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 mauvaise.
16. Oscillations violentes 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. § D.1 (Page 267). Ces oscillations ont plusieurs causes décrites, cf. § 7.10.10 (Page 154). Bien que beaucoup d'efforts aient été consacrés à cette difficulté, jusqu'à présent, le moins mauvais remède consiste à supprimer "à la main" les couches incriminées.

La diffusion des éléments chimique évite en grande partie ce phénomène. Lorsque celle-ci n'est pas requise et qu'une bonne représentation de la fréquence de Brunt-Väissälä est nécessaire, dans le fichier de données, cf. § 3.4 (Page 26), coder 'sa' pour le type de précision, et introduire un peu de diffusion turbulente :

```

/
&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'
/

```

qui aura pour effet d'éliminer les discontinuités de la dérivée première de la composition chimique. Le coût à payer est un effort de calcul nettement plus important qu'avec `DIFFUSION=.FALSE.`. La figure 6.1 (Page 70) 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, *CESAM2k* utiliser l'atmosphère standard.
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. § 3.4 (Page 26), permet, parfois, de faire disparaître d'anomalie.
19. Profil très bruité du taux de variation de poids moléculaire moyen Λ , cf. § 7.9.5.1 (Page 133), au voisinage des limites zones radiatives / zones convectives. Remède inconnu. Cette anomalie n'a, en général, pas de conséquence directe sur le profil de la vitesse angulaire.

Deuxième partie
DESCRIPTION DE CESAM2k

Chapitre 7

Numérisation de la physique

Dans ce chapitre, on expose la physique implantée dans CESAM2*k* conjointement aux méthodes numériques utilisées. Le but poursuivi est de donner le maximum d'éléments, afin de faciliter la compréhension tant de la physique implantée que des méthodes numériques utilisées. Certaines parties sont d'un niveau élémentaire, mais tout développement ultérieur de CESAM2*k* est à ce prix.

Les équations de l'évolution stellaire constituent un problème intégro-différentiel non linéaire aux dérivées partielles : c'est un problème aux limites de conditions initiales.

Une seule dimension spatiale est prise en compte dans CESAM2*k*.

La méthode de résolution retenue, la méthode des "lignes" est classique, voir *e.g.* Henrici (1962). Elle consiste, pour chaque pas temporel, à résoudre itérativement et de façon découplée, le problème de conditions initiales (problème de Cauchy) puis le problème aux limites (problème de Dirichlet) – une résolution couplée n'est envisageable que si le nombre d'espèces chimiques dont on doit suivre l'évolution temporelle est réduit, ce qui n'est pas le cas en général.

Le problème aux limites, non linéaire, est résolu par itérations Newton-Raphson qualifiée de "méthode d'Henyei" (Henyei et al., 1959) en structure interne.

Le problème de valeurs initiales est un problème raide, car les diverses échelles de temps en présence diffèrent entre elles par plusieurs décades.

Une difficulté numérique est due à l'hypothèse physique suivant laquelle les mouvements convectifs homogénéisent les espèces chimiques et rigidifient la rotation. De ce fait la continuité de certaines fonctions inconnues est détruite aux limites zones radiatives / zones convectives.

7.1 Intégration par splines-collocation

On décrit dans ce paragraphe la méthode d'intégration numérique utilisée dans CESAM2*k* pour résoudre les problèmes aux limites que constituent les équations de l'équilibre quasi-statique et la restitution de l'atmosphère. Elle utilise des techniques et des algorithmes qui sont, pour la plupart, peu usités en évolution stellaire. Afin d'être accessible sans trop d'investissement en analyse numérique, mais avec le risque de plagier les manuels spécialisés, on donne un aperçu de la réflexion qui a conduit aux algorithmes retenus avant de décrire ces derniers.

Cette méthode consiste à développer les fonctions inconnues sur une base de polynômes par morceaux, les B-splines, et à écrire que ces développements satisfont

les conditions limites et les équations différentielles en un nombre *ad-hoc* de points. Les fonctions de base ne satisfaisant pas les conditions limites, cette méthode est qualifiée de "quasi-spectrale" dans les ouvrages spécialisés.

7.1.1 Les B-splines normalisées

Etant donnée une partition de $[a, b] \subset \mathbb{R} : a = x_0 < x_1 < \dots < x_n = b$, l'ensemble des polynômes par morceaux d'ordre m (degré $+1$), qui se raccordent en x_1, x_2, \dots, x_{n-1} est un espace vectoriel de dimension finie M , dont une base est celle des B-splines. Elle sera notée par la suite : $\{N_i^m\}_{i=1}^M$. Pour $m = 2$ et $n = 5$ ce sont les "fonctions chapeaux" de la Fig. 7.1 (Page 78) et les courbes en cloche pour $m = 4$. Les diverses définitions mathématiques de ces fonctions sont abscondes :

FIGURE 7.1 – B-splines normalisées d'ordre $m=2, 3$ & 4 pour $n=5$ points de grille équidistants.

i -ème différence divisée de la fonction puissance translatée, par exemple. On les trouve dans De Boor (1978, p. 108), Schumaker (1981, p. 118), Trenoguine (1980, p. 341), et dans de nombreux autres ouvrages. Ces fonctions de base, malgré leur algèbre compliquée se prêtent très bien au calcul numérique car les calculs sont faciles, stables et efficaces. En particulier :

- les B-splines sont des fonctions à support borné,
- en un point de leur support il existe au plus m B-splines d'ordre m non nulles, une conséquence est la structure bande des systèmes linéaires apparaissant dans les calculs,
- une B-spline est toujours positive ou nulle,
- la dérivée d'une B-spline est une combinaison linéaire de B-splines d'ordre immédiatement inférieur,
- il existe des algorithmes numériques efficaces et stables pour les calculs avec B-splines.

7.1.2 Résolution d'un problème différentiel

Le principe de l'utilisation des B-splines pour résoudre un problème différentiel est celui de toute méthode spectrale : les fonctions inconnues sont développées sur une base finie, en l'occurrence celle des B-splines. Pour calculer les coefficients, on écrit que ces développements satisfont les équations différentielles et les conditions limites. Pour ce faire il y a une alternative :

- On satisfait les équations en un nombre *ad-hoc* de points, c'est la *collocation* qui est utilisée pour résoudre l'équilibre quasi-statique *cf.* § 7.2 (Page 82) et restituer l'atmosphère *cf.* § 7.3.2 (Page 92) .

FIGURE 7.2 – B-splines normalisées d'ordre $s=2$ (gauche) et $s=3$ (droite) calculées avec la base de C. de Boor pour $n=3$ points de grille équidistants. Les vecteurs nodaux sont schématisés par des croix disposées sous chaque graphe.

- On impose à la base d'être orthogonale à son image par l'opérateur différentiel (*i.e.* au résidu), c'est la méthode des éléments finis (Galerkin) qui est utilisée pour résoudre les équations de diffusion des espèces chimiques, *cf.* § 7.8 (Page 113), et du moment cinétique *cf.* § 7.9 (Page 129).

Ainsi, pour le problème différentiel aux limites :

$$f''(x) = \Phi(x; f, f'), \quad \phi_1(x_1; f, f') = \alpha, \quad \phi_2(x_n; f, f') = \beta, \quad \text{sur } [x_1, x_n] \subset \mathfrak{R}$$

où ϕ_1, ϕ_2 sont des fonctions et α, β des réels donnés. En utilisant la base $\{N_i^m\}_{i=1}^M$ de B-splines d'ordre $m \geq 4$, sur une partition de $[x_1, x_n]$ à préciser, on aura :

$$f(x) \simeq \sum_{i=1}^M f_i N_i^m(x), \quad f'(x) \simeq \sum_{i=1}^M f_i \frac{dN_i^m(x)}{dx}, \quad f''(x) \simeq \sum_{i=1}^M f_i \frac{d^2 N_i^m(x)}{dx^2}$$

Avec la méthode de collocation, les M coefficients f_i sont obtenus en écrivant que le développement de f vérifie les 2 conditions limites et les équations :

$$\sum_{i=1}^M f_i \frac{d^2 N_i^m(z_c)}{dx^2} = \Phi(z_c; f, f')$$

en $M - 2$ points $z_c \in]x_1, x_n[$.

La solution obtenue est en $O(\|\Delta x\|^m)$. Si on adopte un choix judicieux pour les points de collocation la solution est en $O(\|\Delta x\|^{2(m-1)})$: c'est la superconvergence (De Boor, 1978, chap. XV).

Avec la méthode de Galerkin les coefficients f_i seraient, par exemple, obtenus en écrivant que le développement de f vérifie les M équations :

$$\sum_{i=1}^M f_i \left\langle \frac{d^2 N_i^m}{dx^2} \cdot N_l^m \right\rangle = \left\langle \Phi(x_l; f, f') \cdot N_l^m \right\rangle, \quad 1 < l < M$$

où $\langle g, h \rangle$ représente un produit scalaire à définir. Les conditions limites sont introduites par une intégration par parties, *cf.* Trenoguine (1980), Marchouk & Agochkov (1985).

Avec les deux méthodes, si Φ est linéaire, on obtient les coefficients f_i par résolution d'un système linéaire. Dans la cas de l'évolution stellaire, il est nécessaire d'utiliser une méthode itérative *e.g.* point fixe ou méthode de Newton-Raphson les équations étant non linéaires.

7.1.3 L'architecture de CESAM2k

L'algèbre des B-splines étant assez complexe, l'architecture de CESAM2k repose sur une séparation bien distincte entre l'espace physique et celui des fonctions de base. Les équations à résoudre sont écrites sous une forme proche de leur formulation dans l'espace physique. Dans l'espace des B-splines ces équations sont résolues pour les coefficients des fonctions de base. Des routines servent d'interfaces entre les deux espaces. Par exemple, les coefficients des équations de l'équilibre quasi-statique sont calculés dans l'espace physique par la routine `static_m`. Les équations sont résolues dans l'espace des splines par la routine `coll_qs`, la routine `resout` servant d'interface entre les deux espaces, voir la figure 1.1 (Page 7).

Les programmes d'exploitation des résultats opèrent le plus souvent dans l'espace physique.

7.1.4 Collocation pour un système non linéaire

On explicite la méthode spline/collocation pour le système de n_e équations différentielles ordinaires, non linéaires, du premier ordre, avec des conditions limites aux extrémités de l'intervalle de définition *e.g.* les équations de la structure interne. Ces systèmes s'écrivent formellement :

$$\mathcal{F}_j(x; \mathbf{f}, \mathbf{f}') = 0, \quad x \in [x_1, x_n], \quad \mathbf{f} = {}^T(f_1, f_2, \dots, f_{n_e}) \quad 1 \leq j \leq n_e$$

avec les conditions limites :

$$\alpha_q(\mathbf{f}(x_1)) = 0, \quad \forall q \in Q, \quad Q \subseteq J = \{1, 2, \dots, n_e\}, \quad \beta_r(\mathbf{f}(x_n)) = 0, \quad \forall r \in \mathcal{C}_J Q$$

la notation $\mathcal{C}_J Q$ signifiant le complémentaire de Q dans l'ensemble d'indices J . En remplaçant chaque composante f_j de \mathbf{f} , $1 \leq j \leq n_e$, par son développement sur la base des B-splines :

$$f_j(x) = \sum_{i=1}^M f_{i,j} N_i^m(x), \quad f'_j(x) = \sum_{i=1}^M f_{i,j} \frac{dN_i^m(x)}{dx} \quad (7.1)$$

les inconnues sont désormais les réels $f_{i,j}$. Ils vérifient :

$$\left\{ \begin{array}{l} \mathcal{F}_j \left(x; \sum_{i=1}^M f_{i,1} N_i^m(x), \dots, \sum_{i=1}^M f_{i,n_e} N_i^m(x), \sum_{i=1}^M f_{i,1} \frac{dN_i^m(x)}{dx}, \dots, \right. \\ \left. \sum_{i=1}^M f_{i,n_e} \frac{dN_i^m(x)}{dx} \right) = 0, \\ \forall x \in]x_1, x_n[, \quad \forall j \in J \\ \alpha_q \left(\sum_{i=1}^M f_{i,1} N_i^m(x_1), \dots, \sum_{i=1}^M f_{i,n_e} N_i^m(x_1) \right) = 0, \quad \forall q \in Q \\ \beta_r \left(\sum_{i=1}^M f_{i,1} N_i^m(x_n), \dots, \sum_{i=1}^M f_{i,n_e} N_i^m(x_n) \right) = 0, \quad \forall r \in \mathcal{C}_J Q \end{array} \right.$$

Pour un système différentiel linéaire, les $f_{i,j}$ se rassemblent. Ils sont évalués en écrivant (7.1) en $M - 1$ points de collocation $z_c \in]x_1, x_n[$ et en résolvant le système linéaire obtenu. Les équations de la structure interne ne sont pas linéaires. On utilise la méthode itérative de Newton-Raphson pour résoudre le système (??). Ce qui nécessite une solution initiale \mathbf{f}^0 . En notant \mathbf{f}^p la solution obtenue à l'itération $p \geq 0$, les équations linéarisées sont :

$$\left\{ \begin{array}{l} \mathcal{F}_j \left(z_c; \sum_{i=1}^M f_{i,1}^p N_i^m(z_c), \dots, \sum_{i=1}^M f_{i,n_e}^p N_i^m(z_c), \sum_{i=1}^M f_{i,1}^p \frac{dN_i^m(z_c)}{dx}, \dots, \right. \\ \left. \sum_{i=1}^M f_{i,n_e}^p \frac{dN_i^m(z_c)}{dx} \right) = \\ = \sum_{k=1}^{n_e} \sum_{i=1}^M \left(\frac{\partial \mathcal{F}_j}{\partial f_k} N_i^m(z_c) + \frac{\partial \mathcal{F}_j}{\partial f'_k} \frac{dN_i^m(z_c)}{dx} \right) df_{i,k}^p, \quad \forall j \in J, \quad 1 \leq c < M \\ \alpha_q \left(\sum_{i=1}^M f_{i,1}^p N_i^m(x_1), \dots, \sum_{i=1}^M f_{i,n_e}^p N_i^m(x_1) \right) = \\ \sum_{k=1}^{n_e} \frac{\partial \alpha_q}{\partial f_k} \sum_{i=1}^M N_i^m(x_1) df_{i,k}^p, \quad \forall q \in Q, \\ \beta_r \left(\sum_{i=1}^M f_{i,1}^p N_i^m(x_n), \dots, \sum_{i=1}^M f_{i,n_e}^p N_i^m(x_n) \right) = \\ \sum_{k=1}^{n_e} \frac{\partial \beta_r}{\partial f_k} \sum_{i=1}^M N_i^m(x_n) df_{i,k}^p, \quad \forall r \in \mathcal{C}_J Q \end{array} \right. \quad (7.2)$$

Les sommes sur i ne portent, en fait, que sur m indices puisqu'en un point x il y a, au plus, m B-splines non nulles. Le système (??) est linéaire pour les corrections $df_{i,k}^p$. On aura :

$$f_{i,k}^{p+1} = f_{i,k}^p - df_{i,k}^p, \quad 1 \leq i \leq M, \quad 1 \leq k \leq n_e$$

Sans diffusion, à cause du mélange convectif, aux limites zones radiatives / zones convectives la composition chimique présente une discontinuité, en résulte une discontinuité de la densité. Avec diffusion, les discontinuités n'affectent que les dérivées premières. Ces discontinuités ont conduit au choix de la pression comme variable dépendante au lieu de la densité. De plus, l'utilisation de cette dernière nécessiterait l'évaluation des dérivées partielles $(\partial P/\partial \rho)_{T,X}$, $(\partial P/\partial T)_{\rho,X}$ et $(\partial P/\partial X)_{T,\rho}$ dans les deux premières équations du système (7.2), et non seulement les dérivées secondes "thermodynamiques" :

$$\left(\frac{\partial^2 P}{\partial T^2}\right)_{\rho,X}, \quad \left(\frac{\partial^2 P}{\partial \rho^2}\right)_{T,X}, \quad \left(\frac{\partial^2 P}{\partial \rho \partial T}\right)_X, \quad \left(\frac{\partial^2 P}{\partial \rho \partial X}\right)_T, \quad \left(\frac{\partial^2 P}{\partial T \partial X}\right)_\rho$$

Les équations de l'équilibre quasi-statique sont intégrées avec un schéma supposant la continuité de chaque variable dépendante avec, éventuellement, une discontinuité de la dérivée première, *cf.* § 7.1.5 (Page 80). Sans diffusion, en raison de sa nature discontinue, au voisinage des limites zones radiatives / zones convectives, le profil de γ présentera des oscillations, toutefois sans importance car, sans diffusion, γ n'est pas couplé aux autres variables. Il n'en est pas de même avec diffusion, mais alors γ est continue, non dérivable aux limites zones radiatives / zones convectives, situation prise en compte par le schéma d'intégration.

7.1.5 Choix de la base de la collocation

Ainsi qu'ils sont définis dans De Boor (1978, p. 280), un choix particulier de la base et des points de collocation permet d'obtenir un ordre de précision supérieur à celui des fonctions d'interpolation c'est la *superconvergence*. On se donne l'ordre des B-splines m et la grille des points de raccord. Les morceaux de polynômes se raccordent en ces points en assurant la continuité de la B-spline et de ses dérivées jusqu'à un ordre inférieur d'une unité à celui du système différentiel. Dans CESAM2k on utilise la grille équidistante $\{1, 2, \dots, n\}$, l'ordre des équations différentielles est $r = 1$ et l'ordre des B-splines est fixé par le type de précision indiqué dans le fichier de données *cf.* § 3.4 (Page 26), ou, éventuellement dans un fichier **reglages** *cf.* § 4.9 (Page 60).

Entre deux points de la grille, les variables dépendantes sont interpolées par des morceaux de polynômes d'ordre $m + r = 2$ *i.e.* droites, ou 3 *i.e.* paraboles. Les morceaux de polynômes se raccordent sur les points de grille par continuité, leurs dérivées premières pouvant être, au besoin, différentes. Cette particularité est exploitée pour décrire correctement le profil de la densité aux limites zones radiatives / zones convectives, *cf.* § 7.2.1 (Page 83).

La figure 3 représente cette base pour $n = 3$. Aux n points de la grille, la solution est d'ordre $2m = 2$ ou 4; cette superconvergence est due à cette disposition particulière des points de collocation. Entre les points de raccord la solution est d'ordre $m+r = 2$ ou 3; la superconvergence ne peut donc être exploitée directement en tout point. S'il est nécessaire de connaître la solution en des points autres que ceux définis par la grille, pour exploiter la superconvergence, il faut faire une interpolation d'ordre $2m^1$.

1. L'expérience des calculs a montré que l'exploitation de la superconvergence était délicate à cause des discontinuités dues aux déplacements des limites zone radiative / zone convective.

à chaque point de collocation correspond une ligne de \times . Chacun d'eux² représente une matrice carrée dont l'ordre est le nombre d'inconnues $n_e = 7$, $n_e = 8$ avec pression turbulente. Par exemple, avec $m = 2$, à chaque intervalle entre deux points de raccord, correspond un bloc de $m = 2$ lignes de $(m + r) \times = (2 + 1) \times = 3 \times$. Deux blocs consécutifs se raccordant par $r = 1$ colonne de blocs. Les lignes correspondant aux conditions limites sont représentées par des \circ . Ce sont des matrices de format $3 \times n_e$, puisqu'il y a trois conditions limites à chaque extrémité de l'intervalle $[1, n]$ d'intégration.

Pour la structure interne, le système linéaire est formé et résolu dans le sous-programme `coll_qs` cf. § 8.16 (Page 177) à partir des coefficients calculés dans `static_m`, cf. § 8.126 (Page 232), ou `static_r`, cf. § 8.126 (Page 232). Pour la restitution de l'atmosphère, il s'agit respectivement des routines `coll_atm`, cf. § 8.15 (Page 176) et `eqatm`, cf. § 8.43 (Page 189).

On peut remarquer qu'il suffit de garder en mémoire des lignes de longueur $(m + r)n_e = (2 + 1)6 = 18$ pour résoudre le système. Il comporte $n_e((n - 1)m + r) = 6((150 - 1)2 + 1) = 1794$ lignes pour $n = 150$. Cette remarque est exploitée dans la routine `gausdp_g` servant à résoudre le système par élimination de Gauss avec pivot partiel, cf. § 9.1 (Page 241).

S'il n'est pas tenu compte de la pression turbulente une équation est omise.

7.2 Equations de la structure interne

CESAM2*k* résout les équations de la structure interne sous une forme assez éloignée de leur expression classique. Dans ce paragraphe, on établit les relations utilisées, puis on donne le changement de variable permettant d'adapter automatiquement le réseau à l'évolution des variations des inconnues, enfin on décrit la méthode utilisée pour fixer un point de la grille de discrétisation à chaque limite entre une zone radiative et une zone convective.

Avec, comme variable indépendante lagrangienne, la masse M contenue à l'intérieur d'une sphère de rayon R , les équations de l'évolution stellaire sont *e.g.* Kippenhahn & Weigert (1991, par. 9.1) :

$$\left\{ \begin{array}{l} \frac{\partial P}{\partial M} = -\frac{GM}{4\pi R^4} + \frac{\Omega^2}{6\pi R} \\ \frac{\partial T}{\partial M} = \frac{\partial P}{\partial M} \frac{T}{P} \nabla \\ \frac{\partial R}{\partial M} = \frac{1}{4\pi R^2 \rho} \\ \frac{\partial L}{\partial M} = \epsilon - \epsilon_G = \epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \\ P = P_{\text{gaz}} + P_{\text{tur}} \\ \frac{\partial X_i}{\partial t} = -\frac{\partial F_i}{\partial M} + \Psi_i(P_{\text{gaz}}, T; \mathcal{X}) \\ 1 \leq i \leq n_{\text{elem}}. \end{array} \right. \left\{ \begin{array}{l} \text{conditions limites :} \\ R(0, t) = 0, \quad L(0, t) = 0, \\ P(M_b, t) = P_b(L, R), \\ P_{\text{gaz}}(M_b, t) = P_{\text{gaz}_b}(L, R), \\ T(M_b, t) = T_b(L, R), \\ \text{conditions initiales :} \\ X_i(M, 0) = X_{i,0}, \\ i = 1, \dots, n_{\text{elem}}, \\ P(M, 0) = P_0(M), \\ P_{\text{gaz}}(M, 0) = P_{\text{gaz}_0}(M), \\ T(M, 0) = T_0(M), \\ R(M, 0) = R_0(M), \\ L(M, 0) = L_0(M), \\ 0 \leq M \leq M_b. \end{array} \right. \quad (7.3)$$

2. Si on ne tient pas compte de la pression turbulente.

les notations classiques ont été utilisées ; R : rayon (distance au centre de l'étoile), $P \equiv P_{\text{tot}}$: pression totale, P_{gaz} : pression gazeuse, P_{tur} : pression turbulente, ρ : densité, Ω : vitesse angulaire, G : constante de la gravitation, T : température, $\nabla \equiv \left(\frac{\partial \ln T}{\partial \ln P} \right)_t$: gradient, L : luminosité, $\epsilon = \epsilon_\nu - \frac{\partial \epsilon_\Omega}{\partial t} - \epsilon_G$: débit d'énergie résultant des réactions nucléaires, de la dissipation locale du moment cinétique *cf.* § 7.6 (Page 108), et des pertes par neutrinos, ϵ_G est l'énergie gravifique, Eq. 7.4.2 (Page 100), M : masse incluse dans la sphère de rayon R , t : temps, X_i : abondance de l'élément chimique i , F_i : flux diffusif pour l'élément chimique i , $\mathcal{X} = \{X_i\}$: vecteur de composition chimique, Ψ_i : taux de variation de l'abondance de l'élément chimique i par les réactions thermonucléaires, n_{elem} : nombre d'espèces chimiques considérées. $M_b(R, L)$, $P_b(L, R)$, $P_{\text{gaz}_b}(L, R)$ et $T_b(L, R)$ sont quatre fonctions décrivant respectivement les variations de la masse, de la pression totale, de la pression gazeuse et de la température, en fonction du rayon et de la luminosité à la limite extérieure de l'enveloppe qui coïncide avec la base de l'atmosphère. Les 5 fonctions : $P_0(M)$, $P_{\text{gaz}_0}(M)$, $T_0(M)$, $R_0(M)$ et $L_0(M)$ décrivent, en fonction de la masse, respectivement la pression totale, la pression gazeuse, la température, le rayon et la luminosité du modèle d'âge zéro.

Cette formulation suppose la symétrie sphérique et ne tient pas compte du champ magnétique. Les équations décrivant la diffusion des éléments chimiques et du moment cinétique seront respectivement décrites au § 7.8 (Page 113) et § 7.9 (Page 129).

7.2.1 Discontinuités de la densité

L'échelle de temps de retournement des éléments convectifs étant petite devant l'échelle de temps nucléaire il est supposé que la convection homogénéise les zones convectives, *cf.* § 7.9.7 (Page 144). La composition chimique, et donc la densité, sont **discontinues** aux limites zones radiatives / zones convectives³. CESAM2*k* utilise des vecteurs nodaux pour les diverses intégrations (structure, composition chimiques, moment cinétique) construits de façon telle que les points où sont formées les équations ne coïncident pas avec les points de discontinuité, ainsi ces dernières sont implicitement prises en compte.

7.2.2 Variables d'intégration

Les variables naturelles par rapport auxquelles sont écrites les équations (7.2) ne sont pas bien adaptées au calcul numérique. En particulier, les dérivées de la pression et du rayon ont des singularités en $M \equiv 0$, *i.e.* au centre.

Ces singularités disparaissent en utilisant le rayon R comme variable indépendante. La condition limite externe devient alors une condition de limite libre, puisque le rayon total n'est pas fixe. Bien que cette limite libre ne soit pas une difficulté pour la méthode numérique utilisée dans CESAM2*k*, le rayon n'est pas, pour autant, le meilleur choix pour la variable indépendante ; on peut en effet montrer (Morel, 1997) que les singularités sont levées et qu'une utilisation optimale de l'approximation numérique est obtenue lorsqu'on utilise respectivement pour la masse, le rayon et la

3. Quand la diffusion des éléments chimiques est prise en compte, la composition chimique et la densité sont **continues non dérivables** en ces limites.

luminosité les variables suivantes : $M^{2/3}$, R^2 et $L^{2/3}$. Il est apparu pratique d'utiliser des variables normalisées aux valeurs solaires :

$$\xi = \ln P, \quad \xi_g = \ln P_{\text{gaz}}, \quad \eta = \ln T, \quad \mu = \left(\frac{M}{M_\odot} \right)^{\frac{2}{3}}, \quad \zeta = \left(\frac{R}{R_\odot} \right)^2, \quad \lambda = \left(\frac{L}{L_\odot} \right)^{\frac{2}{3}}, \quad \gamma = \ln \rho$$

Avec ces nouvelles variables, le système (7.2) devient :

$$\left\{ \begin{array}{l} \frac{\partial \xi}{\partial \mu} = \left[-\frac{3G}{8\pi} \left(\frac{M_\odot}{R_\odot^2} \right)^2 \left(\frac{\mu}{\zeta} \right)^2 + \frac{M_\odot}{4\pi R_\odot} \sqrt{\frac{\mu}{\zeta}} \Omega^2 \right] \exp(-\xi) \\ \frac{\partial \eta}{\partial \mu} = \frac{\partial \xi_t}{\partial \mu} \nabla \\ \frac{\partial \zeta}{\partial \mu} = \frac{3}{4\pi} \frac{M_\odot}{R_\odot^3} \frac{1}{\rho} \left(\frac{\mu}{\zeta} \right)^{\frac{1}{2}} \\ \exp(\xi) = \exp(\xi_g) + P_{\text{tur}} \\ \gamma = \ln \rho(\exp(\xi_g), \exp(\eta), X_i) \\ \frac{\partial \lambda}{\partial \mu} = \frac{M_\odot}{L_\odot} \left(\frac{\mu}{\lambda} \right)^{\frac{1}{2}} \left[\epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \right] \\ \frac{\partial X_i}{\partial t} = -\frac{2}{3M_\odot \sqrt{\mu}} \frac{\partial F_i}{\partial \mu} + \Psi_i(\xi_t, \eta; \mathcal{X}), \quad 1 \leq i \leq n_{\text{elem}} \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{conditions limites} \\ \zeta(1, t) = 1, \quad \lambda(1, t) = 1 \\ \xi(\mu_{\text{ext}}, t) = \ln \left(P_{\text{ext}} \left(\sqrt{\lambda}^3 L_\odot, \sqrt{\zeta} R_\odot \right) \right) \\ \xi_g(\mu_{\text{ext}}, t) = \xi(\mu_{\text{ext}}, t) \\ \eta(\mu_{\text{ext}}, t) = \ln \left(T_{\text{ext}} \left(\sqrt{\lambda}^3 L_\odot, \sqrt{\zeta} R_\odot \right) \right) \\ \gamma = \ln \rho(P_{\text{ext}}, T_{\text{ext}}, X_i) \\ \text{conditions initiales} \\ X_i(\mu, 0) = X_{i,0}, \quad i = 1, \dots, n_{\text{elem}} \\ \xi(\mu, 0) = \xi_0(\mu), \quad \xi_g(\mu, 0) = \xi_{g,0}(\mu), \\ \eta(\mu, 0) = \eta_0(\mu), \quad \gamma(\mu, 0) = \gamma_0(\mu) \\ \zeta(\mu, 0) = \zeta_0(\mu), \quad \lambda(\mu, 0) = \lambda_0(\mu), \\ 1 \leq \mu \leq \mu_{\text{ext}} \end{array} \right.$$

Au centre, la singularité des gradients est levée :

$$\lim_{\mu \rightarrow 0} \left| \frac{\partial \xi}{\partial \mu} \right| = \frac{G}{2} \left(\frac{4\pi M_\odot^2}{3} \right)^{\frac{1}{3}} \exp(-\xi(0, t)) \rho(0, t)^{\frac{4}{3}} < \infty,$$

$$\lim_{\mu \rightarrow 0} \left| \frac{\partial \zeta}{\partial \mu} \right| = \left(\frac{3M_\odot}{4\pi R_\odot^3} \right)^{\frac{2}{3}} \rho^{-\frac{2}{3}}(0, t) < \infty,$$

et, de plus :

$$\lim_{\mu \rightarrow 0} \left| \frac{\mu}{\lambda} \right| = \lim_{\mu \rightarrow 0} \left| \frac{\partial \lambda}{\partial \mu} \right| = \frac{M_\odot}{L_\odot} \left[\epsilon - T \left(\frac{\partial S}{\partial t} \right)_\mu \right],$$

$$\lim_{\mu \rightarrow 0} \frac{\mu}{\zeta} = \left(\frac{4\pi}{3} \right)^{\frac{2}{3}} \left(\frac{R_\odot^2}{M_\odot} \right)^{\frac{2}{3}} \rho^{\frac{2}{3}},$$

quantités dérivables une fois au moins. Sans l'exposant 2/3 qui affecte la luminosité, le second membre de l'équation d'énergie eut été :

$$\frac{\partial \lambda}{\partial \mu} = \frac{M_\odot}{L_\odot} \mu^{\frac{1}{2}} \left[\epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \right]$$

la racine carrée, mal représentée par un polynôme, au voisinage de $\mu = 0$, entraînant une instabilité numérique. L'exposant $2/3$ permet donc une meilleure représentation numérique de la luminosité au voisinage du centre, mais interdit à cette variable dépendante de devenir négative, ce qui, dans certaines phases de l'évolution, peut devenir rédhibitoire ce qui peut survenir, lorsque, localement, les pertes énergétiques par diltation ou encore par neutinos surpassent la création d'énergie d'origine nucléaire.. Lorsque cette éventualité peut se présenter, bien que moins satisfaisant numériquement, il est nécessaire d'utiliser le rayon comme variable indépendante eulérienne et, dans ce cas, l'ensemble de variables est :

$$\xi = \ln P, \quad \xi_g = \ln P_{\text{gaz}}, \quad \eta = \ln T, \quad m = \left(\frac{M}{M_\odot} \right), \quad r = \left(\frac{R}{R_\odot} \right), \quad l = \left(\frac{L}{L_\odot} \right), \quad \gamma = \ln \rho.$$

A l'issue d'innombrables tentatives infructueuses, force a été d'admettre que ces variables "eulériennes" n'étaient pas satisfaisantes : à cause de la convergence incertaine de la résolution numérique.

A la suite du travail de L.Piau, les difficultés précédentes ont été contournées par l'utilisation des variables :

$$\xi = \ln P, \quad \xi_g = \ln P_{\text{gaz}}, \quad \eta = \ln T, \quad \nu = m^{\frac{1}{3}} = \left(\frac{M}{M_\odot} \right)^{\frac{1}{3}}, \quad r = \frac{R}{R_\odot}, \quad l = \frac{L}{L_\odot}, \quad \gamma = \ln \rho$$

En effet la singularité est levée :

$$\lim_{r \rightarrow 0} \frac{m^{\frac{1}{3}}}{r} = \left(\frac{1}{3} 4\pi \rho \right)^{\frac{1}{3}}.$$

$$\begin{aligned} \frac{\partial \xi}{\partial \nu} &= \left[-\frac{3G}{4\pi} \frac{M_\odot^2}{R_\odot^4} \frac{\nu^5}{r^4} + \frac{M_\odot}{2\pi R_\odot} \frac{\nu^2}{r} \Omega^2 \right] \exp(-\xi) \\ \frac{\partial \eta}{\partial \nu} &= \frac{\partial \xi}{\partial q} \nabla \\ \frac{\partial r}{\partial \nu} &= \frac{3}{4\pi} \frac{M_\odot}{R_\odot^3} \frac{\nu^2}{\rho r^2} \\ \exp(\xi) &= \exp(\xi_g) + P_{\text{tur}} \\ \gamma &= \ln \rho(\exp(\xi_g), \exp(\eta), X_i) \\ \frac{\partial l}{\partial \nu} &= \frac{3M_\odot}{L_\odot} \nu^2 \left[\epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \right] \\ \frac{\partial X_i}{\partial t} &= -\frac{1}{3M_\odot \nu^2} \frac{\partial F_i}{\partial \nu} + \Psi_i(\xi_t, \eta; \mathcal{X}), \quad i = 1, \dots, n_{\text{elem}} \end{aligned}$$

7.2.3 Adaptation de la discrétisation

Les zones dans lesquelles les solutions ont une variation rapide se déplacent au cours de l'évolution, si bien que le réseau initial devient plus ou moins rapidement inadapté. Pour ne pas dégrader la précision, il faut alors resserrer les nœuds dans les régions à variations fortes, les relacher sur les plateaux. Deux solutions peuvent être apportées à ce problème numérique difficile :

- on ajoute/retire une ou plusieurs couches lorsque l'incrément d'une variable est trop/pas assez important (Kippenhahn Kippenhahn et al. (1968)).
- on dispose les nœuds automatiquement de façon à satisfaire certains critères (Eggleton, 1971; Press et al., 1986).

CESAM2*k* utilise la seconde solution. L'idée consiste à effectuer un changement de variable. On utilise comme variable dépendante une fonction, à définir, des variables dépendantes dont on désire contrôler les variations, fonction à laquelle on impose une variation constante d'un nœud de la grille au suivant. Dans un premier temps le nombre total de nœuds est fixé. Ceux-ci sont disposés automatiquement par le calcul aux abscisses *ad-hoc*. La masse, variable indépendante devient une variable dépendante dont la valeur n'est connue qu'à l'issue de l'intégration. A cette fin on définit la fonction d'espacement $Q(\mu, t)$ à partir de la connaissance que l'on a, *a priori*, de la solution. Par exemple, pour la structure interne, on pourrait⁴ prendre : $Q(\mu, t) = \ln P$. On cherche à l'instant t une répartition des abscisses $\mu_i, i = 1, \dots, n$ telle que l'incrément de la fonction d'espacement soit constant d'un nœud à l'autre de la grille. On cherchera donc à réaliser :

$$Q(\mu_i, t) - Q(\mu_{i+1}, t) = Cte(t), \quad i = 1, \dots, n - 1 \quad (7.4)$$

Cela impose une monotonie stricte à la fonction d'espacement $Q(\mu, t)$. Le nombre de couches n étant donné, pour t fixé, on introduit la fonction "indice" $q(\mu, t)$ appliquant l'intervalle de variation de $\mu : [0, \mu_{\text{ext}}]$ sur $[1, n] \subset \mathbf{R}$. La condition Eq. 7.3 (Page 86) s'écrit alors de façon équivalente :

$$\left(\frac{\partial Q}{\partial q} \right)_t = Cte(t) = \psi(t) \iff \left(\frac{\partial^2 Q}{\partial q^2} \right)_t = \left(\frac{\partial \psi}{\partial q} \right)_t = 0$$

et, pour les conditions limites : $\mu = 0$ à $q = n$ et $\mu = \mu_{\text{ext}}$ à $q = 1$, puisque la couche d'indice 1 correspond au centre, et celle d'indice n à la surface.

Pour t fixé, le changement de variable : $\mu \rightarrow q(\mu, t)$ permet d'écrire :

$$\left(\frac{\partial Q}{\partial q} \right)_t = \left(\frac{\partial Q}{\partial \mu} \right)_t \left(\frac{\partial \mu}{\partial q} \right)_t = \theta \left(\frac{\partial \mu}{\partial q} \right)_t$$

où $\theta(\mu, t) = \left(\frac{\partial Q}{\partial \mu} \right)_t$ est calculé à partir de l'expression de la fonction d'espacement $Q(\mu, t)$. Il y a donc deux fonctions inconnues supplémentaires : $\psi(t) = \left(\frac{\partial Q}{\partial q} \right)_t$ et $\mu(q, t)$, solutions du système différentiel du premier ordre avec conditions limites :

$$\left(\frac{\partial \mu}{\partial q} \right)_t = \frac{\psi}{\theta}, \quad \left(\frac{\partial \psi}{\partial q} \right)_t = 0, \quad \text{avec} : \begin{cases} q = 1, \mu = 0 \\ q = n, \mu = \mu_{\text{ext}} \end{cases}$$

Ces équations sont résolues simultanément avec le système des équations (7.2) écrites par rapport aux variables q et t ; pour ce faire, on dispose des relations suivantes, où $f(\mu, t)$ désigne une variable dépendante quelconque :

$$\left(\frac{\partial f}{\partial t} \right)_\mu = \left(\frac{\partial f}{\partial t} \right)_q - \left(\frac{\partial f}{\partial \mu} \right)_t \left(\frac{\partial \mu}{\partial t} \right)_q, \quad \left(\frac{\partial f}{\partial \mu} \right)_t = \left(\frac{\partial f}{\partial q} \right)_t / \left(\frac{\partial \mu}{\partial q} \right)_t.$$

Les abscisses utilisées sont 1, 2, ..., n , on dispose donc d'une **grille équidistante**.

4. Ce qui équivaut à prendre la pression comme variable indépendante.

7.2.4 Modification du nombre total de couches

On a supposé jusqu'à présent que le nombre total de couches était fixé une fois pour toutes, ce qui constitue évidemment une contrainte, puisque pour une phase donnée de l'évolution, il y a le risque de travailler avec une discrétisation trop fine (*respt.* grossière) avec comme conséquence des calculs inutiles (*respt.* imprécis). Avec une augmentation (*respt.* diminution) du nombre de couches, le saut⁵ $\psi(t)$ de la fonction de répartition d'une couche à la suivante diminue (*respt.* augmente), si bien que d'un pas temporel au suivant, il est possible d'ajuster le nombre de couches afin de maintenir $\psi(t)$ dans un intervalle fixé. Cette disposition correspond exactement au sens physique que l'on peut attribuer à la fonction de répartition qui est de maintenir dans un intervalle donné les variations de certaines grandeurs physiques.

Alors $\psi(t) = \left(\frac{\partial Q}{\partial q}\right)_\mu$ et $\theta(\mu, t) = \left(\frac{\partial Q}{\partial \mu}\right)_t$, $\Rightarrow \frac{\psi}{\theta} = \left(\frac{\partial \mu}{\partial q}\right)_t$ et les équations (7.2) deviennent :

$$\left\{ \begin{array}{l} \frac{\partial \xi}{\partial q} = \left[-\frac{3G}{8\pi} \left(\frac{M_\odot}{R_\odot^2}\right)^2 \left(\frac{\mu}{\zeta}\right)^2 + \frac{M_\odot}{4\pi R_\odot} \sqrt{\frac{\mu}{\zeta}} \Omega^2 \right] \exp(-\xi) \frac{\psi}{\theta} \\ \frac{\partial \eta}{\partial q} = \frac{\partial \xi}{\partial q} \nabla \\ \frac{\partial \zeta}{\partial q} = \frac{3}{4\pi} \frac{M_\odot}{R_\odot^3} \frac{1}{\rho} \left(\frac{\mu}{\zeta}\right)^{\frac{1}{2}} \frac{\psi}{\theta} \\ \exp(\xi) = \exp(\xi_g) + P_{\text{tur}} \\ \gamma = \ln \rho(\exp(\xi_g), \exp(\eta), X_i) \\ \frac{\partial \lambda}{\partial q} = \frac{M_\odot}{L_\odot} \left(\frac{\mu}{\lambda}\right)^{\frac{1}{2}} \left[\epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \right] \frac{\psi}{\theta} \\ \frac{\partial \mu}{\partial q} = \frac{\psi}{\theta} \\ \frac{\partial \psi}{\partial q} = 0 \\ \frac{\partial X_i}{\partial t} = -\frac{2}{3M_\odot \sqrt{\mu}} \frac{\partial F_i}{\partial \mu} + \Psi_i(\xi_t, \eta; \mathcal{X}), \quad 1 \leq i \leq n_{\text{elem}} \\ i = 1, \dots, n_{\text{elem}} \end{array} \right. \quad (7.5)$$

$$\left\{ \begin{array}{l} \text{conditions limites} \\ \mu(n, t) = \mu_{\text{ext}} \left(\sqrt{\lambda^3(n, t)} L_\odot, \sqrt{\zeta(n, t)} R_\odot \right), \\ \xi(n, t) = \ln \left(P_{\text{ext}} \left(\sqrt{\lambda^3(n, t)} L_\odot, \sqrt{\zeta(n, t)} R_\odot \right) \right), \\ \xi_g(n, t) = \xi(n, t), \\ \eta(n, t) = \ln \left(T_{\text{ext}} \left(\sqrt{\lambda^3(n, t)} L_\odot, \sqrt{\zeta(n, t)} R_\odot \right) \right), \\ \gamma(n, t) = \ln \rho(P_{\text{ext}}, T_{\text{ext}}, X_i) \\ \zeta(1, t) = 0, \quad \lambda(1, t) = 0, \quad \mu(1, t) = 0. \\ \text{conditions initiales} \\ X_i(q, 0) = X_{i,0}, \quad i = 1, \dots, n_{\text{elem}} \\ \xi(q, 0) = \xi_0(q), \quad \xi_g(q, 0) = \xi_{g,0}(q), \quad \eta(q, 0) = \eta_0(q), \\ \zeta(q, 0) = \zeta_0(q), \quad \lambda(q, 0) = \lambda_0(q), \quad \gamma(q, 0) = \gamma_0(q) \\ 1 \leq q \leq n. \end{array} \right.$$

5. $\psi(t)$ est soit positif, soit négatif, ici il est supposé positif.

Ces équations de l'espace physique sont formées dans l'une des routines `static_m13` ou `static_m23`, cf. § 8.126 (Page 232). La fonction d'espacement Q doit être strictement monotone ; il faut donc qu'elle ne soit composée que de fonctions variant dans le même sens. Il va de soi que son expression doit être la plus simple possible. Puisque ξ et η varient en sens inverse de ζ , λ et μ , quand il n'y a aucune raison de favoriser une variable plutôt qu'une autre, formellement, il conviendrait d'utiliser (Eggleton, 1971) :

$$Q(\mu, t) = \frac{\xi}{\Delta\xi} + \frac{\eta}{\Delta\eta} - \frac{\zeta}{\Delta\zeta} - \frac{\lambda}{\Delta\lambda} - \frac{\mu}{\Delta\mu}, \quad \Delta\xi = \xi(n) - \xi(1), \\ \Delta\eta = \eta(n) - \eta(1), \quad \Delta\zeta = \zeta(n) - \zeta(1), \quad \Delta\lambda = \lambda(n) - \lambda(1), \quad \Delta\mu = \mu(n) - \mu(1).$$

On aurait donc :

$$\theta(\mu, t) = \left(\frac{\partial Q}{\partial \mu}\right)_t = \frac{1}{\Delta\xi} \frac{3G}{8\pi} \left(\frac{M_\odot}{R_\odot^2}\right)^2 \exp(-\xi) \left(\frac{\mu}{\zeta}\right)^2 \left(1 + \nabla \frac{\Delta\xi}{\Delta\eta}\right) - \\ - \frac{1}{\Delta\zeta} \frac{3}{4\pi} \frac{M_\odot}{R_\odot^3} \frac{1}{\rho} \left(\frac{\mu}{\zeta}\right)^{\frac{1}{2}} - \frac{1}{\Delta\lambda} \frac{M_\odot}{L_\odot} \left(\frac{\mu}{\lambda}\right)^{\frac{1}{2}} \epsilon - 1. \quad (7.6)$$

Les facteurs de répartition $\Delta\xi$, $\Delta\eta$, $\Delta\zeta$, $\Delta\lambda$ et $\Delta\mu$ peuvent être, ou bien fixés une fois pour toutes ou bien, ajustés à l'issue de chaque pas temporel de façon à suivre l'évolution.

Si cette forme "complète" de la fonction de répartition a été utilisée pour les versions 0, 1, et 2 de `CESAM2k`, à partir de la version 3, il s'est avéré que la forme simplifiée $Q \equiv a\xi + b\mu$ était la plus robuste et, de fait, la plus efficace. Les valeurs **fixes** $a = -1$ et $b = 15$ des facteurs de répartition utilisés, sont définies dans la routine `cesam`, cf. § 8.10 (Page 173) qui fait office de programme principal.

Cette disposition s'est avérée insuffisante pour les modèles évolués, en particulier lors de l'amorçage de la combustion de l'hélium ou en présence de pulses thermiques. Dans ces situations l'énergie nucléaire est libérée dans une fraction si restreinte de l'étoile qu'une fonction de répartition reposant uniquement sur le profil de pression est bien en peine "d'étaler". La situation est complexifiée par les extensions et les contractions du milieu qui libèrent ou absorbent de l'énergie gravitationnelle si bien que la luminosité n'a pas une variation monotone ne peut pas figurer dans la fonction de répartition.

À partir de la version 4 de `CESAM2k` cette difficulté a été contournée en introduisant, dans la fonction de répartition, la variable dépendante Ln , la "luminosité nucléaire". Elle correspond à l'énergie d'origine nucléaire libérée et vérifie le problème différentiel :

$$\frac{\partial Ln}{\partial M} = \epsilon \implies \frac{\partial ln}{\partial q} = \frac{3M_\odot}{L_\odot} \nu^2 \epsilon \frac{\psi}{\theta}, \quad ln(0) = 0, \quad ln \equiv Ln/L_\odot.$$

La fonction de répartition utilisée est alors :

$$Q \equiv a \xi + b \mu + c ln$$

L'approximation initiale de $\psi(t_0) = \left(\frac{\partial Q}{\partial q}\right)_t$ s'obtient en dérivant numériquement $Q(\mu, t)$ (obtenu à partir de la solution initiale provisoire) par rapport à la nouvelle variable indépendante : la fonction de discrétisation $q(\mu, t)$.

Avec la variable indépendante $\nu = m^{\frac{1}{3}}$ on a : $\psi(t) = \left(\frac{\partial Q}{\partial q}\right)_\nu$ et $\theta(\nu, t) = \left(\frac{\partial Q}{\partial \nu}\right)_t$ et le problème différentiel s'écrit :

$$\begin{aligned} \frac{\partial \xi}{\partial q} &= \left[-\frac{3G M_\odot^2 \nu^5}{4\pi R_\odot^4 r^4} + \frac{M_\odot \nu^2}{2\pi R_\odot r} \Omega^2 \right] \exp(-\xi) \frac{\psi}{\theta} \\ \frac{\partial \eta}{\partial q} &= \frac{\partial \xi}{\partial q} \nabla \\ \frac{\partial r}{\partial q} &= \frac{3 M_\odot \nu^2 \psi}{4\pi R_\odot^3 \rho r^2 \theta} \\ \exp(\xi) &= \exp(\xi_g) + P_{\text{tur}} \\ \gamma &= \ln \rho(\exp(\xi_g), \exp(\eta), X_i) \\ \frac{\partial l}{\partial q} &= \frac{3M_\odot}{L_\odot} \nu^2 \left[\epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} \right] \frac{\psi}{\theta} \\ \frac{\partial \ln}{\partial q} &= \frac{3M_\odot}{L_\odot} \nu^2 \epsilon \frac{\psi}{\theta} \\ \frac{\partial \nu}{\partial q} &= \frac{\psi}{\theta} \\ \frac{\partial \psi}{\partial q} &= 0 \\ \frac{\partial X_i}{\partial t} &= -\frac{1}{3M_\odot \nu^2} \frac{\partial F_i}{\partial \nu} + \Psi_i(\xi_t, \eta; \mathcal{X}), \quad i = 1, \dots, n_{\text{elem}} \end{aligned}$$

$$\left\{ \begin{array}{l} \text{conditions limites} \\ \nu(n, t) = \nu_{\text{ext}}(l(n, t)L_\odot, r(n, t)R_\odot), \\ \xi(n, t) = \ln(P_{\text{ext}}(l(n, t)L_\odot, r(n, t)R_\odot)), \\ r(n, t) = \ln(T_{\text{ext}}(l(n, t)L_\odot, r(n, t)R_\odot)), \\ \xi_g(n, t) = \xi(n, t), \quad \gamma(n, t) = \ln \rho(P_g(n, t), T(n, t), X_i), \\ r(1, t) = 0, \quad l(1, t) = 0, \quad \ln(1, t) = 0, \quad \mu(1, t) = 0. \\ \text{conditions initiales} \\ X_i(q, 0) = X_{i,0}, \quad i = 1, \dots, n_{\text{elem}}, \\ \xi(q, 0) = \xi_0(q), \quad r(q, 0) = r_0(q), \quad \gamma(q, 0) = \gamma_0(q), \\ r(q, 0) = r_0(q), \quad l(q, 0) = l_0(q), \quad \ln(q, 0) = l_0(q) \\ 1 \leq q \leq n. \end{array} \right.$$

Ces équations de l'espace physique sont formées dans la routine `static_m13`, cf. § 8.126 (Page 232).

FIGURE 7.3 – Deux itérations ont été nécessaires pour positionner un point de grille sur une limite zone radiative / zone convective. Au cours du processus itératif, le gradient radiatif ∇_{rad} a légèrement diminué, et la limite s'est déplacée vers la droite, alors que le gradient adiabatique ∇_{ad} restait inchangé. Les points de grille se sont successivement déplacés de leurs positions initiales, carrés vides, vers leurs positions finales, cercles vides puis cercles pleins.

7.2.5 Nœud sur une limite zone radiative / zone convective

A chaque limite entre une zone convective et une zone radiative le gradient :

$$\nabla \equiv \left(\frac{\partial \ln T}{\partial \ln P} \right)_t$$

n'est pas dérivable. Il en est donc de même de la dérivée première de la température. Entre deux points de la grille d'intégration, les inconnues sont représentées par des polynômes par morceaux. Une limite entre une zone radiative et une zone convective doit donc nécessairement se trouver sur une maille du réseau. Avec CESAM2k il suffit de placer un point de grille à mieux que $\sim 5\%$ de la distance entre les nœuds entourant immédiatement chaque limite entre une zone convective et une zone radiative, il n'y alors pas de risque d'approcher les fonctions inconnues par un morceau de polynôme traversant une discontinuité, les points de collocation étant situés à une distance des points de grille, supérieure à 5% de la largeur de la maille concernée.

Pour assurer cette disposition, on utilise un paramètre libre de la fonction de répartition. D'une maille à l'autre, la variation de la fonction de répartition $Q(\mu, t)$, est constante. Les facteurs de répartition a et b définissent les poids relatifs des variables à l'intérieur de chaque maille, ces poids sont définis à un facteur multiplicatif près. Jusqu'à présent ce paramètre libre a été, implicitement, posé égal à l'unité. La modification de cette valeur pour les mailles situées au voisinage d'une limite entre une zone radiative et une zone convective va permettre de rejeter la limite dans un voisinage immédiat des points de grille. On définit la fonction en escalier $\omega(q)$, ajustable d'une maille à l'autre par :

$$Q(\mu, t) = \omega(q)Q_0(\mu, t) \quad (7.7)$$

$Q_0(\mu, t)$ notant maintenant la fonction de répartition retenue, c'est à dire celle définie par la relation Eq. 7.4 (Page 87). Tout en respectant la constante du saut de $Q(\mu, t)$ d'une maille à l'autre, $\omega(q)$ permet d'ajuster le saut de Q_0 dans chaque maille, donc les sauts de ξ , η , ζ , λ et μ de façon à ce qu'ils correspondent à la quantité nécessaire pour atteindre exactement la limite. Une fois la condition :

$$\frac{\partial^2 Q}{\partial q^2} = \frac{\partial \psi}{\partial q} = 0$$

satisfaite en tous points du réseau, dans une maille donnée, la variation de la fonction Q étant constante :

$$\frac{\partial Q}{\partial q} = \psi = cte = \frac{\partial \omega Q_0}{\partial q} = \omega \frac{\partial Q_0}{\partial q} \Rightarrow \omega = \psi / \frac{\partial Q_0}{\partial q} \sim \psi / \frac{\Delta Q_0}{\Delta q} = \psi / \Delta Q_0$$

puisque d'une maille à la suivante ⁶ $\Delta q = 1$. Pour chaque maille $[i, i+1]$, $i = 1, \dots, n-1$, on détermine ω_i de façon à ce que le saut de Q_0 :

$$\Delta Q_0 = a(\xi_{i+1} - \xi_i) + b(\mu_{i+1} - \mu_i)$$

6. C'est cette dernière relation qui définit $\omega(q)$, une fonction en escalier n'étant pas dérivable! Pour faciliter l'exposé on a préféré introduire $w(q)$ par Eq. 7.5 (Page 90).

correspondre à ce qu'il est nécessaire, pour que chaque limite entre une zone convective et une zone radiative se place sur un point du réseau⁷.

Malheureusement il n'y a pas de relation permettant de prévoir le déplacement d'une limite entre une zone convective et une zone radiative en fonction des variations des variables locales, si bien qu'il est nécessaire de procéder par ajustements successifs, il en résulte que la convergence du processus itératif global ne peut être que du premier ordre.

A chaque itération de la méthode de Newton-Raphson, la limite entre une zone radiative et une zone convective est localisée par interpolation linéaire en fonction de la variable d'indice q . Suivant la position interpolée pour les limites, les mailles concernées sont agrandies ou resserrées. A l'aide de la valeur obtenue pour la constante $\psi(t)$ et des valeurs de la variable Q_0 , de nouvelles valeurs de $\omega(q)$ sont calculées pour chaque maille. Un exemple d'un tel ajustement est illustré Fig. 7.3 (Page 89).

Dans la routine `lim_zc`, cf. § 8.70 (Page 202), les limites zones radiatives / zones convectives sont localisées avec précision, ce qui permet de déterminer les poids à affecter à chaque maille; l'expérience des calculs montre que le positionnement de la limite s'effectue sur un nœud avec une précision supérieure à 1%, si la limite est bien définie. Quand la transition zone radiative / zone convective est floue *i.e.* quand les gradients adiabatique et radiatif sont proches au voisinage de la limite, le positionnement n'est pas précis cf. § 6.2 (Page 70). Dans ces cas, la discontinuité de la dérivée du gradient est *petite* et l'erreur qui en résulte n'est importante ni pour les variables de structure ni pour la composition chimique qui ne peut, avec une limite floue, que présenter une discontinuité peu marquée.

7.3 Restitution de l'atmosphère

La lumière parvenant des étoiles est la principale source d'informations permettant d'induire leur structure. Un modèle d'évolution stellaire doit décrire le transfert du rayonnement à travers l'atmosphère. Bien que cette dernière ne représente qu'une infime partie de l'étoile, le calcul de ce transfert est un problème complexe qui ne peut être raisonnablement résolu simultanément avec celui de la structure interne. Une des difficultés réside dans le fait que c'est dans l'atmosphère que le rayonnement transite du milieu opaque de l'intérieur où il est quasi isotrope, vers le milieu inter-stellaire où il est anisotrope. La nécessité est de transporter les conditions limites du milieu transparent, où elles sont définies par l'observation, vers le milieu épais de l'intérieur. La restitution de l'atmosphère consiste alors à construire un modèle aussi précis que possible, assurant le transfert de l'énergie lumineuse et l'équilibre quasi-statique.

Les conditions limites externes du problème différentiel de la structure interne portent sur deux des trois variables thermodynamiques P , T et ρ . `CESAM2k` offre la possibilité d'utiliser deux types de méthodes : l'approximation monocouche et la reconstitution de l'atmosphère, cette dernière constituant un problème différentiel aux limites résolu numériquement par collocation, (Morel et al., 1994).

7. Des relations similaires peuvent être obtenues avec les facteurs de répartition $\Delta\xi$, $\Delta\eta$, $\Delta\zeta$, $\Delta\lambda$.

7.3.1 L'approximation monocouche

Le "Solar Model Comparison Project" (Christensen-Dalsgaard, 1988), utilise une atmosphère simplifiée pour transporter les conditions limites externes dans le milieu optiquement épais; elle est obtenue à l'aide de deux relations déduites d'une discrétisation simple des équations de l'atmosphère :

$$\left\{ \begin{array}{l} \frac{dP}{d\tau} = \frac{GM_\star}{R_\star^2\kappa} - \frac{2\Omega^2 R_\star}{3\kappa}, \quad \frac{\Delta P}{\Delta\tau} \sim \frac{P-P_{\text{ext}}}{\tau-0} \\ L_\star = 4\pi R_\star^2 \sigma T^4 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} P = \frac{GM_\star}{R_\star^2\kappa} - \frac{2\Omega^2 R_\star}{3\kappa} \\ L_\star = 4\pi R_\star^2 \sigma T^4 \end{array} \right.$$

sur la couche externe on a alors $T_{\text{ext}} = T_{\text{eff}}$; σ est la constante de Stéfán, R_\star , M_\star , Ω et L_\star sont respectivement, le rayon, la masse, la vitesse angulaire et la luminosité de l'étoile. Les deux dernières relations étant obtenues avec $\tau = 1$ et $P_{\text{ext}} = 0$ ce qui, physiquement, peut être interprété en disant que les conditions limites externes sont écrites à $\tau = 1$. Cette formulation est utilisée dans la routine `lim_tau1`, cf. § 8.69 (Page 201). L'évolution est arrêtée si l'accélération centrifuge devient supérieure à 90% de la gravité – ce qui se traduirait par une pression négative.

Dans la routine `lim_gong1`, cf. § 8.69 (Page 201), deux facteurs $\beta = 7.22$ et $\lambda = 6$, affectent respectivement les équations de la pression et de la luminosité, leur effet étant, avec une équation d'état supposant l'ionisation totale, d'obtenir un modèle solaire qui ressemble au soleil actuel.

PB Avec l'approximation monocouche, la pression turbulente est ignorée dans l'atmosphère.

7.3.2 Reconstitution d'une atmosphère

Dans les cas où une grande précision doit être atteinte, en particulier pour les modèles solaires servant au calcul des oscillations, il n'est pas possible d'ignorer la structure fine des couches externes où se réfléchissent les modes d'oscillation. Il est nécessaire de reconstituer le moins mal possible cette couche limite; c'est l'objet de la routine `lim_atm`, cf. § 8.68 (Page 201). Dans un calcul de structure interne, il n'est pas réaliste de traiter le transfert radiatif avec toute sa complexité. L'approximation de diffusion, cas gris, utilisé dans la structure interne, n'est justifiée que pour des profondeurs optiques Rosseland $\tau_{\text{rac}} \geq 20$, limite au delà de laquelle, cf. Morel et al. (1994), le flux radiatif calculé par l'approximation de diffusion et le flux réel ne diffèrent que par moins de 1%. Pour éviter de traiter explicitement l'atmosphère, on transporte de $\tau \sim 0$ à $\tau = \tau_{\text{rac}}$ les conditions externes *i.e.* celles au raccord avec le milieu interstellaire, à savoir : la densité ou la pression, la température et la masse. Avec `CESAM2k`, afin de relier le modèle de l'atmosphère et celui de la structure interne, il faut disposer pour la pression, la température et la masse de trois fonctions⁸ au point de raccord en $\tau = \tau_{\text{rac}}$ *i.e.* à la limite externe de l'enveloppe :

$$P_{\text{rac}}(R, L), T_{\text{rac}}(R, L), M_{\text{rac}}(R, L),$$

avec $M_{\text{rac}}(R, L) \simeq M_{\text{tot}}$ masse contenue à l'intérieur de la couche la plus externe de l'enveloppe. Dans l'atmosphère, la luminosité L est supposée constante $L =$

8. On assurera ensuite, par itérations, l'égalisation de ces valeurs résultant de l'intégration de l'atmosphère et de celles résultant de l'intégration de la structure interne.

L_* . La composition chimique, la vitesse angulaire qui, éventuellement, varient en fonction du temps *e.g.* par diffusion et mélange convectif, sont supposées constantes et fixées à leur valeurs sur la couche la plus extérieure de l'enveloppe. Pour une longueur de mélange et une composition chimique fixées, la température est donnée dans l'atmosphère par une loi $T(\tau)$ de la forme : $T(\tau_R, T_{\text{eff}}, g)$ dépendant, de la profondeur⁹ optique Rosseland τ_R , $d\tau = -\kappa\rho dR$, de la température effective T_{eff} et de la gravité g :

$$T_{\text{eff}} = \left(\frac{L}{4\pi R_{\text{tot}}^2 \sigma} \right)^{\frac{1}{4}}, \quad g = \frac{GM}{R^2} \cong \frac{GM_{\text{tot}}}{R_{\text{tot}}^2},$$

R_{tot} est le rayon au point de raccord avec le milieu interstellaire *cf.* Mihalas (1978) p. 49, et κ est l'opacité.

Cette définition de la température effective correspond à la température du corps noir rayonnant la même densité d'énergie que l'étoile, elle n'est pas universelle. Une seconde définition, souvent utilisée, permet de définir un "rayon observable" R_* :

$$R_* = R(\tau = \frac{2}{3}) \text{ et } T_{\text{eff}} = \left(\frac{L}{4\pi\sigma R_*^2} \right)^{\frac{1}{4}}.$$

Dans ces conditions, la masse $M_* < M_{\text{tot}}$ est celle à la profondeur optique $\tau = 2/3$: $M_* = M(\tau = 2/3)$.

Dans CESAM2*k*, une troisième définition de T_{eff} , *cf.* Schatzman & Praderie (1990), est utilisée : le rayon de l'étoile R_* est défini à la profondeur optique à laquelle la température est égale à la température effective, R_* est le "rayon bolométrique". Pour certaines lois $T(\tau)$, les deux dernières définitions sont équivalentes. Par la suite, on notera τ_* la profondeur optique où le rayon est celui de l'étoile : $R(\tau_*) = R_*$.

La composition chimique, l'opacité et la définition retenue pour la température effective doivent, évidemment, être les mêmes dans :

- le programme de calcul du modèle d'atmosphère à partir duquel est déduite la loi $T(\tau, T_{\text{eff}}, g)$,
- la partie atmosphère du programme de structure interne,
- la partie supérieure de l'enveloppe.

Connaissant R_{rac} et L on peut intégrer de $\tau = \tau_{\text{min}}$ à $\tau = \tau_{\text{rac}}$ où on obtient, au rayon R_{rac} , point de raccord entre l'enveloppe et l'atmosphère, les trois quantités :

$$P_{\text{rac}}(R, L), T_{\text{rac}}(R, L), M_{\text{rac}}(R, L).$$

En utilisant τ comme variable indépendante, le système des équations à intégrer

9. Pour plus de lisibilité l'indice R pour "Rosseland" est omis dans la suite.

est :

$$\left\{ \begin{array}{l} \frac{dP}{d\tau} = \frac{GM}{\kappa R^2} - \frac{2\Omega^2 R}{3\kappa} \\ \frac{dT}{d\tau} = \frac{T}{P} \frac{dP}{d\tau} \nabla \\ \frac{dR}{d\tau} = -1 / \kappa \rho \\ \frac{dM}{d\tau} = -4\pi R^2 / \kappa \\ \frac{dR_\star}{d\tau} = 0 \\ \frac{d\tau_\star}{d\tau} = 0 \\ P_{\text{gaz}} = P \left\{ 1 - \frac{a\alpha^2\delta}{8} \frac{\Gamma}{\Gamma+1} (\nabla - \nabla_{\text{ad}}^*) \right\} \end{array} \right. \left\{ \begin{array}{l} \text{conditions limites} \\ \rho(\tau_{\text{min}}) = \rho_{\text{ext}} \\ \text{ou } P(\tau_{\text{min}}) = g\kappa\tau_{\text{min}} \\ P_{\text{gaz}}(\tau_{\text{min}}) = P(\tau_{\text{min}}) \\ M(\tau_\star) = M_\star \\ R(\tau_\star) = R_\star \\ T_{\text{eff}} = \mathcal{T}(\tau_\star, T_{\text{eff}}, g_\star) \\ T(\tau_\star) = \mathcal{T}(\tau_\star, T_{\text{eff}}, g_\star) \\ R(\tau_{\text{rac}}) = R_{\text{rac}} \end{array} \right. \quad (7.8)$$

Les deux équations différentielles et les deux conditions limites portant sur R_\star et τ_\star ont été ajoutées pour résoudre le problème différentiel avec la limite libre à $\tau = \tau_\star$ cf. Stoer & Bulirsch (1979, par. 7.3.0), elles couplent R_\star et τ_\star aux autres variables. En codant `lim_ro=.TRUE.` dans le fichier de données, à la limite optiquement mince, définie par $\tau = \tau_{\text{min}}$, la condition limite porte sur la densité fixée à la valeur ρ_{ext} donnée par le modèle d'atmosphère ; avec `lim_ro=.FALSE.`, la condition limite porte sur la pression ; on utilise alors la relation dérivée de l'approximation monocouche. La condition sur la pression permet de s'affranchir de la contrainte sur la densité à une profondeur optique fixée arbitrairement. En général on fixe $\tau_{\text{min}} \equiv 10^{-4}$ ce qui correspond, approximativement, au minimum de température dans la photosphère solaire, cela ne constitue évidemment pas une règle absolue¹⁰. A ce niveau, l'atmosphère est supposée radiative si bien que $P_{\text{gaz}} \equiv P$.

La dernière équation qui relie P et P_{gaz} est celle qui est vérifiée dans les zones radiatives où l'efficacité de la convection est nulle *i.e.* $\Gamma \equiv 0$.

Les conditions limites sont donc écrites aux trois niveaux : $\tau = \tau_{\text{rac}}$, $\tau = \tau_\star(t)$ et $\tau = \tau_{\text{min}}$, en se rappelant que $\tau_\star(t)$ est susceptible de varier au cours du temps. De plus, ρ_{ext} dépend aussi du temps, puisque dépendant de la gravité $\rho_{\text{ext}}(g_\star(t))$.

Au temps t , la solution des équations Eq. 7.6 (Page 93) donne P_{rac} , T_{rac} et M_{rac} pour R et L donnés.

Evidemment, autant que faire se peut, la loi $T(\tau)$ et la structure interne doivent être calculées avec des physiques aussi équivalentes que possible à savoir : équation d'état, opacité, longueur de mélange, composition chimique *etc...* Malgré de nombreuses tentatives, à cause des incohérences entre la physique, les données, les paramètres, utilisés dans le modèle d'atmosphère d'une part, et dans la structure interne d'autre part, il s'est avéré impossible de raccorder non seulement les fonctions (*i.e.* pression, température, masse), mais aussi leurs gradients à la limite entre atmosphère et enveloppe ; si le gradient de pression *i.e.* $\frac{dP}{dr} = -g\rho$, se raccorde parfaitement, il n'en est pas de même du gradient de température ni, par conséquent, du gradient $\nabla \equiv \frac{d \ln T}{d \ln P}$; des différences de l'ordre de 0.1 à 0.2 pouvant exister de part et d'autre du raccord. Pour certaines applications de telles différences sont prohibitives et, au risque de ne pas vérifier parfaitement la loi $T(\tau)$, un artifice numérique semi-original a été mis au point pour assurer aussi le raccord de ∇ .

10. Il faut s'assurer que l'équation d'état et les tables d'opacité donnent des valeurs réalistes pour les conditions physiques auxquelles on peut s'attendre jusqu'à τ_{min} .

PB

La restitution de l'atmosphère n'est effectuée que si l'accélération centrifuge est inférieure à 90% de la gravité; si tel n'est pas le cas, l'évolution est arrêtée.

7.3.3 Artifices numériques utilisés pour le raccord de ∇

La gravité g étant fixée, la loi $T(\tau)$ est supposée pouvoir s'écrire sous la forme : $T^4 = \frac{3}{4}T_{\text{eff}}^4 f(\tau)$; en utilisant l'expression de la luminosité, $L = 4\pi R_\star^2 \sigma T_{\text{eff}}^4$, il vient :

$$T^4 = \frac{3}{4\pi ac} \frac{L_\star}{R_\star^2} f(\tau), \quad (\sigma = \frac{ac}{4}), \quad \text{d'où} \quad \frac{dT}{d\tau} = \frac{3}{16\pi ac} \frac{L_\star}{R_\star^2 T^3} \frac{df}{d\tau},$$

si bien qu'avec $d\tau = -\kappa\rho dr$:

$$\begin{aligned} \nabla \equiv \frac{d \ln T}{d \ln P} &= -\frac{P}{T} \frac{dT}{d\tau} \kappa\rho \frac{dr}{dP} = \frac{P\kappa}{T} \frac{dT}{d\tau} \frac{R^2}{GM} = \frac{3}{16\pi acG} \frac{P\kappa L_\star}{MT^4} \left(\frac{R}{R_\star}\right)^2 \frac{df}{d\tau} = \\ &= \nabla_{\text{rad}} \frac{L}{L_\star} \left(\frac{R}{R_\star}\right)^2 \frac{df}{d\tau} \end{aligned} \quad (7.9)$$

puisque le gradient radiatif a pour expression :

$$\nabla_{\text{rad}} = \frac{3}{16\pi acG} \frac{P\kappa L}{MT^4}.$$

Ici $L \equiv L_\star$ et, dans le cas solaire $R/R_\star < 1 \pm 10^{-3}$. Dans une zone convective le flux total F est la somme du flux radiatif et du flux convectif : $F = F_{\text{rad}} + F_{\text{conv}}$. *Si et seulement si* l'approximation de diffusion est vérifiée *i.e.* à la limite $\tau \gg 1$, le flux radiatif $F_{\text{rad}}(\tau) = \int_0^\infty F_\nu d\nu$ a pour expression :

$$F_{\text{rad}} = \frac{4acT^4}{3\kappa\rho H_p} \nabla$$

avec pour le flux total F :

$$F = \frac{4acT^4}{3\kappa\rho H_p} \nabla_{\text{rad}}$$

alors :

$$\lim_{\tau \gg 1} \frac{F_{\text{rad}}}{F} = \left(\frac{R}{R_\star}\right)^2 \frac{df}{d\tau} \sim \frac{df}{d\tau},$$

puisque dans l'atmosphère, du moins pour les étoiles naines, le quotient $(R/R_\star)^2 \sim 1 \pm 10^{-8}$ peut être pris égal à l'unité.

L'artifice numérique utilisé pour le raccord de ∇ repose sur la formulation Eq. 7.7 (Page 94) : pour le calcul de la température, au lieu d'utiliser explicitement la loi $T(\tau)$, on intègre l'équation différentielle :

$$\frac{dT}{d\tau} = \frac{T}{P} \frac{dP}{d\tau} \nabla = \frac{T}{P} \frac{dP}{d\tau} \nabla_{\text{rad}} \left(\frac{R}{R_\star}\right)^2 \frac{df}{d\tau},$$

et une condition limite sur la température, cette dernière étant écrite, *par exemple*, en τ_\star :

$$T(\tau_\star) = T(\tau_\star, T_{\text{eff}}, g_\star).$$

Ainsi, à un changement de variable près, les équations *sont les mêmes* dans l'atmosphère restituée et dans la structure interne.

Pour des lois $T(\tau)$ purement radiatives, telles que celles d'Eddington ou de Hopf, $\lim_{\tau \gg 1} \frac{df}{d\tau} = 1$. On remarque qu'alors, le flux total est posé égal au flux radiatif même si l'approximation de diffusion est injustifiée. Pour des lois $T(\tau)$ incorporant la convection, telles que celles dérivées de modèles d'atmosphères solaires, $\lim_{\tau \gg 1} \frac{df}{d\tau} < 1$ (typiquement $\frac{df}{d\tau} \sim 0.2$ à $\tau = 20$). Il faudra donc un traitement différent suivant que la loi $T(\tau)$ est, ou non, purement radiative.

7.3.4 Loi $T(\tau)$ purement radiative

La loi $T(\tau)$ ne contient pas la partie convective de l'atmosphère, force est donc d'utiliser le formalisme de la structure interne pour le calcul du gradient dans la partie convective, sachant qu'elle sera incorrecte tant que l'approximation de diffusion ne sera pas justifiée *i.e.* tant que $\tau \lesssim 20$. Dans la partie radiative le gradient aura l'expression Eq. 7.7 (Page 94), dans la partie convective, son expression sera celle déduite de la théorie de la convection utilisée *cf.* § 7.10.2 (Page 149),

la continuité du gradient sera assurée s'il est calculé avec des expressions qui ont mêmes limites de part et d'autre de la transition convectif/radiatif, ce qui est réalisé en utilisant, dans le formalisme de la longueur de mélange, un ∇_{rad} *modifié* :

$$\nabla_{\text{rad}}^* \equiv \nabla_{\text{rad}} \frac{df}{d\tau}.$$

Cet artifice numérique est dû à Henyey Henyey et al. (1965), M. Gabriel & J. Christensen-Dalsgaard, *communications privées*. A la limite $\tau \gg 1$, le gradient sera ainsi exactement celui donné par le formalisme de la longueur de mélange puisque, pour une loi purement radiative, $\lim_{\tau \gg 1} \frac{df}{d\tau} = 1$.

Lois $T(\tau)$ purement radiatives implantées dans CESAM2k :

- hopf : Loi $T(\tau)$ de Hopf, *cf.* § 8.58 (Page 197).
- edding : Loi $T(\tau)$ d'Eddington, *cf.* § 8.42 (Page 189).

7.3.5 Lois $T(\tau)$ avec convection

L'artifice numérique précédent n'est pas utilisable puisque $\lim_{\tau \gg 1} \frac{df}{d\tau} < 1$ et, par ailleurs, l'utilisation d'une loi $T(\tau)$ non purement radiative perdrait de son intérêt, puisque la description de la partie convective aux profondeurs optiques $\tau \sim 1$ serait effectuée avec une théorie de la convection qui utilise une expression du flux radiatif déduite de l'approximation de diffusion. Dans ce cas, pour assurer la continuité du gradient, l'artifice numérique utilisé consiste en une interpolation linéaire, en fonction de l'épaisseur optique, entre la valeur du gradient déduit de la loi $T(\tau)$:

$$\nabla = \nabla_{\text{rad}} \left(\frac{R}{R_\star} \right)^2 \frac{df}{d\tau} \quad (7.10)$$

et de celle ∇_{mel} calculée par la théorie de la longueur de mélange ainsi qu'il est décrit § 7.10.2 (Page 149). Ainsi, pour les profondeurs optiques $\tau \lesssim 1$ le gradient est proche de celui dérivé de la loi $T(\tau)$, tandis qu'il tend vers celui de la longueur

de mélange pour les profondeurs optiques $\tau \geq \tau_{\text{rac}}$. L'interpolation linéaire utilisée a pour expression :

$$\nabla = (1 - x)\nabla_{\text{rad}} \left(\frac{R}{R_{\star}} \right)^2 \frac{df}{d\tau} + x\nabla_{\text{mel}}, \quad x = \frac{\tau - 1}{\tau_{\text{rac}} - 1}. \quad (7.11)$$

Pour les profondeurs optiques intermédiaires $\tau \sim 1$, cet artifice numérique permet de restituer, en partie, le modèle d'atmosphère avec convection, tout en assurant, plus profond, le raccord des gradients malgré les incohérences de la physique.

PB Avec une loi $T(\tau)$ non purement radiative, la valeur $\tau_{\text{rac}} = 20$ est imposée indépendamment de celle indiquée pour τ_{max} dans le fichier de données.

Lois $T(\tau)$ non purement radiatives implantées dans CESAM2k :

- **k5750** : Loi $T(\tau)$ solaire $T_{\text{eff}} = 5750$ K calculée par C. Van't Veer avec Atlas 9 de Kurucz, *cf.* § 8.66 (Page 200).
- **k5777** : Loi $T(\tau)$ solaire $T_{\text{eff}} = 5777$ K calculée par C. Van't Veer avec Atlas 12 de Kurucz, *cf.* § 8.66 (Page 200).
- **roger00** : Lois $T(\tau)$ à $[\text{Fe}/\text{H}]=0$, calculs avec Atlas 12 de Kurucz, interpolations de R. Cayrel. Utilise la table `fresh00.data` du directory `DATA` de la source de CESAM2k; avant utilisation indiquer, dans le sous programme, le chemin d'accès de la table, *cf.* § 8.121 (Page 229).
- **roger02** : Lois $T(\tau)$ à $[\text{Fe}/\text{H}]=-0.2$, calculs avec Atlas 12 de Kurucz, interpolations de R. Cayrel. Utilise la table `fresh02.data` du directory `DATA` de la source de CESAM2k; avant utilisation indiquer, dans le sous programme, le chemin d'accès de la table, *cf.* § 8.121 (Page 229).
- **roger05** : Lois $T(\tau)$ à $[\text{Fe}/\text{H}]=-0.5$, calculs avec Atlas 12 de Kurucz, interpolations de R. Cayrel, *cf.* § 8.121 (Page 229).
- **roger10a** : Lois $T(\tau)$ à $[\text{Fe}/\text{H}]=-1.0$ enhanced, calculs avec Atlas 12 de Kurucz, interpolations de R. Cayrel, *cf.* § 8.121 (Page 229).

7.3.6 Résolution numérique

Pour résoudre numériquement le problème différentiel défini par le système Eq. 7.6 (Page 93), on impose (dans la routine `lim_atm`) un indice *fixe* $n_{\star} = 3/4n_a$ à la limite libre $\tau = \tau_{\star}(t)$ parmi les n_a points de grille que comporte l'atmosphère. Pour ce faire, on utilise la **bijection linéaire par morceaux** : $\tau \mapsto \omega$ de $[\tau_{\text{rac}}, \tau_{\text{min}}]$ dans $[1, n_a]$ définie par :

$$\ln \tau \mapsto \varphi(\omega) = \begin{cases} \ln \tau_{\text{rac}} + (\omega - 1)\Delta\varphi^+, & \text{si } \omega \in [1, n_{\star}]; \\ \ln \tau_{\text{min}} + (\omega - n_a)\Delta\varphi^-, & \text{sinon.} \end{cases} \quad (7.12)$$

Les deux valeurs de la pente $\Delta\varphi^+$ et $\Delta\varphi^-$ étant définies par :

$$\Delta\varphi^+ \equiv \frac{\ln \tau_{\star} - \ln \tau_{\text{min}}}{n_{\star} - n_a}, \quad \Delta\varphi^- \equiv \frac{\ln \tau_{\star} - \ln \tau_{\text{rac}}}{n_{\star} - 1};$$

ainsi on a :

$\omega = 1$, au fond de l'atmosphère où $\tau = \tau_{\text{rac}}$,

$\omega = n_*$, sur la limite libre interne à $\tau = \tau_*$,

$\omega = n_a$, sur la limite externe où $\tau = \tau_{\min}$.

On travaille donc avec un pas de grille constant, ce qui est un avantage de la méthode. Pour l'intégration numérique, on a utilisé les variables : $\xi = \ln P$, $\xi_g = \ln P_{\text{gaz}}$, $\eta = \ln T$, $\zeta_a = R/R_\odot$, $\mu_a = M/M_\odot$, $\zeta_* = R_*/R_\odot$ et $\varphi_* = \ln \tau_*$, si bien que le système Eq. 7.6 (Page 93) devient :

$$\left\{ \begin{array}{l} \frac{d\xi}{d\omega} = \left[\frac{GM_\odot \mu_a}{R_\odot^2 \zeta_a^2} - \frac{2R_\odot \Omega^2 \zeta_a}{3} \right] \frac{\Delta\varphi^\pm}{\kappa} \exp(\varphi - \xi) \\ \frac{d\eta}{d\omega} = \frac{d\xi}{d\omega} \nabla \\ \frac{d\zeta_a}{d\omega} = -\frac{1}{R_\odot} \frac{\Delta\varphi^\pm \exp(\varphi)}{\kappa \rho} \\ \frac{d\zeta_*}{d\omega} = 0 \\ \frac{d\mu_a}{d\omega} = -\frac{4\pi R_\odot^2}{M_\odot} \frac{\Delta\varphi^\pm}{\kappa} \exp(\varphi) \zeta_a^2 \\ \frac{d\varphi_*}{d\omega} = 0 \\ \frac{d\varphi}{d\omega} = \Delta\varphi^\pm \\ \exp(\xi) \left\{ 1 - \frac{a\alpha^2 \delta}{8} \frac{\gamma}{\gamma+1} (\nabla - \nabla_{\text{ad}}^*) \right\} - \exp(\xi_g) = 0 \end{array} \right. \quad (7.13)$$

avec $\Delta\varphi^\pm = \Delta\varphi^+$ (*respt.* $\Delta\varphi^-$) si $\omega \in [1, n_*]$ (*respt.* $\omega \in [n_*, n_a]$).

Les conditions limites sont alors :

$$\left\{ \begin{array}{l} \zeta_a(1) = \zeta_{\text{rac}} \\ \mu_a(n_*) = M_*/M_\odot \\ \zeta_*(n_*) = R_*/R_\odot \\ \eta(n_*) = \ln \mathcal{T}(\exp(\varphi_*), T_{\text{eff}}, g_*) \\ \eta(n_*) = \ln T_{\text{eff}} \\ \varphi(n_a) = \ln \tau_{\min} \\ \rho(n_a) = \rho_{\text{ext}} \\ \xi(n_a) = \xi_g(n_a) \end{array} \right.$$

avec :

$$T_{\text{eff}}^4 = \frac{L_\odot \sqrt{\lambda_{\text{rac}}^3}}{4\pi R_\odot^2 \sigma \zeta_*^2}, \quad g = \frac{GM_\odot \mu_a}{R_\odot^2 \zeta_a^2}.$$

La solution numérique de ce problème différentiel non-linéaire utilise la méthode de spline-collocation. Le processus itératif est initialisé, lors du premier calcul, par un modèle d'atmosphère solaire et, pendant l'évolution, par le modèle d'atmosphère du modèle au pas de temps précédent.

PB A cause de la discontinuité de la dérivée de la bijection linéaire par morceaux *cf.* §6.2 (Page 70), au point $\tau = \tau_*$, il semble se produire des difficultés de convergence lorsque τ_* tente de franchir la discontinuité, un grand nombre d'itérations est alors nécessaire. En fait, une propriété de la méthode de spline-collocation permet d'atteindre effectivement la solution.

Typiquement, on utilise de $n_a = 30$ à $n_a = 100$ points de grille pour l'intervalle $[\tau_{\text{rac}}, \tau_{\text{min}}]$. Avec des B-splines d'ordre 3 il y a de 59 à 199 points d'intégration où les Eqs. (7.11) sont satisfaites, en tenant compte de la superconvergence, l'ordre du schéma est 4.

Ces équations sont résolues dans l'espace des splines par la routine `coll_atm`, cf. §8.15 (Page 176), en utilisant les coefficients calculés dans l'espace physique par la routine `eqatm` cf. §8.43 (Page 189). La gestion et l'interface entre les deux espaces étant assurée par la routine `lim_atm`, cf. §8.68 (Page 201). Avec les lois $T(\tau)$ déduites de modèles d'atmosphère sophistiqués, la profondeur optique τ_* varie avec la température effective et la gravité, l'équation implicite $T(\tau_*) = T_{\text{eff}}$ est alors résolue dans la routine `taueff`, cf. §8.128 (Page 234).

Dans le programme de résolution, le point d'indice 1 correspond au raccord à $\tau = \tau_{\text{rac}}$ et $R = R_{\text{rac}}$, le point d'indice n_a à $\tau = \tau_{\text{min}}$. La densité externe ρ_{ext} à $\tau = \tau_{\text{min}}$ dépend de la loi $T(\tau)$ utilisée. La profondeur optique $\tau = \tau_{\text{rac}}$ du fond de l'atmosphère est un paramètre externe introduit dans la NAMELIST `NL-ATM` du fichier de données : $\tau_{\text{rac}} = \text{tau_max}$.

La solution consiste en $\xi_{\text{rac}}(\zeta_{\text{rac}}, \lambda_{\text{rac}})$, $\eta_{\text{rac}}(\zeta_{\text{rac}}, \lambda_{\text{rac}})$, $\mu_{\text{rac}}(\zeta_{\text{rac}}, \lambda_{\text{rac}})$; les dérivées par rapport à η_{rac} et λ_{rac} sont calculées numériquement.

7.4 Evolution temporelle de l'énergie gravifique

L'énergie gravifique a un rôle prépondérant dans les parties de l'étoile où elle se substitue à l'énergie thermonucléaire, lorsque le fuel nucléaire a disparu, par exemple dans le cœur, après la séquence principale ou encore dans la phase d'Hayashi. C'est cette équation qui, traduisant l'hypothèse d'équilibre quasi-statique, évite un traitement hydrodynamique de l'évolution stellaire. Elle n'est pas justifiée dans toutes les phases de l'évolution.

7.4.1 L'approximation de Kippenhahn

Bien qu'elle soit formellement incorrecte cf. Cox & Giuli (1968); Strittmatter et al. (1970), à une abscisse lagrangienne où la variation temporelle du nombre de particules libres n'est pas nulle *e.g.* dans un cœur convectif, l'approximation de Kippenhahn et al. (1968) s'écrit :

$$\epsilon_G \sim T \left(\frac{\partial S}{\partial t} \right)$$

où S est l'entropie spécifique. Cette approximation permet d'exprimer l'énergie gravifique explicitement en fonction des variations de pression et de température :

$$\epsilon_G \sim T \left(\frac{\partial S}{\partial t} \right) = c_P \frac{\partial T}{\partial t} - \frac{\delta}{\rho} \frac{\partial P_{\text{gaz}}}{\partial t}$$

est d'un emploi facile. Pour les étoiles de masse ($M \geq 1.2M_{\odot}$) l'extension du cœur convectif varie au cours de l'évolution, créant ainsi une discontinuité de la composition chimique donc de la densité, suivant l'axe temporel; discontinuité dont il y

a lieu de tenir compte dans l'évaluation de la dérivée temporelle de la densité dans Eq. 7.12 (Page 100). L'approximation de Kippenhahn qui n'utilise pas explicitement la dérivée d'une fonction discontinue permet de contourner la difficulté et de faire supporter l'erreur par le numérique.

7.4.2 Discrétisation

La variation d'énergie gravifique :

$$\epsilon_G = \left(\frac{\partial U}{\partial t} \right)_\mu + P \left(\frac{\partial V}{\partial t} \right)_\mu$$

où U est l'énergie interne spécifique, $V = 1/\rho$ le volume spécifique, et μ la variable d'espace lagrangienne, est approchée dans CESAM2k par un schéma aux différences finies implicite :

$$\epsilon_G \simeq \frac{U^{t+\Delta t} - U^t}{\Delta t} - \frac{P^{t+\Delta t}}{(\rho^{t+\Delta t})^2} \frac{\rho^{t+\Delta t} - \rho^t}{\Delta t} \quad (7.14)$$

où les quantités U^t et ρ^t à l'instant t sont calculées en utilisant l'équation d'état avec des valeurs de la pression et de la température obtenues par interpolation à μ *i.e.* masse constante, de la solution précédente :

$$\mu \leftrightarrow q \leftrightarrow P^t, T^t$$

et, pour la composition chimique X , par une interpolation directe par rapport à μ , d'ordre mc, ordre des B-splines d'interpolation de la composition chimique.

Avec l'approximation de Kippenhahn, la formulation aux différences finies totalement implicite est écrite :

$$\epsilon_G \simeq c_P^{t+\Delta t} \frac{T^{t+\Delta t} - T^t}{\Delta t} - \frac{\delta^{t+\Delta t}}{\rho^{t+\Delta t}} \frac{P_{\text{gaz}}^{t+\Delta t} - P_{\text{gaz}}^t}{\Delta t} \quad (7.15)$$

Ces équations sont formées dans les routines `static_m` et `static_r` cf. §8.126 (Page 232).

7.4.3 Initialisation

Pour le calcul de la pré-séquence principale, il est nécessaire de disposer d'un modèle en équilibre quasi-statique dont la seule source d'énergie est d'origine gravitationnelle. Le modèle initial est obtenu par une méthode due à Iben (1965) qui a été adaptée à CESAM2k en collaboration avec A.Baglin. Au début de la pré-séquence principale l'étoile est complètement convective donc isentropique sauf, peut-être, dans une partie superadiabatique externe. Dans ces conditions, la composition chimique est spatialement et temporellement **constante**, et l'équation d'énergie s'écrit :

$$\frac{\partial L}{\partial m} = \epsilon_G = -T \frac{\partial S}{\partial t} = cT$$

où c est la "**constante de contraction**" qui caractérise un modèle en équilibre quasi-statique, totalement convectif et sans réactions thermonucléaires. La donnée

de la constante de contraction c , permet de résoudre le système des équations au temps $t + dt$ sans avoir à déterminer explicitement la quantité $T \frac{\partial S}{\partial t}$.

La méthode de Iben consiste à construire un premier modèle (indice 1) avec une première valeur c_1 pour c . Puis un second modèle (indice 2) avec une valeur c_2 voisine – CESAM2*k* utilise $c_2 = 1.1 \times c_1$. Si on admet que ces deux modèles appartiennent à la même séquence évolutive, l'énergie rayonnée pendant l'intervalle de temps Δt les séparant est alors égale à la variation d'énergie gravitationnelle, si bien que :

$$\frac{L_1 + L_2}{2} \Delta t \sim \left(\frac{GM^2}{R_2} - \frac{GM^2}{R_1} \right) \Rightarrow \Delta t \sim 2GM^2 \frac{R_1 - R_2}{(L_1 + L_2)R_1R_2}.$$

On peut alors débiter le calcul de l'évolution en utilisant cette valeur de Δt et le modèle 2 comme modèle initial. Une valeur typique pour la constante est $c = 0.02$, la température centrale du modèle initial avoisine alors 100 000K ; on obtient 500 000K (*respt.* 1 000 000K) avec $c = 0.0005$ (*respt.* $c = 0.00008$). Ces ordres de grandeur dépendent assez peu de la masse du modèle. Au cours du déroulement du calcul CESAM2*k* demande la valeur de la constante de contraction c à utiliser. A chaque étape de la détermination du modèle initial de pré-séquence principale, il est possible de vérifier les valeurs de la température centrale, de la luminosité et du rayon total de façon à pouvoir modifier c jusqu'à obtenir le modèle désiré. L'initialisation pour le calcul du premier modèle peut être faite, soit à partir d'un modèle de pré-séquence principale calculé avec des paramètres voisins, fichier en binaire d'extension `.B.pms`, ou encore avec l'un des fichiers ASCII de modèle de pré-séquence principale `2d-2.pms`, `5d-4.pms`, `8d-5.pms` du sous-directory `EXPLOIT`. Une fois le modèle initial de pré-séquence principale obtenu celui-ci est écrit, en vue d'une autre initialisation, dans le fichier binaire `mon_modele.B.pms`. L'énergie gravimétrique initiale de pré-séquence principale est calculée par la routine `iben`, *cf.* § 8.59 (Page 197).

7.5 Evolution de la composition chimique sans diffusion

L'évolution temporelle de la composition chimique pose des difficultés particulières car, même sur la séquence principale où les termes non linéaires des équations n'ont pas encore de rôle déterminant, les équations d'évolution sont délicates à intégrer. Pour ne pas dégrader l'ordre de précision de la résolution des équations de l'équilibre quasi-statique, il faudrait que le schéma utilisé pour le problème de valeurs initiales soit d'ordre aussi élevé que possible. Même sans diffusion, l'analyse montre qu'il est difficile d'utiliser une formule d'intégration d'ordre supérieur à 2 qui satisfasse aux contraintes suivantes imposées par la physique de l'évolution stellaire :

- Coexistence d'échelles de temps d'évolution très différentes. Par exemple le deutérium a un temps d'évolution caractéristique de l'ordre de l'année, celui de l'hydrogène est de l'ordre d'une centaine de million d'années. Un tel problème différentiel est qualifié de "raide".
- Présence de zones convectives qui mélangent les espèces chimiques. Comme il est souhaitable d'utiliser en tout point du modèle la même formule d'intégration, on ne peut qu'utiliser des schémas à un ou deux pas. Au delà,

la logique de l'algorithme gérant le mélange convectif et le mouvement des limites devient très compliqué.

- Conservation de la charge, du nombre de nucléons, la formule d'intégration doit donc conserver toute combinaison linéaire des abondances, on dira par la suite qu'elle doit être "conservative".
- Les abondances sont des quantités positives.

Le problème de l'évolution temporelle de la composition chimique étant un problème différentiel raide, afin de justifier les choix faits dans CESAM2k, on donne quelques indications concernant la résolution numérique de ce type de problème ; pour plus d'informations consulter Hairer & Wanner (1991).

7.5.1 Problème raide

Pour intégrer numériquement un problème différentiel de Cauchy :

$$y'(x) = f(x, y), \text{ sur } [x_0, x_N], y(x_0) = y_0$$

on se donne un pas $h > 0$ et, de proche en proche, à partir de la valeur initiale y_0 , on obtient la solution numérique en utilisant une relation de la forme : $y_n = y_{n-1} + h\phi$, dans laquelle la fonction d'incrément ϕ dépend de façon plus ou moins compliquée de la solution obtenue aux points $x_i = x_0 + ih$, $0 \leq i \leq n$, $n \geq 1$. Ce faisant, deux notions se dégagent : la *convergence* et la *stabilité*. On dit que la formule est convergente si, en $x \in [x_0, x_N]$ fixé ($x = x_0 + nh$, $h = (x - x_0)/n$, $n \leq N$), la limite pour $h \rightarrow 0$ de la solution numérique coïncide avec la solution exacte du problème différentiel, soit :

$$\lim_{n \rightarrow \infty} (x_0 + nh) = x_N \Rightarrow \lim_{n \rightarrow \infty} y_n = y(x_N)$$

On montre que la plupart des formules classiques sont convergentes. La stabilité est reliée au comportement à l'infini de la solution numérique. Une formule d'intégration est stable si, le pas h étant fixé :

$$\lim_{n \rightarrow \infty} (x_0 + nh) = \infty \Rightarrow \lim_{n \rightarrow \infty} y_n = y(\infty)$$

La non stabilité est due à l'accumulation des erreurs de troncature. Si celles-ci augmentent moins vite que la solution, la formule est stable, instable sinon.

La stabilité dépend, non seulement de la formule d'intégration, mais aussi du problème différentiel posé. L'étude ne peut en être faite que pour des équations simples. Ainsi est-on obligé d'en restreindre la définition : pour qu'il y ait stabilité, il ne faut pas que les erreurs de troncature croissent exponentiellement à l'infini ; pour cette raison on ramène l'étude de la stabilité à celle de la solution numérique du problème type : $y'(x) = \lambda y(x)$, $\lambda \in C$. h étant fixé, la formule est stable si la solution numérique tend vers la solution nulle lorsque n tend vers l'infini. On dit alors que la formule est A-stable. De plus, on la qualifiera de A-positive si, dans le domaine où elle est A-stable et $\lambda \in C$, avec $y_0 > 0$, lorsque n tend vers l'infini, la solution numérique tend vers la solution nulle en restant positive.

De façon assez générale, les formules implicites sont A-stables dans de grandes régions du plan complexe mais peu d'entre elles sont A-positives.

Avec un problème différentiel raide dans lequel les échelles de temps d'évolution des diverses variables sont très différentes la difficulté est double :

- Obtenir une solution stable pour toutes les variables avec un pas temporel acceptable, c'est à dire sans suivre pour autant la plus petite échelle de temps.
- Intégrer les variables significatives avec une précision suffisante.

Pour l'évolution stellaire, à moins qu'on ne désire suivre en détail l'évolution de l'espèce chimique dont l'échelle de temps caractéristique est la plus petite, il faudra utiliser une formule pour problème raide. Ce faisant, il faut toutefois avoir conscience que toutes les variables ne seront pas intégrées avec la même précision, le rôle du schéma pour problème raide est seulement de prévenir contre la croissance exponentielle des erreurs de troncature affectant les variables ayant les plus petites échelles de temps. Par ailleurs, l'analyse du comportement à l'infini du schéma ne peut être faite que sur le problème type à une dimension. Pour un système aussi complexe que celui de l'évolution temporelle de la composition chimique, la théorie n'est pas faite. En transposant à plusieurs dimensions un résultat établi pour une seule dimension il faut s'attendre à ce que, dans certains cas il puisse y avoir des difficultés. Dans ces cas le seul remède simple est de diminuer le pas temporel.

7.5.2 Résumé des contraintes

Pour l'intégration numérique du système d'équations différentielles de l'évolution temporelle de la composition chimique, il est nécessaire d'utiliser une formule :

- Conservative, *i.e.* conservant toute combinaison linéaire des abondances $\sum_i \alpha_i X_i^t = cte.$
- Stable ou, tout au moins A-stable, *i.e.* que les erreurs de troncature relatives aux variables dont l'échelle de temps d'évolution est la plus courte n'augmentent pas exponentiellement lorsque le pas temporel est grand par rapport à cette échelle de temps.
- Positive.
- Simple et précise.

La première condition est remplie pour la plupart des schémas classiques : Runge–Kutta ou schémas à pas liés, pour lesquels la fonction d'incrémentacion dépend linéairement des dérivées temporelles estimées en divers points. La condition de A-stabilité est remplie si on utilise un schéma implicite. Il est souhaitable, par ailleurs, d'utiliser le même schéma d'intégration en tous les points du modèle.

Le fait que les limites des zones convectives se déplacent en fonction du temps et la nécessité du mélange convectif élimine malheureusement une partie de l'intérêt des schémas à pas liés qui possèdent des propriétés intéressantes de stabilité pour les problèmes raides *e.g.* schémas BDF. De même, il ne sera possible d'utiliser des schémas de Runge–Kutta d'ordre supérieur à 1, que si on ne tient pas compte du mélange convectif pour les modèles intermédiaires comme avec la formule de Wagner (1969).

Lorsque la diffusion microscopique est omise, CESAM2*k* utilise un schéma d'intégration pour problème raide de type Runge–Kutta implicite de type LobattoIIIC.

7.5.3 Les formules IRK Lobatto IIIC.

Dans une zone radiative, sans diffusion, le système des équations de l'évolution de la composition chimique s'écrit formellement :

$$\frac{\partial x_i}{\partial t} = \Psi(T, \rho, \mathcal{X}), 1 \leq i \leq n_{elem}. \quad (7.16)$$

Dans ce qui suit, on décrit l'utilisation de la formule IRK LobattoIIIC, pour alléger le formalisme, on ne considère qu'une seule espèce chimique. On notera x_0 (*respt.* x) l'abondance de cet élément au temps t (*respt.* $t + dt$). Avec les coefficients de la table Table 7.1 (Page 105) le schéma IRK LobattoIIIC est :

$$\begin{aligned} x_i &= x_0 + dt \sum_{j=1}^s a_{ij} \Psi(T_j, \rho_j, x_j), \quad i = 1, \dots, s \\ x &= x_0 + dt \sum_{j=1}^s b_j \Psi(T_j, \rho_j, x_j) \end{aligned} \quad (7.17)$$

où T_j , ρ_j et x_j sont, respectivement, la température, la densité, l'abondance aux temps intermédiaires, $t + c_j dt$, $j = 1, \dots, s$; s est le nombre d'étapes de la formule IRK LobattoIIIC.

Toutefois, cette formulation élémentaire d'un schéma de Runge-Kutta ne donne pas des résultats satisfaisants pour le problème de l'évolution des espèces chimiques, des valeurs notablement négatives des abondances pouvant survenir; la cause est la multiplication des erreurs numériques par les constantes de Lipschitz qui ont des valeurs élevées *e.g.* 10^{18} . Cette difficulté est surmontée en écrivant la première équation Eq. 7.15 (Page 104) pour les écarts $z_i = x_i - x_0$ aux quantités initiales *cf.* Hairer & Wanner (1991, par. IV.8) :

$$\begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_s \end{pmatrix} = dt \mathcal{A} \begin{pmatrix} \Psi(T_1, \rho_1, x_0 + z_1) \\ \Psi(T_2, \rho_2, x_0 + z_2) \\ \vdots \\ \Psi(T_s, \rho_s, x_0 + z_s) \end{pmatrix}. \quad (7.18)$$

La matrice $\mathcal{A} \equiv (a_{i,j})_{i,j=1}^s$ d'une quelconque formule IRK Lobatto IIIC étant inversible, en notant $(d_1, d_2, \dots, d_s) \equiv (b_1, b_2, \dots, b_s) \mathcal{A}^{-1}$, la seconde équation Eq. 7.15 (Page 104) devient simplement¹¹ :

$$x = x_0 + z_s. \quad (7.19)$$

7.5.4 Mélange des éléments chimiques sans diffusion

Dans une zone convective, lorsque l'échelle de temps de la convection est petite par rapport à celle des réactions nucléaires, la composition chimique est homogénéisée par les mouvements de convection. Pour l'isotope i , le taux de destruction Ψ_i n'est plus la quantité locale $\Psi_i(T, \rho, \mathcal{X})$ mais une quantité moyennée sur

11. Cette particularité résulte du fait que la dernière ligne de A est identique au vecteur b *e.g.* toutes les formules IRK Lobatto IIIC sont construites avec cette propriété.

TABLE 7.1 – Coefficients de Eq. 7.15 (Page 104) pour les formules IRK Lobatto IIIC d'ordre $p = 1, 2, 4$ avec $s = 1, 2, 3$ étapes. On utilise la notation usuelle, les c_i , $i = 1, \dots, s$ sont éléments de la première colonne, les b_i ceux de la dernière ligne, les autres coefficients sont les a_{ij} de la matrice caractérisant la formule de Runge-Kutta.

1	1	0	$\frac{1}{2}$	$-\frac{1}{2}$	0	$\frac{1}{6}$	$-\frac{1}{3}$	$\frac{1}{6}$
1	1	1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{5}{12}$	$-\frac{1}{12}$
1	1		$\frac{1}{2}$	$\frac{1}{2}$	1	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$
			$\frac{1}{2}$	$\frac{1}{2}$		$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

toute la zone convective. Ainsi dans une zone convective Eq. 7.14 (Page 104) doit être remplacée par :

$$\frac{d\bar{x}_i}{dt} = \frac{\int_{ZC} \Psi_i(\rho, T, \bar{\mathcal{X}}, t) dm}{\int_{ZC} dm} = \frac{\int_{ZC} \Psi_i(\rho, T, \bar{\mathcal{X}}, t) \sqrt{\nu} d\nu}{\int_{ZC} \sqrt{\nu} d\nu}, \quad \nu = \left(\frac{m}{M_\odot} \right)^{\frac{2}{3}} \quad (7.20)$$

on remarquera que le taux de destruction local Ψ_i est calculé avec $\bar{\mathcal{X}}$ *i.e.* la valeur moyenne des abondances sur la zone mélangée, ce qui évidemment complique la situation.

Comme il est rappelé plus haut, on fait l'hypothèse selon laquelle, dans une zone convective, l'abondance de chaque isotope est constante. Pour reprendre l'exemple d'un seul élément, notons \bar{x} sa valeur dans une zone convective, et à chaque étape temporelle $t + c_j dt$, $j = 1, \dots, s$ de l'intégration Runge-Kutta, notons sa valeur moyenne \bar{x}_j . On estime par la formule du point milieu les intégrales de Eq. 7.18 (Page 105) si bien que :

$$\bar{x}_j = \sum_{k \in \text{MZ}} x_{j,k+1/2} \Delta\nu_{k+1/2}, \quad (7.21)$$

où la somme porte sur toutes les couches que comporte la zone convective, $x_{0,k+1/2}$ est l'abondance au point milieu, $\nu_{k+1/2} \equiv (\nu_{k+1} + \nu_k)/2$ et :

$$\Delta\nu_{k+1/2} \equiv \frac{(\nu_{k+1} - \nu_k) \sqrt{\nu_{k+1/2}}}{\sum_{j \in \text{MZ}} (\nu_{j+1} - \nu_j) \sqrt{\nu_{j+1/2}}}. \quad (7.22)$$

L'équation 7.16 devient :

$$\begin{pmatrix} \bar{z}_1 \\ \bar{z}_2 \\ \vdots \\ \bar{z}_s \end{pmatrix} = dt \mathcal{A} \begin{pmatrix} \sum_k \Psi(T_{1,k+\frac{1}{2}}, \rho_{1,k+\frac{1}{2}}, \bar{x}_0 + \bar{z}_1) \Delta\nu_{k+\frac{1}{2}} \\ \sum_k \Psi(T_{2,k+\frac{1}{2}}, \rho_{2,k+\frac{1}{2}}, \bar{x}_0 + \bar{z}_2) \Delta\nu_{k+\frac{1}{2}} \\ \vdots \\ \sum_k \Psi(T_{s,k+\frac{1}{2}}, \rho_{s,k+\frac{1}{2}}, \bar{x}_0 + \bar{z}_s) \Delta\nu_{k+\frac{1}{2}} \end{pmatrix} \quad (7.23)$$

où $T_{j,k+\frac{1}{2}}$ (*respt.* $\rho_{j,k+\frac{1}{2}}$) est la température (*respt.* la densité) au temps intermédiaire $t + c_j dt$; \bar{x} s'écrit :

$$\bar{x} = \bar{x}_0 + \bar{z}_s. \quad (7.24)$$

Formellement, cette formulation est aussi valable en un point d'une zone radiative, les sommes ne portant que sur une seule valeur de l'indice k . Pour simplifier, la valeur de \bar{x}_0 est obtenue en faisant porter la sommation Eq. 7.19 (Page 105) sur l'intervalle de masse correspondant à la zone convective au temps $t + dt$ qui peut ne pas *exactement* coïncider avec l'intervalle de masse au temps t . L'imprécision engendrée est faible si la zone convective se déplace peu, par contre, dans le cas d'une apparition ou d'une disparition l'erreur engendrée peut être conséquente. Tenir compte de ces subtilités entraînerait de grandes complications dans les algorithmes, car il faudrait introduire explicitement le déplacement de la zone convective dans la relation Eq. ?? (Page ??).

L'extension à plusieurs isotopes ne présente que des difficultés algorithmiques assez techniques. Avec l'approximation des intégrales Eq. 7.19 (Page 105) sur la zone convective à l'aide de la formule du point milieu, la précision numérique n'est que du second ordre; une formule intégrale de Gauss à deux points permettrait d'atteindre le troisième ordre¹². Dans le cas solaire, l'erreur reste limitée, car les réactions thermonucléaires sont peu efficaces dans la zone convective; pour le cas stellaire, les imprécisions de la théorie et des données physiques surpassent largement celles engendrées par un ordre d'intégration peu élevé.

Dans la routine `rkimps` cf. § 8.120 (Page 227), les équations Eq. 7.16 (Page 104) à Eq. ?? (Page ??) sont résolues itérativement à l'aide du schéma de Newton-Raphson¹³ avec un jacobien recalculé pour chaque itération; la méthode implicite est initialisée avec des valeurs initiales nulles pour les z_i ; la convergence est supposée atteinte dès que les corrections sur les z_j sont, en valeurs relatives inférieures à 10^{-8} ; la plupart du temps, l'algorithme converge en moins de 5 itérations.

La routine `rkimps` est appelée par `evol`, cf. § 8.54 (Page 195), qui gère l'évolution temporelle de la composition chimique.

7.5.5 Conservation du nombre de nucléons

Pour chaque isotope d'indice i , les équations d'évolution de la composition chimique sont de la forme générale :

$$\frac{dX_i}{dt} = \dot{X}_i = \sum_{j=1}^m a_j^i X_j + \sum_{\substack{j=1 \\ k \geq j}}^m a_{jk}^i X_j X_k + \sum_{\substack{j=1 \\ l \geq k \geq j}}^m a_{jkl}^i X_j X_k X_l$$

Les abondances X_i des m différentes espèces chimiques satisfont des relations linéaires de conservation (nombre de nucléons, charges) de la forme :

$$\sum_{i=1}^m \gamma_i X_i = cte \Rightarrow \frac{\partial}{\partial t} \sum_{i=1}^m \gamma_i X_i = 0 = \sum_{i=1}^m \gamma_i \dot{X}_i.$$

(Cette dernière relation constitue d'ailleurs un test analytique qui permet de vérifier qu'il n'y a pas d'erreur dans le formalisme des équations d'évolution.)

12. Ce qui n'entraînerait qu'une complication supplémentaire d'algorithme.

13. A cause de la raideur, les constantes de Lipschitz sont grandes et un schéma du type prédicteur-correcteur est inefficace.

Un algorithme linéaire d'intégration permet de conserver parfaitement de telles relations. Il est de la forme générale *cf.* Henrici (1962) :

$$y_n = \sum_{i=1}^k \alpha_i y_{n-i} + h \sum_{i=0}^k \beta_i y'_{n-i}, \quad \sum_{i=1}^k \alpha_i = 1, \quad n \geq k$$

où α_i et β_i , $i = 1, \dots, k$ sont des constantes, h le pas d'intégration, y_n le vecteur des abondances des espèces chimiques à l'instant $t_0 + hn$, t_0 l'instant initial et $k + 1$ le nombre de points sur lesquels s'appuie l'algorithme $1 \leq k \sim 7$.

Avec un algorithme linéaire, il y a conservation parfaite d'une combinaison linéaire des X_j puisque :

$$\begin{aligned} \sum_{j=1}^m \gamma_j X_{j,n} &= \sum_{j=1}^m \gamma_j \sum_{i=1}^k \alpha_i X_{j,n-i} + \sum_{j=1}^m \gamma_j h \sum_{i=0}^k \beta_i \dot{X}_{j,n-i} = \\ &= \sum_{i=1}^k \alpha_i \sum_{j=1}^m \gamma_j X_{j,n-i} + \sum_{i=0}^k \beta_i h \sum_{j=1}^m \gamma_j \dot{X}_{j,n-i} = \\ &= cte + 0. \end{aligned}$$

7.5.6 Conservation des baryons et de la charge

A l'instant t , pour un nombre d'isotopes $n_{\text{elem}} > 1$, les expressions de la conservation du nombre de baryons et de la charge, ont la forme générale d'une relation linéaire entre les abondances, avec des coefficients α_i **constants** :

$$\sum_{i=1}^{n_{\text{elem}}} \alpha_i x_i^t = cte,$$

alors :

$$\frac{\partial}{\partial t} \sum_{i=1}^{n_{\text{elem}}} \alpha_i x_i = 0 \implies \sum_{i=1}^{n_{\text{elem}}} \alpha_i \Psi_i = 0.$$

en tenant compte de la linéarité de Eq. 7.16 (Page 104) et Eq. ?? (Page ??), avec des notations évidentes :

$$\sum_{i=1}^{n_{\text{elem}}} \alpha_i z_i = 0,$$

alors :

$$\sum_{i=1}^{n_{\text{elem}}} \alpha_i x_i^{t+dt} = \sum_{i=1}^{n_{\text{elem}}} \alpha_i x_i^t + \sum_{i=1}^{n_{\text{elem}}} \alpha_i z_i = cte;$$

en particulier, avec le schémas IRK LobattoIIIC, le nombre de baryons et de charges est conservé au niveau :

- de la fermeture du schéma de Newton-Raphson,
- de l'ordre du schéma d'intégration utilisé pour le mélange convectif,
- des erreurs d'arrondi.

Des tests numériques ont montré, dans le cas solaire, que sur un pas temporel, le nombre de baryons était conservé à mieux que 10^{-13} en valeur relative *cf.* Morel (1997) .

7.5.7 Normalisation de la somme des abondances

Bien que les algorithmes utilisés pour la résolution numérique des équations d'évolution temporelle des espèces chimiques assurent la plupart des relations de conservation, ils ne peuvent le faire qu'à la précision requise. Pour limiter les conséquences de ces approximations, une normalisation de la somme des abondances par unité de masse est effectuée de façon à assurer la relation $X + Y + Z = 1$. La normalisation est effectuée dans la routine `evol`, cf. § 8.54 (Page 195) à l'issue de l'intégration temporelle de la composition chimique.

L'abondance par masse de l'élément chimique s'écrit $X_i = x_i \mu_i$ où x_i est l'abondance par mole et μ_i est la masse atomique de l'élément. On a donc : $X + Y + Z = \sum_i x_i \mu_i$. Avec le développement sur la base de B-splines d'ordre m , au point d'abscisse ν , $\nu_{l-1} \leq \nu < \nu_l$, l'abondance par mole s'écrit :

$$x_i = \sum_{j=l-m+1}^l x_{ij} N_j^m(\nu).$$

et de là :

$$\sum_i X_i = \sum_i \mu_i \sum_{j=l-m+1}^l x_{ij} N_j^m(\nu) = \sum_{j=l-m+1}^l N_j^m(\nu) \sum_i \mu_i x_{ij}. \quad (7.25)$$

On obtient une normalisation des abondances par unité de masse X_i en posant :

$$X_i = x'_i \mu_i, \quad x'_i = \sum_{j=l-m+1}^l x'_{ij} N_j^m(\nu), \quad x'_{ij} = \frac{x_{ij}}{\eta_j}, \quad \eta_j = \sum_i \mu_i x_{ij}.$$

La somme des B-splines en tout point ν étant égale à l'unité, on obtient la normalisation désirée :

$$\begin{aligned} X + Y + Z &= \sum_{j=l-m+1}^l N_j^m(\nu) \sum_i \mu_i x'_{ij} = \sum_{j=l-m+1}^l N_j^m(\nu) \sum_i \mu_i \frac{x_{ij}}{\eta_j} = \\ &= \sum_{j=l-m+1}^l N_j^m(\nu) \frac{1}{\eta_j} \sum_i \mu_i x_{ij} = \sum_{j=l-m+1}^l N_j^m(\nu) = 1. \end{aligned}$$

7.5.8 Estimation de la précision de l'intégration

Malgré un grand nombre d'essais, il n'a pas été possible d'utiliser valablement une méthode numérique d'estimation de la précision de l'intégration, la raison étant principalement un coût excessif en calcul de ces méthodes ; dans `CESAM2k` la précision est assurée par la limitation de la variation temporelle relative des abondances des éléments que l'utilisateur désire contrôler.

7.6 Evolution du moment cinétique sans diffusion

`CESAM2k` ne considère qu'une dimension radiale ; pour les modèles en rotation, on tient compte de la valeur moyenne de l'accélération centrifuge sur la calotte sphérique de rayon r .

Compte tenu de l'hypothèse de sphéricité, le moment d'inertie $d\mathcal{I}$ du volume sphérique élémentaire $dV = r \sin \theta dr d\theta d\phi$ au point de coordonnées sphériques (r, θ, ϕ) est :

$$d\mathcal{I} = r \sin \theta \rho r dr d\theta d\phi.$$

Le moment d'inertie par unité de masse de la calotte sphérique s'écrit :

$$\frac{dI}{dm} = \frac{\int_0^{2\pi} d\mathcal{I} d\phi}{dm} = \frac{2\pi r^4 \rho dr}{4\pi r^2 \rho dr} \int_0^\pi \sin^2 \theta d\theta = \frac{2}{3} r^2.$$

Le moment cinétique spécifique *i.e.* le moment cinétique par unité de masse est alors :

$$\mathcal{M}_\Omega(t) = \frac{2}{3} r^2 \Omega(t).$$

$\Omega(t)$ est la vitesse angulaire moyenne de la calotte sphérique au temps t .

L'hypothèse de **conservation locale du moment cinétique spécifique** s'écrit :

$$\frac{D\mathcal{M}_\Omega}{Dt} = \frac{Dr^2\Omega}{Dt} = 0.$$

Avec l'hypothèse de rotation rigide, le moment cinétique total change au cours de l'évolution, en raison des variations locales de densité et de rayon. On peut imaginer qu'un processus physique, non décrit, permet de conserver globalement le moment cinétique, auquel cas la vitesse angulaire *rigide* varie au fur et à mesure de l'évolution¹⁴. Compte tenu de l'hypothèse de sphéricité, le moment cinétique total au temps t est :

$$\mathcal{M}_\Omega(t) = \frac{2}{3} \int_0^{M_\star(t)} r(t, m)^2 \Omega(t, m) dm,$$

où $\Omega(t, m)$ est la vitesse angulaire locale au temps t pour l'abscisse lagrangienne m . Avec l'hypothèse de **rotation rigide** *i.e.* $\Omega(t, m) \equiv \Omega(t)$ et conservation globale du moment cinétique *i.e.* $\mathcal{M}_\Omega(t) \equiv \mathcal{M}_\Omega = cte.$, la vitesse angulaire $\Omega(t + dt)$ à l'instant $t + dt$ vérifie :

$$\mathcal{M}_\Omega(t + dt) = \frac{2}{3} \Omega(t + dt) \int_0^{M_\star(t+dt)} r(t + dt, m)^2 dm,$$

on a donc :

$$\mathcal{M}_\Omega(t + dt) = \int_0^{M_\star(t)} r(t, m)^2 \Omega(t, m) dm \quad / \quad \int_0^{M_\star(t+dt)} r(t + dt, m)^2 dm.$$

Avec l'hypothèse de **conservation locale du moment cinétique**, on suppose que les zones convectives, avec leurs extensions par overshoot, sont en rotation solide. Dans ces zones la vitesse angulaire est donnée par des relations analogues à la précédente dans lesquelles les bornes d'intégration correspondent aux limites des zones convectives. Dans les parties radiatives la relation de conservation locale :

$$0 = \frac{Dr^2\Omega}{Dt} = 2r\Omega \frac{Dr}{Dt} + r^2 \frac{D\Omega}{Dt},$$

14. La possibilité de la conservation globale du moment cinétique a été implantée dans CESAM2k à l'initiative de MJ.Goupil.

est discrétisée :

$$0 = 2r\Omega \frac{r - r^{(t)}}{\Delta t} + r^2 \frac{\Omega - \Omega^{(t)}}{\Delta t} \implies \Omega = \frac{r\Omega^{(t)}}{3r - 2r^{(t)}}.$$

En l'absence d'un noyau convectif, cette relation est indéterminée au centre. On suppose que la vitesse angulaire en $R = 0$ est égale à celle déterminée au point de grille le plus proche.

L'énergie cinétique spécifique de rotation est :

$$\epsilon_\Omega = \frac{1}{2} \frac{dI}{dm} \Omega^2 = \frac{1}{3} r^2 \Omega^2.$$

Au cours de l'évolution, la vitesse angulaire locale varie en raison de la perte de moment cinétique, ou de la conservation locale ou globale du moment cinétique ; ces variations conduisent à celles de la vitesse aérolaire qui ont pour conséquences des variations de l'énergie cinétique spécifique locale ϵ_Ω :

$$\frac{\partial \epsilon_\Omega}{\partial t} = \frac{1}{3} \frac{\partial (r\Omega)^2}{\partial t} \simeq \frac{1}{3} \frac{r^2(t + \Delta t)\Omega^2(t + \Delta t) - r(t)^2\Omega(t)^2}{\Delta t}$$

On suppose que ces variations d'énergie sont, selon leur signe, prises ou restituées au milieu sous la forme de sources de chaleur. L'équation d'énergie, Eq. 7.2 (Page 83), estimée dans la routine `static_m` ou `static_r` cf. §8.126 (Page 232), devient alors :

$$\frac{\partial L}{\partial M} = \epsilon - \frac{\partial U}{\partial t} + \frac{P}{\rho^2} \frac{\partial \rho}{\partial t} - \frac{\partial \epsilon_\Omega}{\partial t}.$$

Avec une rotation solide de vitesse angulaire non nulle, le moment cinétique est réestimé pour chaque modèle intermédiaire dans la routine `resout`, cf. §8.116 (Page 223).

7.7 Evolution temporelle avec diffusion

7.7.1 Intégration par éléments finis

En raison du mélange convectif, les équations de la diffusion des éléments chimiques et du moment cinétique forment un problème intégral-différentiel aux limites du quatrième ordre avec conditions initiales. Bien que ces équations soient couplées, CESAM2k les résout séparément en alternance (méthode des lignes) :

1. La diffusion du moment cinétique, problème du quatrième ordre.
2. La diffusion des éléments chimiques, problème différentiel parabolique.

CESAM2k utilise la méthode des éléments finis pour leur intégration numérique. Formellement ces équations sont de la forme :

$$a \frac{Dy}{Dt} + bG = \frac{\partial F}{\partial x} \quad (7.26)$$

$F(x, y)$, $a(x, y)$ et $b(x, y)$ étant des fonctions de l'abscisse x et de la fonction inconnue $y(x)$.

Comme il a été évoqué précédemment, les fonctions inconnues sont projetées sur une base fonctionnelle de dimension finie. Les projections sont obtenues en écrivant que les développements satisfont les équations différentielles et les conditions limites. Pour ces fonctions de base, CESAM2k utilise des polynômes par morceaux d'ordre m (polynômes de degré $m - 1$), se raccordant avec une oscurarité de classe \mathcal{C}_i , $0 \leq i \leq m - 1$, (continuité des dérivées d'ordre 0 à $m - 1$). Ces polynômes par morceaux sont exprimés sous la forme de B-splines polynômiales normalisées d'ordre m et de dimension $d : \{S_j^m\}_{j=1}^d$. Ainsi chacune des fonctions est approchée par une fonction d'interpolation de la forme :

$$f(x) \simeq \sum_{j=1}^d f_j S_j^m(x) \text{ donc } : \frac{df}{dx} \simeq \sum_{j=1}^d f_j \frac{dS_j^m}{dx}.$$

la connaissance des f_j détermine la fonction d'interpolation de f avec une précision que l'on sait estimer *cf.* Schumaker (1981). Des algorithmes stables et précis permettent de calculer les $S_j^m(x)$ ainsi que toutes leurs dérivées, objet de la routine **bvald**, *cf.* § 9.2.7 (Page 244), de CESAM2k.

Le principe de la résolution des équations de diffusion consiste à imposer à la base d'être orthogonale à son image par l'opérateur différentiel, *i.e.* orthogonale au résidu. C'est la "méthode de Galerkin". Son application aux cas de la diffusion des espèces chimiques et du moment cinétique est "semi-spectrale", les fonctions de base ne vérifiant pas les conditions limites. Les solutions numériques obtenues sont dites *faibles*. En formant les produits scalaires :

$$\langle f \bullet g \rangle = \int_{x_0}^{x_1} fg dx \quad (7.27)$$

on obtient :

$$\langle (a \frac{Dy}{Dt} + bG) \bullet S_j^m \rangle = \langle \frac{\partial F}{\partial x} \bullet S_j^m \rangle$$

En intégrant par parties le second produit scalaire on obtient :

$$\langle \frac{\partial F}{\partial x} \bullet S_j^m \rangle = [F S_j^m]_{x_0}^{x_1} - \langle F \bullet \frac{\partial S_j^m}{\partial x} \rangle$$

On a :

$$[F]_{x_0}^{x_1} \equiv [F S_j^m]_{x_0}^{x_1} = \begin{cases} -F(x_0) & \text{si } j = 1 \\ F(x_1) & \text{si } j = d \\ 0 & \text{sinon} \end{cases}$$

puisque aux extrémités de l'intervalle d'intégration, une seule des B-splines est non nulle et égale à l'unité. *In fine* on obtient la discrétisation suivante :

$$\langle \left(a \frac{y^{t+1} - y^t}{\Delta t} + bG \right) \bullet S_j^m \rangle + \langle F \bullet \frac{\partial S_j^m}{\partial x} \rangle - [F]_{x_0}^{x_1} = 0 \quad (7.28)$$

On est ramené - éventuellement après linéarisation - à la résolution d'un système linéaire ; les B-splines étant à support borné, le système linéaire est bande-diagonal. Il est résolu par élimination de Gauss (pivots partiels) par la routine **gauss_dp**, *cf.*

§ 9.1 (Page 241). Les produits scalaires sont approchés par une intégration numérique de type Gauss, suivant un algorithme inspiré de l'algorithme 5.22, Schumaker (1981, p. 203).

La nature "intégrale" des équations résulte du mélange convectif. Il est effectué dans CESAM2k par le biais d'une diffusion turbulente de coefficient $D_M \gg 1$.

Ces dispositions ont les avantages suivants :

- L'ordre des équations différentielles à résoudre est réduit d'une unité.
- Les discontinuités des dérivées premières sont implicitement prises en compte par la formulation intégrale, Eq. 7.34 (Page 114), des produits scalaires.
- Avec l'utilisation de la base des B-splines les algorithmes restent simples même lorsqu'il se produit des déplacements des limites zone radiative / zone convective.

7.7.2 Bases de B-splines pour Petrov-Galerkin

Une représentation schématique du vecteur nodal pour¹⁵ $m = 4$, avec discontinuité de la **dérivée première** au point fléché \Downarrow , cf. Schumaker (1981), est la suivante :

$$\begin{array}{cccccccc}
 & & & & & \Downarrow & & & \\
 \times & \times & \times & \dots & \times & \times & \times & \dots & \times & \times \\
 & \times & & & & \times & & & & \times \\
 & \times & & & & \times & & & & \times \\
 & \times & & & & & & & & \times
 \end{array}$$

Avec continuité de la fonction et des deux premières dérivées, en tous les points de grille :

$$\begin{array}{cccccccc}
 \times & \times & \times & \dots & \times & \times & \times & \dots & \times & \times \\
 \times & & & & & & & & & \times \\
 \times & & & & & & & & & \times \\
 \times & & & & & & & & & \times
 \end{array}$$

7.7.3 Présence de discontinuités

Dans le cas où la fonction à intégrer présente des discontinuités, la base de B-splines sur laquelle elle est projetée, doit inclure ces discontinuités. Avec une discontinuité au point x_d de $]x_0, x_1[$ l'intégrale du produit scalaire, cf. équation 7.23 (Page 111), est définie par :

$$\langle f \bullet g \rangle = \int_{x_0}^{x_{d^-}} fg dx + \int_{x_{d^-}}^{x_1} fg dx, \quad (7.29)$$

Pour le problème différentiel, cf. équation 7.22 (Page 110), avec le produit scalaire ainsi défini, l'intégration par parties est justifiée, car elle ne porte que sur des fonctions continues. *In fine* pour les B-splines dont le support est contenu dans l'intervalle $[x_0, x_{d^-}[$ on obtient :

$$\left\langle \left(a \frac{y^{t+1} - y^t}{\Delta t} + bG \right) \bullet S_j^m \right\rangle + \left\langle F \bullet \frac{\partial S_j^m}{\partial x} \right\rangle - [F]_{x_0}^{x_{d^-}} = 0, \quad (7.30)$$

15. Polynômes par morceaux de degré 3, se raccordant par la continuité de leurs dérivées secondes.

et pour celles dont le support est contenu dans l'intervalle $[x_1, x_{d+}[$:

$$\left\langle \left(a \frac{y^{t+1} - y^t}{\Delta t} + bG \right) \bullet S_j^m \right\rangle + \left\langle F \bullet \frac{\partial S_j^m}{\partial x} \right\rangle - [F]_{x_{d+}}^{x_1} = 0. \quad (7.31)$$

Les produits scalaires étant définis par l'équation 7.25 (Page 112).

En un point de discontinuité le vecteur nodal présentera m abscisses identiques. Pour un ordre de B-spline $m = 4$, une représentation schématique du vecteur nodal avec discontinuité au point fléché est :

$$\begin{array}{cccccccc} & & & & & \Downarrow & & & \\ \times & \times & \times & \dots & \times & \times & \times & \dots & \times & \times \\ \times & & & & & \times & & & & \times \\ \times & & & & & \times & & & & \times \\ \times & & & & & \times & & & & \times \end{array}$$

Cette disposition est équivalente à autant de domaines d'intégration distincts qu'il y a de discontinuités. Cette représentation "regroupée" n'est utile que si les discontinuités apparaissent, disparaissent ou encore se déplacent. Certains systèmes différentiels peuvent présenter simultanément, en divers points du domaine d'intégration des variables continues, des variables continues non dérivables et des variables discontinues. Un exemple est celui du système des équations de diffusion du moment cinétique, cf. § 7.9 (Page 129). Une représentation schématique d'un vecteur nodal avec une discontinuité de la dérivée première (\downarrow) et une discontinuité (\Downarrow) est :

$$\begin{array}{cccccccc} & & \downarrow & & & \Downarrow & & & \\ \times & \times & \times & \times & \dots & \otimes & \times & \times & \dots & \times & \times \\ \times & & \times & & & & \times & & & & \times \\ \times & & \times & & & & \times & & & & \times \\ \times & & & & & & \otimes & & & & \times \end{array}$$

Quand le problème différentiel présente, en des abscisses identiques, des variables continues et des variables discontinues¹⁶, CESAM2k assure la continuité en égalant les projections des variables concernées sur chacune des deux B-splines localisées de part et d'autre la discontinuité. Dans la représentation schématique précédente, ces B-splines sont identifiées avec le symbole \otimes . Pour obtenir autant d'équations que d'inconnues la relation résultant du produit scalaire avec une des B-splines concerné est supprimé. Suppression qui élimine la contribution de cette B-spline à la quantité intégrée, cf. § 7.7.1 (Page 110).

7.8 Diffusion des éléments chimiques

Sous sa forme conservative, l'équation d'évolution temporelle des espèces chimiques avec diffusion :

$$\frac{Dx_i}{Dt} = -\frac{\partial F_i}{\partial m} + \Psi_i, \quad 1 \leq i \leq n_{\text{elem}} \quad (7.32)$$

16. Tel sera le cas pour le système des équations de diffusion du moment cinétique.

contient une partie diffusive, $\frac{\partial F_i}{\partial m}$, et une partie nucléaire, Ψ_i . Cette dernière est calculée par la routine de type nuc utilisée, cf. § 8.77 (Page 206). Le flux de particules d'abondance en nombre x_i a pour expression (Proffit & Michaud, 1991) :

$$F_i = -4\pi R^2 \rho \left(4\pi R^2 \rho D_i * \nabla_m \mathcal{X} + v_i x_i \right), \quad (7.33)$$

les composantes $d_{i,j}$, $i \neq j$, du vecteur :

$$D_i(m, t) \equiv (d_{i,1}, \dots, d_{i,n_{\text{elem}}})^T,$$

sont les coefficients, $d_{i,j} = d_{i,j}^*$, de diffusion microscopique de l'élément d'indice i , par rapport à l'élément d'indice j ; la (i, i) -ième composante :

$$d_{i,i} \equiv d_{i,i}^* + d_T + d_M \quad (7.34)$$

inclut les coefficients de diffusion turbulente d_T et de mélange d_M ; $v_i(m, t)$ est la vitesse de sédimentation; on a noté à l'aide du symbole "*" le produit scalaire :

$$D_i * \nabla_m \mathcal{X} \equiv \sum_j d_{i,j} \frac{\partial x_j}{\partial m}.$$

En utilisant la variable spatiale $\nu \equiv (m/M_\odot)^{2/3}$, l'équation 7.28 (Page 113) devient :

$$\frac{\partial x_i}{\partial t} = -\frac{\partial F_i}{\partial m} + \Psi_i = -\frac{2}{3M_\odot \sqrt{\nu}} \frac{\partial F_i}{\partial \nu} + \Psi_i, \quad 1 \leq i \leq n_{\text{elem}}. \quad (7.35)$$

Des zones mélangées, *i.e.* zones convectives éventuellement overshootées, apparaissent, se déplacent, disparaissent au cours de l'évolution. On admet que le mélange convectif est suffisamment efficace pour uniformiser la composition chimique, ainsi :

$$\frac{\partial x_i}{\partial R} = 0. \quad (7.36)$$

Ces conditions entraînent la discontinuité des gradients des abondances aux limites entre zones radiatives et zones mélangées; d'où la nécessité d'introduire des limites internes mobiles et en nombre variable, ce qui complique l'intégration numérique. Grâce à un choix *ad hoc* des bases de B-splines utilisées pour la diffusion des éléments chimiques, les discontinuités des gradients sont implicitement prises en compte.

La présence du mélange convectif change la nature du problème, de différentiel il devient **intégro-différentiel**. L'équation de transport des éléments chimiques étant du type advection-diffusion, on contourne l'aspect intégral en uniformisant la composition chimique par une diffusion. Pour ce faire, on impose dans les zones mélangées un coefficient de diffusion dont l'ordre de grandeur, grand devant l'unité, correspond au temps de retournement des éléments convectifs. CE-SAM2k utilise $d_{\text{conv}} = 10^{13} \text{ cm}^2 \text{ s}^{-1}$. A chaque limite entre une zone mélangée et la zone non mélangée adjacente les coefficients de diffusion sont discontinus avec pour conséquence la discontinuité des gradients des abondances des espèces chimiques.

On représente l'abondance de chaque espèce par un polynôme par morceaux d'ordre $m_c \geq 2$ **avec discontinuité de la dérivée première** à chaque limite zone radiative / zone convective, cf. § 7.2.1 (Page 83). Pour le calcul numérique, on projette

ces polynômes sur une base de B-splines avec discontinuité de la dérivée première : $\mathcal{N} \equiv \{N_k^{m_c}\}_{k=1, K_X}$, $\mathcal{N} \subset \mathcal{C}^0[0, \nu_b]$; K_X étant la dimension de la base. On a alors :

$$x_i(\nu, t) = \sum_{k=1}^{K_X} x_{i,k}(t) N_j^{m_c}(\nu). \quad (7.37)$$

On recherche les *solutions faibles* de l'équation 7.31 (Page 114) à l'aide du formalisme de Galerkin. Les produits scalaires s'écrivent :

$$\left\langle \frac{3}{2} M_\odot \sqrt{\nu} \left(\frac{\partial x_i}{\partial t} - \Psi_i \right) \bullet S_j^{m_c} \right\rangle + \left\langle \frac{\partial F_i}{\partial \nu} \bullet S_j^{m_c} \right\rangle = 0, \quad (7.38)$$

En tenant compte de la continuité des flux F_i , l'intégration par parties conduit à :

$$\left\langle \frac{\partial F_i}{\partial \nu} \bullet S_j^{m_c} \right\rangle = \left[F_i S_j^{m_c} \right]_0^{\nu_b} - \left\langle F_i \bullet \frac{\partial S_j^{m_c}}{\partial \nu} \right\rangle, \quad (7.39)$$

avec la condition limite, $\nu \equiv 0$, $F_i(\nu = 0) = 0$ au centre et, à la limite externe de l'enveloppe $\nu = \nu_b$. On montre au § 7.8.1 (Page 115) :

$$F_i(\nu = \nu_b) = \dot{M}(x_{iv} - x_i)$$

où x_{iv} est l'abondance par mole de l'espèce i dans le vent stellaire. Par convention, le taux de perte de masse \dot{M} est négatif s'il y a perte, et positif s'il y a gain. Comme à la limite de l'enveloppe, seule la dernière spline $S_{K_X}^{m_c}$, est non nulle et comme $S_{K_X}^{m_c}(\nu_b) \equiv 1$ la quantité intégrée devient :

$$\left[F_i S_j^{m_c} \right]_0^{\nu_b} = \dot{M}(x_{iv} - x_i)(\nu_b).$$

Physiquement, le vent stellaire ne peut concerner que la limite externe, à défaut de le décrire en détail, on suppose qu'il concerne toute la zone convective externe de masse M_{ZC} . Pour simplifier l'algorithme on transpose la partie intégrée dans le terme nucléaire Ψ_i ainsi qu'il est décrit au § 7.8.1 (Page 115).

La continuité de x_i et de F_i permet d'écrire l'équation 7.34 (Page 114) sous la forme utilisée dans CESAM2k :

$$\begin{aligned} & \left\langle \frac{3}{2} \sqrt{\nu} \left(x_i - x_i^{(t)} - \Psi_i \Delta t \right) \bullet S_j^{m_c} \right\rangle - \frac{\dot{M}}{M_{ZC}} (x_{iv} - x_i) \Delta t - \\ & - \left\langle \left[\frac{32\pi^2 R^4 \rho^2 \Delta t}{3M_\odot^2 \sqrt{\nu}} \{ D_i * \nabla_\nu \mathcal{X} \} + \frac{4\pi R^2 \rho}{M_\odot} \Delta t V_i x_i \right] \bullet \frac{\partial S_j^{m_c}}{\partial \nu} \right\rangle = 0, \quad j = 1, \dots, K_X, \end{aligned} \quad (7.40)$$

les quantités étant prises à l'instant $t + \Delta t$ sauf $x_i^{(t)}$ pris au temps t . Ce schéma complètement implicite est d'ordre m_c en espace, et du premier ordre en temps.

Au centre, par symétrie, les dérivées spatiales des abondances sont nulles ainsi que leurs vitesses d'advection. Il en est de même pour la limite externe toujours située dans une zone de mélange. Le terme intégré qui est nul a été omis dans l'écriture des équations.

7.8.1 Condition limite externe

On désignera par "vent", indistinctement un apport ou un retrait de masse. A défaut de décrire le processus de création du vent, on suppose que celui-ci est issu de la zone convective externe qui existe toujours. Pendant l'intervalle de temps Δt , l'apport en masse d'élément chimique i à la zone convective $\dot{M}X_{iv}\Delta t$, X_{iv} étant l'abondance par masse de l'élément i au temps t dans le vent.

A l'issue du pas temporel, la fraction de masse de l'élément i dans la zone convective sera :

$$X_i^{t+\Delta t} = \frac{M_{ZC}X_i^t + \dot{M}X_{iv}\Delta t}{\sum_i M_{ZC}X_i^t + \dot{M}X_{iv}\Delta t},$$

où X_i^t et X_{iv} sont respectivement l'abondance par masse de l'élément i au temps t et dans le vent. Les fractions de masse vérifiant $\sum_i X_i^t \equiv 1$ et $\sum_i X_{iv}^t \equiv 1$:

$$\begin{aligned} X_i^{t+\Delta t} &= \frac{M_{ZC}X_i^t + \dot{M}X_{iv}\Delta t}{M_{ZC} + \dot{M}\Delta t} = \frac{X_i^t + \frac{\dot{M}}{M_{ZC}}X_{iv}\Delta t}{1 + \frac{\dot{M}}{M_{ZC}}\Delta t} \sim \\ &X_i^t \left(1 - \frac{\dot{M}}{M_{ZC}}\Delta t\right) + \frac{\dot{M}}{M_{ZC}}X_{iv}\Delta t \left(1 - \frac{\dot{M}}{M_{ZC}}\Delta t\right) = \\ &X_i^t - \frac{\dot{M}}{M_{ZC}}X_i^t\Delta t + \frac{\dot{M}}{M_{ZC}}X_{iv}\Delta t - \left(\frac{\dot{M}}{M_{ZC}}X_{iv}\Delta t\right)^2 \sim X_i^t + \frac{\dot{M}}{M_{ZC}}(X_{iv} - X_i^t)\Delta t \end{aligned}$$

On obtient pour la dérivée temporelle de l'abondance de l'espèce chimique i :

$$\dot{X}_i = \lim_{\Delta t \rightarrow 0} \frac{X_i^{t+\Delta t} - X_i^t}{\Delta t} = \frac{\dot{M}}{M_{ZC}}(X_{iv} - X_i^t) \iff \dot{x}_i = \frac{\dot{M}}{M_{ZC}}(x_{iv} - x_i^t), \quad (7.41)$$

où x_{iv} est l'abondance par mole de l'élément i dans le vent. Dans la zone convective externe la variation temporelle de l'isotope X_i est donné par :

$$\dot{x}_i M_{ZC} = \dot{M}(x_{iv} - x_i^t). \quad (7.42)$$

Les abondances ne varient que si la composition chimique du vent est différente de celle des couches externes. Dans la zone convective externe les variations de composition chimique résultant de l'équation 7.38 sont appliquées en addition de celles résultant des réactions thermonucléaires.

Ces dispositions sont appliquées même si la diffusion microscopique des éléments chimique est ignorée.

7.8.2 Chutes de planètes

De façon similaire à ce qui précède, au cours d'une évolution CESAM2k permet de simuler une chute de planètes sur un intervalle de temps limité. La composition chimique de la zone convective externe se trouve alors modifiée par des éléments chimiques dont l'abondance et la nature peuvent, éventuellement, différer de celles des conditions initiales et/ou locales. Dans la zone convective externe la variation temporelle de l'isotope X_i est donné par :

$$\dot{x}_i(t) = \frac{N_P M_{\oplus}}{M_{ZC}} \mathcal{P}(t)(x_{iv} - x_i^t). \quad (7.43)$$

$N_P > 0$ est le nombre total de planétoïdes de masse terrestre M_\oplus reçus par l'étoile, $\mathcal{P}(t)$ la fonction décrivant la dépendance temporelle, *cf.* § 4.6 (Page 57).

CESAM2*k* peut tenir compte de l'apport de moment cinétique à la zone convective externe résultant des chutes de planétoïdes, *cf.* § 7.9.10 (Page 147).

7.8.3 Notations

Le formalisme de Burgers (1969), permet de déterminer les coefficients de diffusion microscopique des éléments chimiques. On reformule le système des équations de Burgers pour les variables utilisées par CESAM2*k*, les abondances par mole x_i et le poids moléculaire moyen μ . Le principe de la méthode numérique de Burgers, Iben & MacDonald (1985), consiste à remarquer que la résolution d'un système linéaire permet d'écrire la vitesse de diffusion w_i de la particule i sous la forme :

$$w_i = v_i + \sum_j b_{ij} \frac{dx_j}{dx}.$$

Iben & MacDonald (1985) n'ont pas tenu compte des "residual heat flow vector" qui furent introduits par la suite par Cox et al. (1989) et repris par Thoul et al. (1994).

On utilise les notations suivantes :

- c : nombre d'ions, il y a $c + 1$ espèces de particules, *i.e.* ions+ électrons,
- E : champ électrique,
- e : charge de l'électron, en unité électrostatique, et indice de la particule "électron",
- g_G : accélération due à la gravité,
- g_Ω : accélération centrifuge,
- g_{Ri} : accélération radiative sur la particule i ,
- g_T : accélération totale,
- $g_{j,i}$: poids statistique du fondamental du niveau d'ionisation j de l'élément chimique i ,
- $m_e = \mu_e m_u$: masse de l'électron,
- $m_i \equiv \mu_i m_u$: masse de l'élément chimique i ,
- $m_{ij} \equiv \frac{m_i m_j}{m_i + m_j}$: masse réduite des particules d'indices i et j ,
- $m_u = 1/N_0$: masse atomique unité, inverse du nombre d'Avogadro,
- n_e : nombre d'électrons par unité de volume,
- n_i : nombre d'ions de l'élément chimique $i \neq e$ par unité de volume,
- $n_{j,i}$: nombre d'ions de l'élément chimique i par unité de volume, dans l'état d'ionisation $j = 0, \dots, Z_i$,
- r : rayon, variable d'espace eulérienne,
- $T, P, P_i, P_C, P_R, \rho$: température, pression totale, pression partielle pour l'élément i , pression collisionnelle, pression de radiation, densité,
- t : temps,
- $\mathcal{V} = \frac{\partial r}{\partial t}$: vitesse d'entraînement du fluide, $\mathcal{V} \sim 0$ (hypothèse de l'équilibre quasi-statique),
- w_i : vitesse de diffusion de la particule i , par rapport à la vitesse moyenne du fluide,
- X_i : proportion en masse de l'abondance de l'élément chimique i , $\sum_i X_i \equiv 1$,

- $x_i \equiv X_i/\mu_i$: proportion par mole de l'abondance de l'élément chimique i ,
- $x_{j,i}$: taux d'ionisation du niveau d'ionisation j de l'ion i ,
- Z_i charge de la particule i , $Z_e = -1$,
- \bar{Z}_i : charge moyenne de l'élément i avec ionisation,
- η : paramètre de dégénérescence,
- μ : poids moléculaire moyen,
- μ_e : masse atomique de l'électron, $\mu_e = m_e/m_u$,
- μ_i : masse atomique de l'élément chimique i ,
- $\chi_{j,i}$: potentiel d'ionisation du niveau d'ionisation j de l'élément i .

7.8.4 Charge moyenne des ions

On utilise l'équation de Saha (Cox & Giuli, 1968, eq. 15-30) dans laquelle on limite les fonctions de partition aux poids statistiques des niveaux fondamentaux :

$$\frac{n_{j-1,i}}{n_{j,i}} = \frac{g_{j-1,i}}{g_{j,i}} \exp\left(\eta - \Delta\nu + \frac{\chi_{j,i}}{kT}\right) \equiv \varphi_{j,i}, \quad j = 1, \dots, Z_i. \quad (7.44)$$

Cette formulation diffère de la forme courante qui exprime le rapport inverse $\frac{n_{j,i}}{n_{j-1,i}}$. Elle permet d'utiliser l'équation 7.41 (Page 119) qui constitue un algorithme de résolution performant (Mihalas, 1978, eq. 5,17-5,20).

Le taux de dégénérescence η vérifie :

$$F_{\frac{1}{2}}(\eta) = \frac{n_e}{4\pi} \left(\frac{h^2}{2m_e kT} \right)^{\frac{3}{2}},$$

$F_{\frac{1}{2}}(\eta)$ est la fonction de Fermi-Dirac (Clayton, 1968, eq. 2-57). $\Delta\nu$ est une correction numérique introduite par Eggleton et al. (1973), pour simuler l'ionisation de pression, et éviter la recombinaison à haute température (Kippenhahn & Weigert, 1991, par. 14.6) :

$$\Delta\nu = n_e c_F \left(\frac{a_0}{\bar{Z}} \right)^3 \left(1 + \frac{20\chi_0}{kT} \right), \quad a_0 = \frac{h^2}{4\pi^2 m_e e^2}, \quad \bar{Z} = X + 2Y + 8Z, \quad c_F = 15, \quad \chi_0 = 13.6.$$

L'expérience des calculs a montré que le formalisme d'Eggleton Eggleton et al. (1973), même avec la modification de Proffit & Michaud (1991, eq. 4), donnait des taux d'ionisation moyens avec des paliers et des gradients beaucoup plus marqués que ceux obtenus avec une meilleure prise en compte de l'ionisation de pression. Ainsi, au centre du Soleil, le formalisme de Eggleton et al. (1973) donne un taux d'ionisation de 100% pour Fe XXV, alors que le calcul plus précis de Gabriel (1997) ne donne que 85%.

Pour tenir compte de l'ionisation de pression et éviter la recombinaison, la routine *saha* de *CESAM2k* utilise une correction numérique dérivée de l'analyse de Clayton (1968, p. 140-145). Le potentiel d'ionisation est réduit lorsque la distance moyenne entre les ions devient de l'ordre de grandeur de la longueur de Debye :

$$R_D = \sqrt{\frac{kT}{4\pi e^2 \rho N_0 \zeta}}, \quad \zeta = \sum_{i \neq e} \bar{Z}_i (\bar{Z}_i + 1) x_i$$

et se réduit¹⁷ à :

$$\chi'_{j,i} = \max(0, \chi_{i,j} - \chi_{i,j}^C), \quad \chi_{i,j}^C = \frac{j e^2}{R_D}.$$

Cette correction appliquée seulement sur le potentiel d'ionisation s'est avérée insuffisante pour éviter la recombinaison à haute température, en particulier, de l'ion III. La routine `saha` utilise un ajustement *numérique* agissant à la fois sur le potentiel d'ionisation et le quotient des fonctions de partition ; celles-ci devant être tronquées de façon cohérente avec la diminution des potentiels d'ionisation, *cf.* Clayton (1968, fig. 2-16). Celà n'est pas possible avec les fonctions de partition limitées aux poids statistiques des niveaux fondamentaux. L'artifice consiste en une diminution progressive de 1 à 0 du rapport des poids statistiques des niveaux fondamentaux $g_{j-1,i}/g_{j,i}$, dès que la quantité :

$$x = \frac{\chi_{i,j}}{\chi'_{j,i}} - 1$$

devient inférieure à une valeur p fixée, pour le niveau d'ionisation $j = 1, \dots, Z_i$ de l'élément chimique i . L'expérience des calculs a montré que la valeur $p = 4$ permettait de retrouver, pour des modèles d'étoiles de la séquence principale de $1M_\odot$ et $1.4M_\odot$ respectivement, des taux d'ionisation voisins de ceux obtenus avec un calcul plus exact, mais beaucoup plus lourd, de la correction de pression d'ionisation. La fonction de raccordement $f(x)$ utilisée est un morceau de cubique sur $[0, p]$, de pente nulle en $x = 0$ et $x = p$:

$$f(x) = 0 \text{ si } x \leq 0, \quad \left(\frac{x}{p}\right)^2 \left[-2\left(\frac{x}{p}\right) + 3\right] \text{ si } x \in [0, p], \quad f(x) = 1, \text{ si } x \geq p.$$

Avec ce formalisme approximatif, l'équation de Saha 7.40 devient :

$$\frac{n_{j-1,i}}{n_{j,i}} = \frac{g_{j-1,i}}{g_{j,i}} f(x) \exp\left(\eta + \frac{\chi_{j,i}}{kT}\right) \equiv \varphi_{j,i}, \quad j = 1, \dots, Z_i. \quad (7.45)$$

On obtient ainsi l'ionisation totale du niveau $j - 1$ lorsque $g_{j-1,i}/g_{j,i}f(x) = 0$ *i.e.* dès que $\chi_{j,i} \leq \chi_{j,i}^C$. Pour ce faire, pour les niveaux j , $Z_i \geq j \geq k$ tel que $\chi_{k,i} \leq \chi_{k,i}^C$, $\varphi_{j,i}$ est fixé non pas à 0 mais à un nombre arbitrairement grand ce qui permet lors de l'utilisation de l'équation 7.41 (Page 119) d'identifier facilement le niveau k à partir duquel l'ionisation cesse.

Le nombre de particules n_i , $i \neq e$ est :

$$\begin{aligned} n_i &= \sum_{j=0}^{Z_i} n_{j,i} = n_{Z_i,i} \left(1 + \sum_{j=0}^{Z_i-1} \frac{n_{j,i}}{n_{Z_i,i}}\right) = n_{Z_i,i} \left(1 + \sum_{j=0}^{Z_i-1} \prod_{k=j+1}^{Z_i} \varphi_{k,i}\right) \\ &= n_{Z_i,i} \left(1 + \frac{n_{0,i}}{n_{Z_i,i}} + \frac{n_{1,i}}{n_{Z_i,i}} + \dots + \frac{n_{Z_i-1,i}}{n_{Z_i,i}}\right) \\ &= n_{Z_i,i} (1 + \varphi_{1,i}\varphi_{2,i}\varphi_{3,i} \dots \varphi_{Z_i,i} + \varphi_{2,i}\varphi_{3,i} \dots \varphi_{Z_i,i} + \dots + \varphi_{Z_i,i}) \\ &= n_{Z_i,i} (1 + \varphi_{Z_i,i} (1 + \varphi_{Z_i-1,i} (1 + \varphi_{Z_i-2,i} (1 + \dots \varphi_{1,i})))) \end{aligned}$$

et comme :

$$\varphi_{j+1,i} = \frac{n_{j,i}}{n_{j+1,i}} = \frac{n_{j,i}}{n_i} \frac{n_i}{n_{j+1,i}} = \frac{x_{j,i}}{x_{j+1,i}},$$

17. $j e$ est la charge de l'ion.

on a :

$$x_{Z_i,i} \equiv \frac{n_{Z_i,i}}{n_i} = \frac{1}{(1 + \varphi_{Z_i,i}(1 + \varphi_{Z_i-1,i}(1 + \varphi_{Z_i-2,i}(1 + \dots \varphi_{1,i}))))},$$

$$x_{j,i} = \varphi_{j+1,i} x_{j+1,i}, \quad j = Z_i - 1, \dots, 0. \quad (7.46)$$

La charge moyenne de l'élément chimique i et le nombre d'électrons par mole et par unité de volume s'écrivent respectivement :

$$\bar{Z}_i \equiv \sum_{j=1}^{Z_i} j x_{j,i}, \quad x_e = \sum_{j \neq e} \bar{Z}_j x_j, \quad n_e = \rho N_0 x_e = \rho N_0 \sum_{j \neq e} \bar{Z}_j x_j.$$

On a aussi :

$$X_i = \frac{n_i \mu_i m_u}{\sum_j n_j \mu_j m_u} = \frac{n_i \mu_i}{\rho N_0}, \quad x_i = \frac{X_i}{\mu_i} = \frac{n_i}{\rho N_0} \Rightarrow n_i = \rho x_i N_0.$$

7.8.5 Equation de diffusion des espèces chimiques

Pour la particule i , l'équation de diffusion s'écrit (Iben & MacDonald, 1985) :

$$\frac{\partial n_i}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} (r^2 n_i w_i) + \left(\frac{\partial n_i}{\partial t} \right)_{\text{nucl.}}$$

La résolution des équations de Burgers (1969, eq. 18-4,5), voir aussi Cox et al. (1989); Thoul et al. (1994), mènent à la vitesse de diffusion w_i pour la particule i :

$$\frac{dP_{Ci}}{dr} - \frac{\rho_i}{\rho} \frac{dP_C}{dr} - n_i \bar{Z}_i e E = \sum_j K_{ij} (w_j - w_i) + \sum_j K_{ij} z_{ij} \frac{m_j r_i - m_i r_j}{m_i + m_j} \quad (7.47)$$

$$\begin{aligned} \frac{5}{2} n_i k \frac{dT}{dr} = & - \frac{5}{2} \sum_j K_{ij} z_{ij} \frac{m_j (w_j - w_i)}{m_i + m_j} - \frac{2}{5} K_{ii} z_{ii}'' r_i \\ & - r_i \sum_{j \neq i} K_{ij} \frac{3m_i^2 + m_j^2 z_{ij}' + \frac{4}{5} m_i m_j z_{ij}''}{(m_i + m_j)^2} \\ & + \sum_{j \neq i} K_{ij} \frac{m_i m_j (3 + z_{ij}' - \frac{4}{5} z_{ij}'')}{(m_i + m_j)^2} r_j. \end{aligned} \quad (7.48)$$

Les équations de conservation dynamique et statique de la masse et de la charge ont respectivement pour expression :

$$\sum_i \mu_i n_i w_i = 0, \quad e \sum_i \bar{Z}_i n_i w_i = 0, \quad \sum_i \bar{Z}_i n_i = 0, \quad \sum_i X_i = \sum_i \mu_i x_i = 1. \quad (7.49)$$

Les coefficients de résistance du flux de chaleur z_{ij} , z_{ij}' , z_{ij}'' sont définis à partir des intégrales de collision $\Omega_{ij}^{(kl)}$, Paquette et al. (1986) Eq. (23) – (25) :

$$z_{ij} = 1 - \frac{2 \Omega_{ij}^{(12)}}{5 \Omega_{ij}^{(11)}}, \quad z_{ij}' = 2.5 - \frac{2 \cdot 5 \Omega_{ij}^{(12)} - \Omega_{ij}^{(13)}}{5 \Omega_{ij}^{(11)}}, \quad z_{ij}'' = \frac{\Omega_{ij}^{(22)}}{\Omega_{ij}^{(11)}},$$

qui sont déterminées à partir de tabulations. Paquette et al. (1986) ont écrit $\Omega_{ij}^{(kl)}$ sous la forme, Eq.(65) :

$$\Omega_{ij}^{(kl)} \equiv F_{ij}^{(kl)} \epsilon_{ij}, \quad \epsilon_{ij} \equiv \frac{e^4}{4} \sqrt{\frac{\pi}{2k^3}} \frac{\bar{Z}_i^2 \bar{Z}_j^2}{\sqrt{m_{ij}}} \frac{1}{\sqrt{T^3}}.$$

Pour $(kl) = (11), (12), (13), (22)$, et selon qu'il s'agit d'un potentiel attractif, *i.e.* électron/ion, ou répulsif, *i.e.* ion/ion ou électron/électron, Paquette Paquette et al. (1986) ont calculé et tabulé les coefficients d'interpolation par splines naturelles de $\ln F_{ij}^{(kl)}$ en fonction de la variable $\psi_{ij} \equiv \ln \left[\ln \left(1 + \gamma_{ij}^2 \right) \right]$ avec :

$$\gamma_{ij} \equiv \frac{4kT\lambda}{\bar{Z}_i \bar{Z}_j e^2}, \quad \lambda \equiv \max \left\{ \left(\frac{kT}{4\pi e^2 \sum_i n_i \bar{Z}_i} \right)^{\frac{1}{2}}, \left(\frac{3}{4\pi \sum_{\text{ions}} n_i} \right)^{\frac{1}{3}} \right\}.$$

Les quantités K_{ij} sont les coefficients de résistance, ils représentent les effets des collisions entre les particules de types i et j , ils ont pour expression *cf.* Michaud & Proffitt (1993) :

$$K_{ij} = \frac{16}{3} n_i n_j m_{ij} \Omega_{ij}^{(11)}. \quad (7.50)$$

La densité partielle ρ_i de la particule i est définie par ($X_i = x_i \mu_i$) :

$$\rho_i = \rho X_i = \rho x_i \mu_i = n_i \mu_i m_u, \quad \sum_i \rho_i = \rho = \sum_i n_i \mu_i m_u, \quad n_i = \rho N_0 \frac{X_i}{\mu_i} = \rho N_0 x_i. \quad (7.51)$$

L'équation d'équilibre hydrostatique la gravité totale g_T à la pression totale $P = P_C + P_R$ somme de la pression collisionnelle $P_C = nkT$ et de la pression radiative P_R :

$$\frac{dP}{dr} = \rho g_T, \quad g_T = g_G + g_\Omega, \quad g_G = -\frac{Gm}{r^2}, \quad g_\Omega = \frac{2}{3} r \Omega^2 \Rightarrow \frac{\rho_i}{\rho} \frac{dP}{dr} = \rho_i g_T = \rho x_i \mu_i g_T \quad (7.52)$$

En remplaçant la charge de chaque ion par sa charge moyenne tenant compte de son taux d'ionisation, le poids moléculaire moyen est :

$$\mu^{-1} = \sum_{j \neq e} (1 + \bar{Z}_j) \frac{X_j}{\mu_j} = \sum_{j \neq e} (1 + \bar{Z}_j) x_j.$$

Pour les dérivations, on utilisera :

$$d\mu = -\mu^2 \sum_{j \neq e} (1 + \bar{Z}_j) dx_j, \quad \frac{\partial \mu}{\partial x_k} = -\mu^2 (1 + \bar{Z}_k), \quad k \neq e.$$

Les accélérations radiatives ont des valeurs différentes pour les divers isotopes. Pour former les équations il convient de différencier pression totale P , pression gazeuse collisionnelle P_C et pression de radiation P_R . En utilisant les notations habituelles :

$$\beta \equiv \frac{P_C}{P}, \quad 1 - \beta = \frac{P_R}{P}, \quad \left(\frac{\partial \beta}{\partial T} \right)_P = -\frac{4}{T} (1 - \beta), \quad \left(\frac{\partial \beta}{\partial P} \right)_T = \frac{1}{P} (1 - \beta),$$

avec l'hypothèse du gaz parfait, la pression gazeuse collisionnelle s'écrit :

$$P_C = \rho \frac{\mathcal{R}T}{\mu} = P - P_R = P \left(1 - \frac{P_R}{P}\right) = \beta P \Rightarrow \rho = \frac{\beta P \mu}{\mathcal{R}T}, \quad P_R = \frac{a}{3} T^4.$$

$$\frac{dP_R}{dr} = \frac{4}{3} a T^3 \frac{dT}{dr} = \frac{4}{3} a T^3 \frac{T}{P} \frac{\partial \ln T}{\partial \ln P} \frac{dP}{dr} = 4\nabla \frac{a T^4}{3P} \rho g_T = 4\nabla \frac{P_R}{P} \rho g_T = 4\nabla (1 - \beta) \rho g_T.$$

$$\frac{d\beta P}{dr} = \frac{dP_C}{dr} = P \frac{d\beta}{dr} + \beta \frac{dP}{dr},$$

$$\frac{d\beta}{dr} = \frac{\partial \beta}{\partial T} \frac{dT}{dr} + \frac{\partial \beta}{\partial P} \frac{dP}{dr} = -\frac{4}{T} (1 - \beta) \frac{T}{P} \frac{\partial \ln T}{\partial \ln P} \frac{dP}{dr} + \frac{1 - \beta}{P} \frac{dP}{dr} = (1 - 4\nabla) \frac{1 - \beta}{P} \frac{dP}{dr}$$

$$\frac{dP_C}{dr} = [(1 - 4\nabla)(1 - \beta) + \beta] \frac{dP}{dr} = \mathcal{B} \frac{dP}{dr} = \rho \mathcal{B} g_T, \quad \mathcal{B} \equiv 1 - 4\nabla(1 - \beta),$$

$$\frac{dP_R}{dr} = 4\nabla(1 - \beta) \rho g_T = \rho(1 - \mathcal{B}) g_T = \rho g_R, \quad g_R = (1 - \mathcal{B}) g_T,$$

g_R étant l'accélération radiative "thermodynamique". Résultat trivial dû au fait que P_R est un monôme en T . La diffusion différenciée des espèces chimiques ne se produisant que dans les zones radiatives où $\nabla \leq \nabla_{\text{ad}}$:

$$\nabla = \nabla_{\text{rad}} \leq \nabla_{\text{ad}}, \quad 0 \leq \beta \leq 1 \Rightarrow$$

$$\left(\beta = 0, \nabla_{\text{ad}} = \frac{1}{4} \Rightarrow \mathcal{B} = 0 \text{ et } \beta = 1, \nabla_{\text{ad}} = \frac{2}{5} \Rightarrow \mathcal{B} = 1 \right) \Rightarrow 0 \leq \mathcal{B} \leq 1.$$

De façon analogue, en notant P_I la pression totale et P_{Ri} la pression de radiation sur l'isotope i et en admettant que l'équilibre quasi-statique est vérifié pour les ions individuellement :

$$P_I = P_i + P_{Ri}, \quad \frac{dP_I}{dr} = \rho_i g_T$$

en utilisant :

$$n_i = \rho x_i N_0, \quad \mathcal{R} \equiv k N_0, \quad \rho = \frac{\mu P_C}{\mathcal{R}T} = \frac{\mu \beta P}{\mathcal{R}T} \Rightarrow$$

$$P_i = n_i k T = \rho x_i N_0 k T = \frac{\beta \mu P}{\mathcal{R}T} x_i N_0 k T = \beta P \mu x_i.$$

On utilisera par la suite :

$$n_i k = \frac{P}{T} \beta \mu x_i. \quad (7.53)$$

$$\begin{aligned} \frac{dP_i}{dr} &= \frac{d}{dr} \beta P \mu x_i = \mu x_i \frac{d\beta P}{dr} + \beta \mu P \frac{dx_i}{dr} + \beta P x_i \frac{d\mu}{dr} \\ &= \mu x_i \mathcal{B} \rho g_T + \beta P \mu \frac{dx_i}{dr} - \beta P \mu^2 x_i \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \\ &= \rho x_i \mu \mathcal{B} g_T + \beta P \mu \left(\frac{dx_i}{dr} - x_i \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) \end{aligned}$$

Pour l'isotope i , l'accélération radiative g_{Ri} résulte de la pression de radiation, troisième moment de l'intensité radiative dont le calcul détaillé n'est pas effectué par CESAM2k. Avec :

$$\frac{dP_{\text{Ri}}}{dr} = \rho_i g_{\text{Ri}} = \rho x_i \mu_i g_{\text{Ri}},$$

et en admettant que l'équilibre quasi-statique est vérifié pour les ions individuellement :

$$\rho_i g_{\text{T}} = \frac{d}{dr} (P_i + P_{\text{Ri}}) \Rightarrow \frac{dP_i}{dr} = \rho_i g_{\text{T}} - \rho x_i \mu_i g_{\text{Ri}}$$

En regroupant :

$$\begin{aligned} \frac{dP_i}{dr} &= \frac{d}{dr} \beta P \mu x_i - \frac{dP_{\text{Ri}}}{dr} = \mu x_i \frac{d\beta P}{dr} + \beta P \mu \frac{dx_i}{dr} + \beta P x_i \frac{d\mu}{dr} - \rho x_i \mu_i g_{\text{Ri}} \\ &= \mu x_i \mathcal{B} \rho g_{\text{T}} - \rho x_i \mu_i g_{\text{Ri}} + \beta P \mu \frac{dx_i}{dr} - \beta P \mu^2 x_i \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \\ &= \rho x_i (\mu \mathcal{B} g_{\text{T}} - \mu_i g_{\text{Ri}}) + \beta P \mu \left(\frac{dx_i}{dr} - x_i \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) \\ &= \rho x_i g_{\text{T}} \left(\mu \mathcal{B} - \mu_i \frac{g_{\text{Ri}}}{g_{\text{T}}} \right) + \beta P \mu \left(\frac{dx_i}{dr} - x_i \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) \end{aligned}$$

Puisque Equation 7.47 (Page 121) :

$$\frac{\rho_i}{\rho} \frac{dP}{dr} = \rho x_i \mu_i g_{\text{T}}$$

$$\frac{dP_i}{dr} - \frac{\rho_i}{\rho} \frac{dP}{dr} = \rho x_i g_{\text{T}} \left(\mu \mathcal{B} - \mu_i \left(1 - \frac{g_{\text{Ri}}}{g_{\text{T}}} \right) \right) + \beta P \mu \left(\frac{dx_i}{dr} - x_i \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) \quad (7.54)$$

On remarquera que si l'accélération radiative a sa valeur "thermodynamique" :

$$\mu \mathcal{B} - \mu_i \left(1 - \frac{g_{\text{Ri}}}{g_{\text{T}}} \right) = \mu \mathcal{B} - \mu_i (1 - 1 + \mathcal{B}) = \mathcal{B} (\mu - \mu_i),$$

la présence des facteurs \mathcal{B} et β dans les équations des moments et d'énergie assurent la cohérence entre la pression de radiation "thermodynamique" et, pendant "microscopique", l'accélération radiative.

On a par ailleurs :

$$n_i \bar{Z}_i e E = \rho N_0 x_i \bar{Z}_i e E, \quad \sum_i \mu_i x_i w_i = 0, \quad \sum_i \bar{Z}_i x_i w_i = 0. \quad (7.55)$$

On introduit les relations précédentes dans les équations de Burgers Eq. 7.42 (Page 120), Eq. 7.43 (Page 120) et de conservation Eq. 7.50 (Page 123). Les coefficients de résistance K_{ij} sont de la forme :

$$K_{ij} = \frac{16}{3} n_i n_j m_{ij} \Omega_{ij}^{(11)} = \frac{16}{3} (\rho N_0)^2 x_i x_j m_{ij} \Omega_{ij}^{(11)} = k_{ij} x_i x_j \rho^2, \quad k_{ij} \equiv \frac{16 N_0^2}{3} m_{ij} \Omega_{ij}^{(11)}.$$

et, Eq. 7.48 (Page 122) :

$$n_i k \frac{dT}{dr} = \frac{P}{T} \beta \mu x_i \frac{T}{P} \frac{\partial \ln T}{\partial \ln P} \frac{dP}{dr} = \beta \mu x_i \nabla \rho g_T$$

En fonction des x_i et de μ les équations de Burgers s'écrivent :

$$\begin{aligned} & \rho x_i g_T \left(\mu \mathcal{B} - \mu_i \left(1 - \frac{g_{Ri}}{g_T} \right) \right) + \beta P \mu \left(\frac{dx_i}{dr} - x_i \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) - \rho N_0 x_i \bar{Z}_i e E \\ &= \rho^2 \left\{ x_i \sum_j k_{ij} x_j (w_j - w_i) - x_i \sum_j k_{ij} x_j z_{ij} \frac{m_i r_j - m_j r_i}{m_i + m_j} \right\}, \quad i = 1, \dots, c, \\ & - \rho x_i \beta \nabla \mu g_T = \rho^2 \left\{ x_i \sum_j k_{ij} x_j z_{ij} \frac{m_j (w_j - w_i)}{m_i + m_j} + x_i \frac{4}{25} k_{ii} x_i z_{ii}'' r_i \right. \\ & \left. + \frac{2}{5} r_i x_i \sum_{j \neq i} k_{ij} x_j \frac{3m_i^2 + m_j^2 z'_{ij} + \frac{4}{5} m_i m_j z''_{ij}}{(m_i + m_j)^2} \right. \\ & \left. - \frac{2}{5} x_i \sum_{j \neq i} k_{ij} x_j r_j \frac{m_i m_j (3 + z'_{ij} - \frac{4}{5} z''_{ij})}{(m_i + m_j)^2} \right\}, \quad i = 1, \dots, c, e, \end{aligned}$$

ou encore, en posant :

$$\mu_{Ri} \equiv \mu_i \left(1 - \frac{g_{Ri}}{g_T} \right), \quad \frac{\partial \mu_{Ri}}{\partial x_i} = -\frac{\mu_i}{g_T} \left(\frac{\partial g_{Ri}}{\partial x_j} - \frac{g_{Ri}}{g_T} \frac{\partial g_T}{\partial x_j} \right)$$

avec arrangements et division de chaque membre par $x_i \rho^2$:

$$\begin{aligned} & \frac{g_T}{\rho} (\mathcal{B} \mu - \mu_{Ri}) + \frac{\mathcal{R}T}{\rho} \left(\frac{1}{x_i} \frac{dx_i}{dr} - \mu \sum_{j \neq e} (1 + \bar{Z}_j) \frac{dx_j}{dr} \right) = \sum_{j \neq i} k_{ij} x_j (w_j - w_i) \\ & - \sum_{j \neq i} k_{ij} x_j z_{ij} \frac{m_i r_j - m_j r_i}{m_i + m_j} + \frac{\bar{Z}_i e N_0 E}{\rho}, \quad i = 1, \dots, c, \quad (7.56) \\ & - \frac{g_T}{\rho} \mu \beta \nabla = \sum_{j \neq i} k_{ij} x_j z_{ij} \frac{m_j (w_j - w_i)}{m_i + m_j} \\ & + r_i \left(\frac{4}{25} k_{ii} x_i z_{ii}'' + \frac{2}{5} \sum_{j \neq i} k_{ij} x_j \frac{3m_i^2 + m_j^2 z'_{ij} + \frac{4}{5} m_i m_j z''_{ij}}{(m_i + m_j)^2} \right) - \\ & - \frac{2}{5} \sum_{j \neq i} k_{ij} x_j r_j \frac{m_i m_j (3 + z'_{ij} - \frac{4}{5} z''_{ij})}{(m_i + m_j)^2}, \quad i = 1, \dots, c, e. \quad (7.57) \end{aligned}$$

Les données sont la température T , la densité ρ , la gravité g_T , le gradient ∇ , les abondances des ions x_i , $i = 1, \dots, c$, les poids atomiques des éléments, accélérations radiatives et centrifuges. Des abondances, on déduit le nombre d'électrons et le poids moléculaire moyen.

Les inconnues sont les $c+1$ vitesses de diffusion w_1, \dots, w_c, w_e , les $c+1$ flux de chaleur r_1, \dots, r_c, r_e et le champ électrique E , soit $2c + 3$ inconnues. On ajoute aux $2c + 1$ Eq. 7.51 (Page 124) et Eq. 7.52 (Page 124), les deux équations de conservation :

$$\sum_{i=1}^c \mu_i x_i w_i = 0, \quad \sum_{i=1}^e \bar{Z}_i x_i w_i = 0. \quad (7.58)$$

Les Eq. 7.51 (Page 124), Eq. 7.52 (Page 124) écrites pour les x_i et μ et Eq. 7.53 (Page 124) forment le système linéaire :

$$A\omega = \gamma + G D_x, \quad G \equiv \frac{\mathcal{R}T}{\rho} \Delta,$$

on a noté :

$$\omega \equiv (w_1, \dots, w_c, w_e, r_1, \dots, r_c, r_e, E)^T,$$

$$D_x \equiv \left(\frac{dx_1}{dr}, \dots, \frac{dx_c}{dr}, 0, \dots, 0 \right)^T,$$

$$\gamma \equiv \frac{g_T}{\rho} (\mathcal{B}\mu - \mu_{R1}, \dots, \mathcal{B}\mu - \mu_{Rc}, -\mu\beta\nabla, \dots, -\mu\beta\nabla, 0, 0)^T.$$

$$\frac{\partial}{\partial x_j} \left(\frac{g_T}{\rho} (\mathcal{B}\mu - \mu_{Ri}) \right) = \frac{g_T}{\rho} \left(\mathcal{B} \frac{\partial \mu}{\partial x_j} + \mu \frac{\partial \mathcal{B}}{\partial x_j} - \frac{\partial \mu_{Ri}}{\partial x_j} \right) - \left\{ \frac{g_T}{\rho^2} (\mathcal{B}\mu - \mu_{Ri}) \frac{\partial \rho}{\partial x_1} \right\}_{j=1}.$$

$$\frac{\partial \mathcal{B}}{\partial x_j} = -4(1 - \beta) \frac{\partial \nabla}{\partial x_j}, \quad \frac{\partial \mu_{Ri}}{\partial x_j} = -\frac{1}{g_T} \frac{\partial g_{Ri}}{\partial x_j} \text{ ou } -\frac{\partial \mathcal{B}}{\partial x_j}$$

$$\frac{\partial}{\partial x_j} \left(\frac{\mu g_T \beta \nabla}{\rho} \right) = \frac{g_T \beta \nabla}{\rho} \frac{\partial \mu}{\partial x_j} + \left\{ \frac{\mu g_T \beta}{\rho} \left(\frac{\partial \nabla}{\partial x_1} - \frac{\nabla}{\rho} \frac{\partial \rho}{\partial x_1} \right) \right\}_{j=1},$$

L'équation d'état ne donnant que les dérivées de la densité par rapport à $x_1 \equiv H$.

$$\Delta \equiv \text{Diag} \left(\frac{1}{x_1}, \frac{1}{x_2}, \dots, \frac{1}{x_c}, 0, \dots, 0 \right) - \mu \mathcal{M}, \quad (7.59)$$

$$\mathcal{M} \equiv \begin{pmatrix} (1 + \bar{Z}_1) & (1 + \bar{Z}_2) & (1 + \bar{Z}_3) & \dots & (1 + \bar{Z}_c) & 0 & 0 & \dots & 0 & 0 \\ (1 + \bar{Z}_1) & (1 + \bar{Z}_2) & (1 + \bar{Z}_3) & \dots & (1 + \bar{Z}_c) & 0 & 0 & \dots & 0 & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ (1 + \bar{Z}_1) & (1 + \bar{Z}_2) & (1 + \bar{Z}_3) & \dots & (1 + \bar{Z}_c) & 0 & 0 & \dots & 0 & 0 \\ (1 + \bar{Z}_1) & (1 + \bar{Z}_2) & (1 + \bar{Z}_3) & \dots & (1 + \bar{Z}_c) & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 \end{pmatrix} \quad (7.60)$$

La structure de la matrice A , de dimension $[2c + 3 \times 2c + 3]$, est la suivante :

$$\begin{pmatrix} A_w & A_r & A_e \\ A_c & 0 & 0 \end{pmatrix},$$

les dimensions des sous matrices sont respectivement $A_w[2c + 1 \times c + 1]$, $A_r[2c + 1 \times c + 1]$, $A_e[2c + 1 \times 1]$ et $A_c[2 \times c + 1]$. En notant :

$$p_{ij} \equiv k_{ij} x_j, \quad q_{ij} \equiv \frac{k_{ij} x_j z_{ij} m_j}{m_i + m_j} = p_{ij} \frac{z_{ij} m_j}{m_i + m_j},$$

$$A_w \equiv \begin{pmatrix} -\sum_{j \neq 1} p_{1j} & p_{12} & p_{13} & \cdots & p_{1c} & p_{1e} \\ p_{21} & -\sum_{j \neq 2} p_{2j} & p_{23} & \cdots & p_{2c} & p_{2e} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ p_{c1} & p_{c2} & p_{c3} & \cdots & -\sum_{j \neq c} p_{cj} & p_{ce} \\ -\sum_{j \neq 1} q_{1j} & q_{12} & q_{13} & \cdots & q_{1c} & q_{1e} \\ q_{21} & -\sum_{j \neq 2} q_{2j} & q_{23} & \cdots & q_{2c} & q_{2e} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ q_{c1} & q_{c2} & q_{c3} & \cdots & -\sum_{j \neq c} q_{cj} & q_{ce} \end{pmatrix}, \quad (7.61)$$

avec les notations :

$$\bar{q}_{ij} \equiv -\frac{k_{ij}x_j z_{ij} m_i}{m_i + m_j} = -q_{ij} \frac{m_i}{m_j}, \quad (7.62)$$

$$d_i \equiv \frac{4}{25} k_{ii} x_i z''_{ii} + \frac{2}{5} \sum_{j \neq i} k_{ij} x_j \frac{3m_i^2 + m_j^2 z'_{ij} + \frac{4}{5} m_i m_j z''_{ij}}{(m_i + m_j)^2}, \quad (7.63)$$

$$f_{ij} \equiv -\frac{2}{5} k_{ij} x_j \frac{m_i m_j (3 + z'_{ij} - \frac{4}{5} z''_{ij})}{(m_i + m_j)^2}, \quad (7.64)$$

$$A_r \equiv \begin{pmatrix} \sum_{j \neq 1} q_{1j} & \bar{q}_{12} & \bar{q}_{13} & \cdots & \bar{q}_{1c} & \bar{q}_{1e} \\ q_{21} & \sum_{j \neq 2} q_{2j} & q_{23} & \cdots & q_{2c} & q_{2e} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \bar{q}_{c1} & \bar{q}_{c2} & \bar{q}_{c3} & \cdots & \sum_{j \neq c} q_{cj} & \bar{q}_{ce} \\ d_1 & f_{12} & f_{13} & \cdots & f_{1c} & f_{1e} \\ f_{21} & d_2 & f_{23} & \cdots & f_{2c} & f_{2e} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ f_{c1} & f_{c2} & f_{c3} & \cdots & d_c & f_{ce} \\ f_{e1} & f_{e2} & f_{e3} & \cdots & f_{ec} & d_e \end{pmatrix}, \quad (7.65)$$

$$A_e \equiv \frac{N_0 e}{\rho} (\bar{Z}_1, \dots, \bar{Z}_c, 0, \dots, 0)^T, \quad (7.66)$$

$$A_c \equiv \begin{pmatrix} \mu_1 x_1 & \mu_2 x_2 & \cdots & \mu_c x_c & 0 & 0 & \cdots & 0 \\ \bar{Z}_1 x_1 & \bar{Z}_2 x_2 & \cdots & \bar{Z}_c x_c & \bar{Z}_e x_e & 0 & \cdots & 0 \end{pmatrix}. \quad (7.67)$$

La solution s'écrit :

$$\omega = A^{-1} \gamma + A^{-1} G D_x = v + B D_x \Rightarrow, \quad (7.68)$$

ou encore, pour les ions, $i = 1, \dots, c$:

$$w_i = v_i + \sum_{j=1}^c b_{ij} \frac{dx_j}{dr}, \quad i = 1, \dots, c,$$

les v_i sont les c premiers coefficients de :

$$v = A^{-1} \gamma, \quad (7.69)$$

les b_{ij} , ($i, j = 1, \dots, c$), sont les coefficients des c premières lignes et colonnes de

$$B = A^{-1}G. \quad (7.70)$$

Pour l'ion i l'équation de diffusion devient :

$$\begin{aligned} \frac{\partial n_i}{\partial t} &= \frac{1}{r^2} \frac{\partial}{\partial r} (r^2 n_i w_i) + \left(\frac{\partial n_i}{\partial t} \right)_{\text{nucl.}} = \\ \frac{\partial \rho N_0 x_i}{\partial t} &= \frac{1}{r^2} \frac{\partial}{\partial r} \left\{ N_0 r^2 \rho x_i \left(\sum_{j=1}^c b_{ij} \frac{\partial x_j}{\partial r} + v_i \right) \right\} + \left(\frac{\partial \rho N_0 x_i}{\partial t} \right)_{\text{nucl.}} \end{aligned}$$

ou encore, en notant $\frac{D^*}{Dt} \equiv \frac{\partial^*}{\partial t} + \mathcal{V} * \nabla^*$ l'opérateur de dérivation lagrangien "en suivant le mouvement" :

$$\begin{aligned} \frac{Dx_i}{Dt} &= \frac{4\pi}{4\pi r^2 \rho} \frac{\partial}{\partial r} \left\{ \frac{4\pi r^4 \rho^2}{4\pi r^2 \rho} x_i \left(\sum_{j=1}^c b_{ij} \frac{\partial x_j}{\partial r} + v_i \right) \right\} + \left(\frac{Dx_i}{Dt} \right)_{\text{nucl.}} \implies \\ \frac{Dx_i}{Dt} &= \frac{\partial}{\partial m} \left\{ 4\pi r^2 \rho \left(4\pi r^2 \rho x_i \sum_{j=1}^c b_{ij} \frac{\partial x_j}{\partial m} + v_i x_i \right) \right\} + \left(\frac{Dx_i}{Dt} \right)_{\text{nucl.}}. \end{aligned}$$

On a utilisé l'équation de continuité :

$$0 = \frac{\partial \rho}{\partial t} + \nabla * \rho \mathcal{V} = \frac{\partial \rho}{\partial t} + \rho \nabla * \mathcal{V} + \mathcal{V} * \nabla \rho = x_i \frac{\partial \rho}{\partial t} + x_i \rho \nabla * \mathcal{V} + x_i \mathcal{V} * \nabla \rho,$$

et la conservation des particules d'espèce i :

$$\begin{aligned} \nabla * x_i \rho \mathcal{V} = 0 &\implies \rho \mathcal{V} * \nabla x_i = -x_i \rho \nabla * \mathcal{V} - x_i \mathcal{V} * \nabla \rho, \\ \implies \frac{\partial \rho x_i}{\partial t} = x_i \frac{\partial \rho}{\partial t} + \rho \frac{\partial x_i}{\partial t} - x_i \frac{\partial \rho}{\partial t} - x_i \rho \nabla * \mathcal{V} - x_i \mathcal{V} * \nabla \rho &= \rho \left(\frac{\partial x_i}{\partial t} + \mathcal{V} * \nabla x_i \right) = \rho \frac{Dx_i}{Dt}. \end{aligned}$$

En posant :

$$D_i \equiv (d_{i1}, \dots, d_{ic},) \equiv -(x_i b_{i1}, \dots, x_i b_{ic},), \quad i = 1, \dots, c, \quad (7.71)$$

on obtient la forme de l'équation de diffusion utilisée dans CESAM2k¹⁸ :

$$\frac{Dx_i}{Dt} = -\frac{\partial F_i}{\partial m} + \Psi_i, \quad F_i = -4\pi R^2 \rho \left(4\pi R^2 \rho D_i * \nabla_m \mathcal{X} + v_i x_i \right)$$

où $\nabla_m \mathcal{X}$ est le vecteur :

$$\left(\frac{\partial x_1}{\partial m}, \dots, \frac{\partial x_c}{\partial m} \right)^T.$$

18. Avec la variable d'espace Lagrangienne m les opérateurs $\frac{D}{Dt}$ et $\frac{\partial}{\partial t}$ sont équivalents.

7.8.6 Jacobien.

CESAM2*k* intégrant l'équation de diffusion avec un schéma implicite, il est nécessaire de calculer les dérivées :

$$\frac{\partial v_i}{\partial x_k}, \frac{\partial d_{ij}}{\partial x_k} = x_i \frac{\partial b_{ij}}{\partial x_k} + b_{ij} \delta_{ik}, \quad i, j, k = 1, \dots, c, \quad \delta_{ik} = 1 \text{ si } i = k, 0 \text{ sinon.} \quad (7.72)$$

En dérivant :

$$A^{-1}A = I \Rightarrow \frac{\partial A^{-1}}{\partial x_k} A + A^{-1} \frac{\partial A}{\partial x_k} = 0 \Rightarrow \frac{\partial A^{-1}}{\partial x_k} = -A^{-1} \frac{\partial A}{\partial x_k} A^{-1},$$

d'où :

$$\frac{\partial A^{-1}\gamma}{\partial x_k} = \frac{\partial A^{-1}}{\partial x_k} \gamma + A^{-1} \frac{\partial \gamma}{\partial x_k} = -A^{-1} \frac{\partial A}{\partial x_k} A^{-1} \gamma + A^{-1} \frac{\partial \gamma}{\partial x_k} = A^{-1} \left(\frac{\partial \gamma}{\partial x_k} - \frac{\partial A}{\partial x_k} v \right),$$

et de façon similaire :

$$\frac{\partial B}{\partial x_k} = A^{-1} \left(\frac{\partial G}{\partial x_k} - \frac{\partial A}{\partial x_k} B \right). \quad (7.73)$$

7.8.7 Accélération radiatives

L'accélération radiative ressentie par les isotopes résulte de l'impulsion des photons qu'ils absorbent. En raison de l'anisotropie du rayonnement, même dans le milieu épais, pour la plupart des isotopes, la composante radiale de l'accélération radiative diminue la gravité locale et donc la sédimentation. Deux descriptions dues à G. Alécian sont implantées dans CESAM2*k*.

On simplifie en considérant chaque isotope dans l'ensemble de ses états d'ionisation. m étant la masse à l'intérieur de la sphère de rayon r , la gravité totale g_T qui compense le gradient de pression dans l'équilibre quasi-statique, est somme de la partie purement gravitationnelle $g_G = -Gm/r^2$, de l'accélération centrifuge moyenne $g_\Omega = \frac{2}{3}r\Omega^2$ et d'une partie radiative g_R . On postule que g_R est la somme, pondérée par les masses partielles des particules ($\sum_{i \neq e} n_i m_i$) des accélérations radiatives g_{Ri} :

$$g_R = \frac{\sum_{i \neq e} n_i m_i g_{Ri}}{\sum_{i \neq e} n_i m_i} = \frac{\sum_{i \neq e} \rho N_0 X_i g_{Ri}}{\sum_{i \neq e} \rho N_0 X_i} = \frac{\sum_{i \neq e} X_i g_{Ri}}{\sum_{i \neq e} X_i} = \sum_{i \neq e} x_i \mu_i g_{Ri}, \quad (7.74)$$

La première description du calcul des accélérations radiatives¹⁹ est effectuée dans le cadre de l'approximation proposée par Alécian (1985); Alécian & Artru (1990). Sont seuls pris en compte les effets des transitions lié-lié. C'est une approximation semi analytique où la dépendance de l'accélération radiative à la concentration apparaît dans un terme séparé de celui contenant les propriétés atomiques des ions. Cette approximation permet donc de ne calculer qu'un nombre réduit de quantités lorsque la concentration locale évolue au cours du temps. Pour l'élément $i \neq e$ l'accélération radiative est obtenue par une somme pondérée des accélérations radiatives de chaque ion j :

$$g_{Ri} = \sum_{j=0}^{Z_i} x_{j,i} g_{Ri,j}, \quad g_{Ri,j} = g_{R0i,j} \sqrt{1 + \frac{C_{j,i}}{C_{Sj,i}}}, \quad C_{Sj,i} = b \Psi_{j,i}^2, \quad g_{R0i,j} = q_i \Phi_{j,i},$$

19. Ce paragraphe a été rédigé en grande partie par G. Alécian.

où $C_{j,i} = n_{j,i}/n_{\text{H}}$ est la concentration, par rapport à l'hydrogène, de l'ion j de l'élément chimique i , $C_{\text{S}j,i}$ sa concentration de saturation et $g_{\text{R}0i,j}$ l'accélération radiative à la limite de concentration nulle. On remarquera que :

$$\frac{C_{j,i}}{C_{\text{S}j,i}} = \frac{x_{j,i}}{x_{\text{S}j,i}}.$$

Les quantités b et q_i sont données par :

$$b = \frac{m_e m_p \mathcal{C} n_e \kappa}{2e^2 \mu_{\text{H}} \sqrt{T}}, \quad q_i = \frac{\pi k^3 e^2}{2a \mathcal{C}^5 m_e m_u} \frac{l}{Tr^2 \mu_i},$$

\mathcal{C} est la célérité de la lumière, m_p la masse du proton, κ l'opacité moyenne de Rosseland, h la constante de Plank, a la constante de la radiation, k la constante de Boltzman, r le rayon local et l la luminosité locale. Les quantités $\Psi_{j,i}^2$ et $\Phi_{j,i}$ sont tabulées dans le cas de l'approximation #2 décrite dans Alécian et al. (1990) et ce, pour des masses stellaires de $1M_{\odot}$ à $2M_{\odot}$. Les données de la base Topbase sont utilisées. Ces tables, qui correspondent à des masses différentes, sont lues au premier appel de la routine `alecian1` décrite au §8.5 (Page 170). Les transitions radiatives pour les isotopes d'un même élément se produisant à des fréquences très proches, l'opacité monochromatique de l'ensemble des isotopes d'un même élément est pratiquement égale à la somme des opacités monochromatiques de ses divers isotopes. Dans `CESAM2k`, les isotopes apparaissent comme des éléments séparés. Pour le calcul des accélérations radiatives, les isotopes d'un même élément sont identifiés et leurs abondances sont additionnées de façon à traiter correctement l'effet de saturation des transitions lié-lié. Les accélérations radiatives sont ensuite restituées isotope par isotope.

7.9 Diffusion du moment cinétique

`CESAM2k` offre la possibilité de traiter la diffusion du moment cinétique selon le formalisme de Talon et al. (1997) (`tz97`) ou selon celui de Mathis & Zahn (2004) (`mz04`). Ces deux formalismes diffèrent principalement par le calcul de la fluctuation du potentiel gravitationnel, simplifié avec `tz97`, complet avec `mz04`. Les coefficients de diffusion peuvent être calculés suivant les prescriptions de Palacios et al. (2003) ou de Mathis et al. (2004).

La diffusion du moment cinétique et celle des espèces chimiques sont couplées par la turbulence du milieu, principalement au voisinage des limites zone radiative / zone convective. La résolution numérique du système d'équations régissant ces deux processus de diffusion s'est avérée extrêmement instable. La solution retenue a été de résoudre séparément les systèmes d'équations correspondant respectivement au moment cinétique et aux espèces chimiques. La méthode des éléments finis utilisée pour résoudre les équations de la diffusion du moment cinétique ne différant de celle utilisée pour la diffusion des éléments chimiques que par la prise en compte des conditions limites.

7.9.1 Changement de variable $M \rightarrow \nu = \left(\frac{M}{M_\odot}\right)^{\frac{2}{3}}$

Les abondances et la vitesse angulaire étant des variables lagrangiennes, pour l'intégration des équations d'évolution de la composition chimique et du moment cinétique, on utilise la variable d'espace ν . Les transformations eulérien \leftrightarrow lagrangien sont les suivantes :

$$\begin{aligned} M &= M_\odot m ; \nu = m^{\frac{2}{3}} ; m = \nu^{\frac{3}{2}} ; M = M_\odot m ; d\nu = \frac{2}{3} m^{-\frac{1}{3}} dm ; dm = \frac{3}{2} \sqrt{\nu} d\nu \\ dM &= M_\odot dm = M_\odot \frac{3}{2} \sqrt{\nu} d\nu ; \frac{\partial}{\partial M} = \frac{2}{3M_\odot \sqrt{\nu}} \frac{\partial}{\partial \nu} \\ R &= R_\odot r ; dM = \frac{3M_\odot}{2} \sqrt{\nu} d\nu = 4\pi R^2 \rho dR = 4\pi R_\odot^2 r^2 \rho dR \\ dR &= \frac{3M_\odot}{8\pi R_\odot^2 r^2 \rho} \sqrt{\nu} d\nu ; \frac{\partial}{\partial R} = \frac{8\pi R_\odot^2 r^2 \rho}{3M_\odot \sqrt{\nu}} \frac{\partial}{\partial \nu} \end{aligned}$$

7.9.2 Quelques notations

- $l = \frac{L}{L_\odot}$: luminosité,
- μ_i : masse atomique de l'élément chimique i ,
- \bar{z}_i : charge moyenne de l'élément chimique i en tenant compte de l'ionisation,
- \dot{M} : taux de perte de masse par unité de temps,
- $\dot{M}_\Omega \leq 0$: taux de perte de moment cinétique par unité de masse et de temps.

Dans les algorithmes, les variables relatives à la rotation ont un nom ou une extension comprenant **rot**, **rota** ou encore **w**.

7.9.3 Expressions de H_P , H_T , ∇_μ , χ etc... et dérivées

$$\begin{aligned} H_P &= -\frac{\partial R}{\partial \ln P} = \frac{P}{\rho g} = \frac{PR^2}{GM\rho} = \frac{R_\odot^2 r^2 P}{GM_\odot \nu^{\frac{3}{2}} \rho} \\ \nabla &= \frac{\partial \ln T}{\partial \ln P} \\ H_T &= -\frac{\partial R}{\partial \ln T} = -\frac{\partial R}{\partial \ln P} \frac{\partial \ln P}{\partial \ln T} = \frac{H_P}{\nabla} \\ g(M, R) &= \frac{GM}{R^2} = \frac{GM_\odot}{R_\odot^2} \frac{\nu^{\frac{3}{2}}}{r^2}, \ln g = \ln G + \ln M - 2 \ln R \Rightarrow \frac{\partial \ln g}{\partial \ln R} = \frac{\partial \ln M}{\partial \ln R} - 2 = \frac{4\pi R_\odot^3 r^3 \rho}{M_\odot \nu^{\frac{3}{2}}} \\ \frac{\tilde{g}}{g} &= \frac{4R^3}{3GM} \Omega^2 = \frac{4R_\odot^3}{3GM_\odot} \frac{r^3}{\nu^{\frac{3}{2}}} \Omega^2 \\ x_i &= \frac{X_i}{\mu_i}, \mu^{-1} = \sum_i (1 + \bar{z}_i) x_i, \frac{\partial \mu}{\partial x_i} = -\mu^2 (1 + \bar{z}_i) \\ \mu^{-1} &\simeq \frac{1}{16} (20X + 12 - 3Z) = \frac{1}{16} (20X + 12 - 3(1 - X - Y)) = \frac{1}{16} (23X + 3Y + 9) = \\ &\frac{1}{16} (23\nu_H x_H + 3\nu_{He} x_{He} + 9) \end{aligned}$$

$$\begin{aligned} \mu^{-1} &\simeq 2X + \frac{3}{4}Y + \frac{9}{16}Z = \frac{1}{16} [32X + 12(1 - X - Y) + 9Z] = \frac{1}{16} (20X + 12 - 3Z) \\ \frac{\partial \ln \mu}{\partial \nu} &= \frac{1}{\mu} \sum_i \frac{\partial \mu}{\partial x_i} \frac{\partial x_i}{\partial \nu} = -\mu \sum_i (1 + \bar{z}_i) \frac{\partial x_i}{\partial \nu} \simeq -\frac{\mu}{16} \left(23\nu_{\text{H}} \frac{\partial x_{\text{H}}}{\partial \nu} + 3\nu_{\text{He}} \frac{\partial x_{\text{He}}}{\partial \nu} \right) \\ \phi &= -\frac{\mu}{80C_{\text{P}}} \{ (12Z - 48) \ln(64 - 48\mu + 12Z\mu) + (32 - 23Z) \ln(-16 + 32\mu - 23Z\mu) \\ &\quad + 5Z \ln(5Z) + (16 + 6Z) \ln(32 + 16\mu + 6Z\mu) \} \\ \varepsilon(T, \mu) \Rightarrow \varepsilon_{\mu} &= \frac{\partial \ln \varepsilon}{\partial \ln \mu} = \frac{\mu}{\varepsilon} \sum_i \frac{\partial \varepsilon}{\partial x_i} \frac{\partial x_i}{\partial \mu} = -\frac{1}{\mu \varepsilon} \sum_i \frac{1}{1 + \bar{z}_i} \frac{\partial \varepsilon}{\partial x_i} = \frac{\rho}{\varepsilon} \frac{\partial \varepsilon}{\partial \rho} \frac{\partial \ln \rho}{\partial \ln \mu} = \varphi \frac{\rho}{\varepsilon} \frac{\partial \varepsilon}{\partial \rho} \\ \frac{\partial \ln \mu}{\partial X} &\simeq \frac{1}{\mu} \frac{\partial \mu}{\partial X} = -\frac{23\mu}{16}, \quad \frac{\partial \ln \mu}{\partial Y} \simeq -\frac{3\mu}{16} \\ \varepsilon_{\mu} &\simeq \frac{1}{\varepsilon} \left(\frac{\partial \varepsilon}{\partial X} \frac{\partial X}{\partial \ln \mu} + \frac{\partial \varepsilon}{\partial Y} \frac{\partial Y}{\partial \ln \mu} \right) = -\frac{16}{\varepsilon \mu} \left(\frac{1}{23\nu_{\text{H}}} \frac{\partial \varepsilon}{\partial x_{\text{H}}} + \frac{1}{3\nu_{\text{He}}} \frac{\partial \varepsilon}{\partial x_{\text{He}}} \right) \\ \varepsilon_{\text{T}} &= \frac{\partial \ln \varepsilon}{\partial \ln T} = \frac{T}{\varepsilon} \frac{\partial \varepsilon}{\partial T} \\ \varphi &= \frac{\partial \ln \rho}{\partial \ln \mu} = \frac{\mu}{\rho} \frac{\partial \rho}{\partial \mu} = \frac{\mu}{\rho} \sum_i \frac{\partial \rho}{\partial X_i} \frac{\partial X_i}{\partial \mu} = -\frac{1}{\mu \rho} \sum_i \frac{\mu_i}{1 + \bar{z}_i} \frac{\partial \rho}{\partial X_i} \\ \text{dln } \rho &= \alpha \text{dln } P - \delta \text{dln } T + \varphi \text{dln } \mu = (\alpha - \delta \nabla) \text{dln } P + \varphi \text{dln } \mu, \\ \frac{\partial P}{\partial R} &= -\frac{GM\rho}{R^2} \Rightarrow \frac{\partial P}{\partial \nu} = -\frac{3GM_{\odot}^2 \nu^2}{8\pi R_{\odot}^4 r^4} \Rightarrow \frac{\partial \ln P}{\partial \nu} = -\frac{3GM_{\odot}^2 \nu^2}{8\pi R_{\odot}^4 r^4 P} \Rightarrow \\ \frac{\partial \ln \rho}{\partial \nu} &= -\frac{3GM_{\odot}^2 (\alpha - \delta \nabla) \nu^2}{8\pi R_{\odot}^4 r^4 P} + \varphi \frac{\partial \ln \mu}{\partial \nu} \\ \rho_m &= \frac{3M}{4\pi R^3} = \frac{3M_{\odot}}{4\pi R_{\odot}^3} \frac{\nu^{\frac{3}{2}}}{r^3} \\ \text{si } \frac{\partial \ln \mu}{\partial \nu} &\neq 0, \quad \varphi = \left(\frac{\partial \ln \rho}{\partial \nu} - \alpha \frac{\partial \ln P}{\partial \nu} + \delta \frac{\partial \ln T}{\partial \nu} \right) / \left(\frac{\partial \ln \mu}{\partial \nu} \right) \\ \nabla_{\mu} &= \frac{\partial \ln \mu}{\partial \ln P} = \frac{\partial \ln \mu}{\partial R} \frac{\partial R}{\partial \ln P} = -\frac{\partial \ln \mu}{\partial R} H_{\text{P}} = -\frac{8\pi R_{\odot}^2 r^2 \rho}{3M_{\odot} \sqrt{\nu}} H_{\text{P}} \frac{\partial \ln \mu}{\partial \nu} = \\ &\frac{8\pi R_{\odot}^2 r^2 \rho \mu H_{\text{P}}}{3M_{\odot} \sqrt{\nu}} \sum_i (1 + \bar{z}_i) \frac{\partial x_i}{\partial \nu} \simeq \frac{\pi R_{\odot}^2 r^2 \rho}{6M_{\odot} \sqrt{\nu}} H_{\text{P}} \left(23\nu_{\text{H}} \frac{\partial x_{\text{H}}}{\partial \nu} + 3\nu_{\text{He}} \frac{\partial x_{\text{He}}}{\partial \nu} \right) \\ \nabla_{\text{rad}} &= \frac{3\kappa L P}{16\pi a c G M T^4} = \frac{3L_{\odot}}{16\pi a c G M_{\odot}} \frac{\kappa l P}{\nu^{\frac{3}{2}} T^4} \\ N_{\text{T}}^2 &= \frac{g\delta}{H_{\text{P}}} (\nabla_{\text{ad}} - \nabla), \quad N_{\mu}^2 = \frac{g\varphi}{H_{\text{P}}} \nabla_{\mu} \\ \chi &= \frac{4acT^3}{3\kappa\rho}, \quad \ln \chi = 3 \ln T - \ln \kappa - \ln \rho - \ln \frac{4ac}{3} \\ \chi_{\text{T}} &= \frac{\partial \ln \chi}{\partial \ln T} = 3 - \frac{\partial \ln \kappa}{\partial \ln T} - \frac{\partial \ln \rho}{\partial \ln T} = 3 - \frac{T}{\kappa} \frac{\partial \kappa}{\partial T} + \delta \left(1 + \frac{\partial \ln \kappa}{\partial \ln \rho} \right) \\ \chi_{\mu} &= \frac{\partial \ln \chi}{\partial \ln \mu} = -\frac{\partial \ln \kappa}{\partial \ln \mu} - \frac{\partial \ln \rho}{\partial \ln \mu} = -\frac{\mu}{\kappa} \sum_i \frac{\partial \kappa}{\partial x_i} \frac{\partial x_i}{\partial \mu} - \varphi = \frac{1}{\mu \kappa} \sum_i \frac{\partial \kappa}{\partial X_i} \frac{\mu_i}{1 + \bar{z}_i} - \varphi \\ \chi_{\mu} &= \frac{\partial \ln \chi}{\partial \ln \rho} \frac{\partial \ln \rho}{\partial \ln \mu} = -\varphi \left(1 + \frac{\partial \ln \kappa}{\partial \ln \rho} \right) \end{aligned}$$

$$\begin{aligned}
K &= \frac{\chi}{\rho c_P} = \frac{4acT^3}{3\kappa\rho^2c_P} \\
\frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} &= \frac{M}{L} \frac{\partial L}{\partial M} = \frac{\partial \ln L}{\partial \ln M} = \frac{\partial \ln l}{\partial \ln m} \\
f_\varepsilon &= \frac{\bar{\varepsilon}}{\bar{\varepsilon} + \bar{\varepsilon}_g} = \frac{\partial M}{\partial L} \varepsilon_{nuc} = \frac{M_\odot}{L_\odot} \varepsilon_{nuc} \frac{\partial m}{\partial l} \\
f_\varepsilon \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} &= \frac{\bar{\varepsilon}}{\bar{\varepsilon} + \bar{\varepsilon}_g} \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} = \frac{\bar{\varepsilon}}{\varepsilon_m} = \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc}}{l} \nu^{\frac{3}{2}}
\end{aligned}$$

7.9.4 Les coefficients de diffusion D_h , D_v et D_{eff}

7.9.4.1 Formalisme de Mathis, Palacios & Zahn

Les coefficients de diffusivité horizontale D_h et verticale D_v sont donnés par Mathis et al. (2004, eq. 7-10-19) :

$$\begin{aligned}
D_h &= R\sqrt{\beta\Omega R|2V - \alpha U|}, \quad \beta = 1.5 \cdot 10^{-6}, \quad V = \frac{1}{6\rho R} \frac{\partial R^2 \rho U}{\partial R}, \quad \alpha = \frac{1}{2} \frac{\partial \ln R^2 \Omega}{\partial \ln R} \\
D_h^2 &= R^3 \beta \Omega \left| \frac{1}{3R\rho} \frac{\partial R^2 \rho U}{\partial R} - \frac{RU}{2R^2 \Omega} \frac{\partial R^2 \Omega}{\partial R} \right| = R^2 \beta \Omega \left| \frac{1}{3\rho} \frac{\partial R^2 \rho U}{\partial R} - \frac{U}{2\Omega} \frac{\partial R^2 \Omega}{\partial R} \right| = \\
&R^2 \beta \Omega \left| \frac{R^2 \rho}{3\rho} \frac{\partial U}{\partial R} + \frac{2R\rho U}{3\rho} + \frac{R^2 U}{3\rho} \frac{\partial \rho}{\partial R} - \frac{2UR\Omega}{2\Omega} - \frac{UR^2}{2\Omega} \frac{\partial \Omega}{\partial R} \right| = \\
&R^2 \beta \Omega \left| \frac{R^2}{3} \frac{\partial U}{\partial R} + \frac{2}{3} UR + \frac{R^2 U}{3\rho} \frac{\partial \rho}{\partial R} - UR - \frac{UR^2}{2\Omega} \frac{\partial \Omega}{\partial R} \right| = \\
&R^4 \beta \Omega \left| \frac{1}{3} \frac{\partial U}{\partial R} - \frac{U}{3R} + \frac{U}{3\rho} \frac{\partial \rho}{\partial R} - \frac{U}{2\Omega} \frac{\partial \Omega}{\partial R} \right| \Rightarrow \\
D_h^2 &= \frac{R_\odot^4 \beta}{3} r^4 \Omega \left| \frac{8\pi R_\odot^2 r^2 \rho}{3M_\odot \sqrt{\nu}} \left[\frac{\partial U}{\partial \nu} + U \left(\frac{\partial \ln \rho}{\partial \nu} - \frac{3}{2\Omega} \frac{\partial \Omega}{\partial \nu} \right) \right] - \frac{U}{R_\odot r} \right| \\
D_h^2 &= \left| C_1 \Omega \frac{\partial U}{\partial \nu} + C_2 \Omega U - C_3 U \frac{\partial \Omega}{\partial \nu} \right|, \quad C_1 = \frac{8\pi R_\odot^6 \beta r^6 \rho}{9M_\odot \sqrt{\nu}} \\
C_2 &= C_1^* - C_2^*, \quad C_1^* = \frac{8\pi R_\odot^6 \beta r^6 \rho}{9M_\odot \sqrt{\nu}} \frac{\partial \ln \rho}{\partial \nu}, \quad C_2^* = \frac{R_\odot^3 \beta}{3} r^3, \quad C_3 = \frac{4\pi R_\odot^6 \beta r^6 \rho}{3M_\odot \sqrt{\nu}} \\
D_v &= \frac{Ri_c (K + D_h) R^2}{N_T^2 + N_\mu^2 (1 + K/D_h)} \left(\frac{\partial \Omega}{\partial R} \right)^2 = \frac{64\pi^2 Ri_c R_\odot^6 r^6 \rho^2}{9M_\odot^2 \nu} \frac{D_h (1 + \frac{K}{D_h})}{\left[N_T^2 + N_\mu^2 \left(1 + \frac{K}{D_h} \right) \right]} \left(\frac{\partial \Omega}{\partial \nu} \right)^2 \\
D_v &= \begin{cases} \nu_{\text{cin}} & \text{si } D_v < \nu_{\text{cin}} \\ D_v + \nu_{\text{cin}} & \text{sinon} \end{cases}, \quad \nu_{\text{cin}} = \frac{2 \cdot 10^{-15} T^{\frac{5}{2}} \sqrt{\bar{\nu}}}{\rho \bar{z}^4 \ln \lambda}, \quad \lambda = 1.3 \cdot 10^4 T^{\frac{3}{2}} n_e^{-\frac{1}{2}} \\
Ri_c &= \frac{1}{6}, \quad \bar{z} = \frac{\sum_i \bar{z}_i x_i}{\sum_i x_i}, \quad \bar{\nu} = \frac{\sum_i x_i \mu_i}{\sum_i x_i} \\
D_{\text{eff}} &= \frac{R_\odot^2 r^2 U^2}{30D_h} + D_v
\end{aligned}$$

Lorsque la diffusion microscopique est prise en compte, le coefficient D_{eff} de diffusion turbulente généré par la diffusion du moment cinétique est ajouté au coefficient de diffusion turbulente.

7.9.4.2 Formalisme simplifié de Castro, Vauclair & Richard

Castro et al. (2007, eq. 3-4-5) ont utilisé des coefficients de diffusivité simplifiés :

$$\begin{aligned} D_h &= C_h R |U|, \quad C_h = 5 \times 10^4 \\ D_v &= C_v R |U|, \quad C_v = \alpha - \frac{1}{30 C_h}, \quad \alpha = 6 \\ D_{\text{eff}} &= \alpha R |U| \end{aligned}$$

7.9.5 Formalisme de Talon et al. (1997)

7.9.5.1 Les variables

- x_i , $i = 1, \dots, n_c$: abondances par mole
- $y_1 \equiv \Omega$: vitesse angulaire s^{-1} ,
- $y_2 \equiv U$: vitesse de circulation méridienne,
- $y_3 \equiv \Theta = \frac{\tilde{\rho}}{\rho}$: taux de fluctuation de la densité le long d'une isobare
- $y_4 \equiv \Lambda = \frac{\tilde{\mu}}{\mu}$: taux de fluctuation du poids moléculaire moyen le long d'une isobare
- $y_5 \equiv \Psi \equiv \frac{\tilde{T}}{T} = \frac{\varphi \Lambda - \Theta}{\delta}$: taux de fluctuation de la température le long d'une isobare
- $y_6 \equiv \mathcal{T}$: taux de transfert du moment cinétique par unité de masse.
- $y_7 \equiv \Upsilon$: second membre de l'équation de la vitesse de circulation méridienne.

Pour l'intégration numérique, le système des équations de la diffusion du moment cinétique est transformé en un système de 7 équations du premier ordre. La variable indépendante lagrangienne est ν .

7.9.5.2 Transport du moment cinétique

Dans une zone radiative, le moment cinétique par unité de masse, $R^2 \Omega = R^2 y_1$, vérifie l'équation de transport (Talon et al., 1997, eq. 4) :

$$\begin{aligned} \rho \frac{DR^2 \Omega}{Dt} &= \frac{1}{R^2} \frac{\partial}{\partial R} \left[R^4 \rho \left(\frac{\Omega U}{5} + D_v \frac{\partial \Omega}{\partial R} \right) \right] - \rho \dot{\mathcal{M}}_\Omega \Rightarrow \quad (7.75) \\ R_\odot^2 \frac{Dr^2 \Omega}{Dt} &= \frac{8\pi R_\odot^6 r^2 \rho}{3M_\odot R_\odot^2 r^2 \rho \sqrt{\nu}} \frac{\partial}{\partial \nu} \left[r^4 \rho \left(\frac{\Omega U}{5} + \frac{8\pi R_\odot^2 r^2 \rho D_v}{3M_\odot \sqrt{\nu}} \frac{\partial \Omega}{\partial \nu} \right) \right] - \dot{\mathcal{M}}_\Omega \Rightarrow \\ \frac{Dr^2 \Omega}{Dt} &= \frac{8\pi R_\odot^2}{3M_\odot \sqrt{\nu}} \frac{\partial}{\partial \nu} \left[r^4 \rho \left(\frac{\Omega U}{5} + \frac{8\pi R_\odot^2 r^2 \rho}{3M_\odot \sqrt{\nu}} D_v \frac{\partial \Omega}{\partial \nu} \right) \right] - \frac{\dot{\mathcal{M}}_\Omega}{R_\odot^2} \Rightarrow \\ \sqrt{\nu} \frac{Dr^2 \Omega}{Dt} &= \frac{8\pi R_\odot^2}{15M_\odot} \frac{\partial}{\partial \nu} (r^4 \rho \Omega U) + \frac{64\pi^2 R_\odot^4}{9M_\odot^2} \frac{\partial}{\partial \nu} \left(\frac{r^6 \rho^2 D_v}{\sqrt{\nu}} \frac{\partial \Omega}{\partial \nu} \right) - \frac{\dot{\mathcal{M}}_\Omega \sqrt{\nu}}{R_\odot^2} \end{aligned}$$

Pour l'intégration par collocation les équations sont mises sous la forme :

$$C_1 \Omega - C_2 = \frac{\partial \mathcal{T}}{\partial \nu} \quad (7.76)$$

$$\mathcal{T} = C_3 \Omega U + C_4 \frac{\partial \Omega}{\partial \nu} \quad (7.77)$$

$$C_1 = \frac{r^2 \sqrt{\nu}}{\Delta t}, \quad C_2 = (C_{11}^* - C_{12}^*) \sqrt{\nu}, \quad C_{11}^* = \frac{r^2(t) \Omega(t)}{\Delta t}, \quad C_{12}^* = \frac{\dot{M}_\Omega}{R_\odot^2}$$

$$C_3 = \frac{8\pi R_\odot^2}{15M_\odot} r^4 \rho, \quad C_4 = \frac{64\pi^2 R_\odot^4}{9M_\odot^2} \frac{r^6 \rho^2 D_\nu}{\sqrt{\nu}}$$

7.9.5.3 Expressions initiales de \check{E}_Ω et \check{E}_μ

Au paragraphe suivant, dans les expressions initiales de Talon et al. (1997), pour E_Ω et E_μ des termes ont été regroupés. Pour référence, on reproduit les équations originales notées \check{E}_Ω pour \tilde{E}_Ω et \check{E}_μ pour \tilde{E}_μ ; \tilde{E}_Ω et \tilde{E}_μ sont définis plus avant.

$$\check{E}_\Omega = -\frac{\rho_m}{\rho} \left\{ \frac{R}{3} \frac{\partial}{\partial R} \left[H_T \frac{\partial}{\partial R} \left(\frac{\Theta}{\delta} \right) + \left(1 - \frac{\chi_T + 1}{\delta} \right) \Theta \right] - \frac{2H_T}{R} \left(1 + \frac{D_h}{K} \right) \frac{\Theta}{\delta} + \frac{2}{3} \Theta \right\}$$

$$- \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} \left\{ H_T \frac{\partial}{\partial R} \left(\frac{\Theta}{\delta} \right) + \left[f_\varepsilon \left(\frac{\varepsilon_T}{\delta} - 1 \right) + 2 - \frac{\chi_T + 1}{\delta} \right] \Theta \right\}$$

$$\check{E}_\mu = \frac{\rho_m}{\rho} \left\{ \frac{R}{3} \frac{\partial}{\partial R} \left[H_T \frac{\partial}{\partial R} \left(\frac{\varphi}{\delta} \Lambda \right) - \left(\chi_\mu + \frac{\chi_T + 1}{\delta} \varphi \right) \Lambda \right] - \frac{2H_T}{R} \frac{\varphi}{\delta} \Lambda \right\}$$

$$+ \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} \left[H_T \frac{\partial}{\partial R} \left(\frac{\varphi}{\delta} \Lambda \right) + \left(f_\varepsilon \varepsilon_\mu + f_\varepsilon \varepsilon_T \frac{\varphi}{\delta} - \chi_\mu - \frac{\chi_T + 1}{\delta} \varphi \right) \Lambda \right]$$

7.9.5.4 Expression vérifiée par E_Ω

Les quantités concernées²⁰ de \check{E}_Ω et \check{E}_μ ont été regroupées dans E_Ω .

$$E_\Omega = 2 \left(1 - \frac{\Omega^2}{2\pi G \rho} + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} \right) \frac{\tilde{g}}{g} + \tilde{E}_\Omega$$

$$= \frac{8R_\odot^3}{3GM_\odot} \left(1 - \frac{\Omega^2}{2\pi G \rho} + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\partial \ln l}{\partial \ln m} \right) \frac{r^3 \Omega^2}{\nu^{\frac{3}{2}}} + \tilde{E}_\Omega$$

$$\tilde{E}_\Omega = \frac{\rho_m}{\rho} \left\{ \frac{R}{3} \frac{\partial}{\partial R} \left[H_T \frac{\partial}{\partial R} \left(\frac{\varphi \Lambda - \Theta}{\delta} \right) \right. \right.$$

$$\left. - (1 + \chi_T) \frac{\varphi \Lambda - \Theta}{\delta} - \Theta - \chi_\mu \Lambda \right]$$

$$\left. - 2 \frac{H_T}{R} \left(\frac{\varphi \Lambda - \Theta}{\delta} + \frac{R\Theta}{3H_T} - \frac{D_h}{K} \frac{\Theta}{\delta} \right) \right\}$$

$$\tilde{E}_\Omega = \frac{3M_\odot}{4\pi R_\odot^3} \frac{\nu^{\frac{3}{2}}}{r^3 \rho} \left\{ \frac{8\pi R_\odot^3}{9M_\odot} \frac{r^3 \rho}{\sqrt{\nu}} \frac{\partial}{\partial \nu} \left[\frac{8\pi R_\odot^2}{3M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial}{\partial \nu} \Psi \right. \right.$$

$$\left. - (1 + \chi_T) \Psi - \Theta - \chi_\mu \Lambda \right]$$

$$\left. - 2 \frac{H_T}{R_\odot r} \left(\Psi + \frac{R_\odot}{3} \frac{r\Theta}{H_T} - \frac{D_h}{K} \frac{\Theta}{\delta} \right) \right\}$$

20. La quantité $\frac{\Omega_0^2}{2\pi G \rho_0}$ a été introduite par S.Mathis & JP. Zahn, pour éliminer une singularité sur U au centre, quand le cœur est radiatif.

$$\begin{aligned}\tilde{E}_\Omega = & \nu \frac{16\pi R_\odot^2}{9M_\odot} \frac{\partial}{\partial \nu} \left(\frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} \right) \\ & + \frac{2}{3} \nu \frac{\partial}{\partial \nu} \{ -(1 + \chi_T) \Psi - \Theta - \chi_\mu \Lambda \} \\ & - \nu \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \left[\Psi + \left(\frac{R_\odot r}{3H_T} - \frac{D_h}{K\delta} \right) \Theta \right]\end{aligned}$$

7.9.5.5 Expression vérifiée par E_μ

Les quantités concernées de \tilde{E}_Ω et \tilde{E}_μ ont été regroupées dans E_μ .

$$\begin{aligned}E_\mu = & \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} \left[H_T \frac{\partial}{\partial R} \left(\frac{\varphi \Lambda - \Theta}{\delta} \right) \right. \\ & \left. + (f_\varepsilon \varepsilon_T - \chi_T - 1) \frac{\varphi \Lambda - \Theta}{\delta} - (2 - f_\varepsilon) \Theta + (f_\varepsilon \varepsilon_\mu - \chi_\mu) \Lambda \right] = \\ & \nu \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \Psi}{\partial \nu} + \nu \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc}}{l} \sqrt{\nu} (\varepsilon_T \Psi + \Theta + \varepsilon_\mu \Lambda) \\ & - \nu \frac{\partial \ln l}{\partial \ln m} \frac{1}{\nu} [(\chi_T + 1) \Psi + 2\Theta + \chi_\mu \Lambda]\end{aligned}$$

7.9.5.6 Expression vérifiée par $U = y_2$

Dans les zones radiatives :

$$\frac{M}{L} \left[\frac{U c_p T}{H_p} \left(\nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right) - \frac{c_p T}{\delta} \frac{D\Theta}{Dt} \right] \quad (7.78)$$

$$-2 \left(1 - \frac{\Omega^2}{2\pi G \rho} + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\bar{\varepsilon} + \bar{\varepsilon}_g}{\varepsilon_m} \right) \frac{\tilde{g}}{g} = \tilde{E}_\Omega + E_\mu \Rightarrow \quad (7.79)$$

$$\begin{aligned}& \frac{M_\odot}{L_\odot} \frac{\nu^{\frac{3}{2}}}{l} \left[\frac{c_p T}{H_p} \left(\nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right) U - \frac{c_p T}{\delta} \frac{\Theta - \Theta(t)}{\Delta t} \right] \\ & - \frac{8R_\odot^3}{3GM_\odot} \left(1 - \frac{\Omega^2}{2\pi G \rho} + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\partial \ln l}{\partial \ln m} \right) \frac{r^3 \Omega^2}{\nu^{\frac{3}{2}}} = \\ & \nu \left\{ \frac{16\pi R_\odot^2}{9M_\odot} \frac{\partial}{\partial \nu} \left(\frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} \right) - \frac{2}{3} \frac{\partial}{\partial \nu} [(1 + \chi_T) \Psi + \Theta + \chi_\mu \Lambda] \right. \\ & + \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \Psi}{\partial \nu} - \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \left[\Psi + \left(\frac{R_\odot r}{3H_T} - \frac{D_h}{K\delta} \right) \Theta \right] \\ & \left. + \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc}}{l} \sqrt{\nu} (\varepsilon_T \Psi + \Theta + \varepsilon_\mu \Lambda) - \frac{\partial \ln l}{\partial \ln m} \frac{1}{\nu} [(\chi_T + 1) \Psi + 2\Theta + \chi_\mu \Lambda] \right\} \\ & \Rightarrow \\ & \frac{M_\odot}{L_\odot} \frac{\nu^{\frac{3}{2}}}{l \nu} \left[\frac{c_p T}{H_p} \left(\nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right) U - \frac{c_p T}{\delta} \frac{\Theta - \Theta(t)}{\Delta t} \right] \\ & - \frac{8R_\odot^3}{3GM_\odot} \left(1 - \frac{\Omega^2}{2\pi G \rho} + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\partial \ln l}{\partial \ln m} \right) \frac{r^3 \Omega^2}{\nu^{\frac{5}{2}}} - \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \Psi}{\partial \nu}\end{aligned}$$

$$\begin{aligned}
& + \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \left[\Psi + \left(\frac{R_\odot r}{3H_T} - \frac{D_h}{K\delta} \right) \Theta \right] - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc}}{l} \sqrt{\nu} (\varepsilon_T \Psi + \Theta + \varepsilon_\mu \Lambda) \\
& + \frac{\partial \ln l}{\partial \ln m} \frac{1}{\nu} [(\chi_T + 1) \Psi + 2\Theta + \chi_\mu \Lambda] = \frac{\partial}{\partial \nu} \left(\frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} - \frac{2}{3} [(1 + \chi_T) \Psi + \Theta + \chi_\mu \Lambda] \right) \\
\Rightarrow & \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l H_p} \left(\nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right) U - \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l \delta} \frac{\Theta - \Theta(t)}{\Delta t} \\
& - \frac{8R_\odot^3}{3GM_\odot} \frac{r^3}{\nu^{\frac{5}{2}}} \Omega^2 + \frac{4R_\odot^3}{3\pi G^2 M_\odot} \frac{r^3}{\rho \nu^{\frac{5}{2}}} \Omega^4 - \frac{4R_\odot^3 \Omega_0^2}{3\pi G^2 M_\odot \rho_0} \frac{r^3}{\nu^{\frac{5}{2}}} \Omega^2 + \frac{8R_\odot^3}{3GM_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^3}{\nu^{\frac{5}{2}}} \Omega^2 \\
& + \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \Psi - \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \Psi}{\partial \nu} + \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \left(\frac{R_\odot}{3} \frac{r}{H_T} - \frac{D_h}{K\delta} \right) \Theta \\
& - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \varepsilon_T \sqrt{\nu}}{l} \Psi - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \sqrt{\nu}}{l} \Theta - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \varepsilon_\mu \sqrt{\nu}}{l} \Lambda + \frac{\partial \ln l}{\partial \ln m} \frac{\chi_T + 1}{\nu} \Psi \\
& + \frac{2}{\nu} \frac{\partial \ln l}{\partial \ln m} \Theta + \frac{\partial \ln l}{\partial \ln m} \frac{\chi_\mu}{\nu} \Lambda = \frac{\partial}{\partial \nu} \left(\frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} - \frac{2}{3} [(1 + \chi_T) \Psi + \Theta + \chi_\mu \Lambda] \right) \\
\Rightarrow & \frac{4R_\odot^3}{3\pi G^2 M_\odot} \frac{r^3}{\rho \nu^{\frac{5}{2}}} \Omega^4 - \frac{8R_\odot^3}{3GM_\odot} \frac{r^3}{\nu^{\frac{5}{2}}} \left(1 + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\partial \ln l}{\partial \ln m} \right) \Omega^2 \\
& + \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l H_p} \left(\nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right) U - \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l \delta \Delta t} \Theta + \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l \delta \Delta t} \Theta(t) \\
& + \left(\frac{M_\odot}{2\pi R_\odot^3} \frac{\sqrt{\nu}}{r^3 \rho} - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \sqrt{\nu}}{l} + \frac{2}{\nu} \frac{\partial \ln l}{\partial \ln m} - \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho K \delta} D_h \right) \Theta \\
& + \left(\frac{\partial \ln l}{\partial \ln m} \frac{\chi_\mu}{\nu} - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \varepsilon_\mu \sqrt{\nu}}{l} \right) \Lambda + \left(\frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} - \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \varepsilon_T \sqrt{\nu}}{l} + \frac{\partial \ln l}{\partial \ln m} \frac{\chi_T + 1}{\nu} \right) \Psi \\
& - \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \Psi}{\partial \nu} = \frac{\partial}{\partial \nu} \left(\frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} - \frac{2}{3} [(1 + \chi_T) \Psi + \Theta + \chi_\mu \Lambda] \right)
\end{aligned}$$

La forme utilisée pour l'intégration numérique est :

$$(C_5 \Omega^2 - C_7) \Omega^2 + C_{10} + C_6 U + C_{11} \Theta + C_{15} \Lambda + C_{12} \Psi - C_{13} \frac{\partial \Psi}{\partial \nu} = \frac{\partial \Upsilon}{\partial \nu} \quad (7.80)$$

$$\Upsilon = C_{17} \Theta + C_{18} \Lambda + C_{16} \Psi + C_{14} \frac{\partial \Psi}{\partial \nu} \quad (7.81)$$

$$C_5 = \frac{4R_\odot^3}{3\pi G^2 M_\odot} \frac{r^3}{\rho \nu^{\frac{5}{2}}}, \quad C_6 = \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l H_p} \left| \nabla_{ad} - \nabla + \frac{\varphi}{\delta} \nabla_\mu \right|,$$

$$C_7 = \frac{8R_\odot^3}{3GM_\odot} \frac{r^3}{\nu^{\frac{5}{2}}} \left(1 + \frac{\Omega_0^2}{2\pi G \rho_0} - \frac{\partial \ln l}{\partial \ln m} \right), \quad C_{10} = \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l \delta \Delta t} \Theta(t)$$

$$C_{11} = C_2^* - C_3^* + C_4^* - C_5^* - C_{15}^*, \quad C_2^* = \frac{M_\odot}{2\pi R_\odot^3} \frac{\sqrt{\nu}}{r^3 \rho}, \quad C_3^* = \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \sqrt{\nu}}{l}$$

$$C_4^* = \frac{2}{\nu} \frac{\partial \ln l}{\partial \ln m}, \quad C_5^* = \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu} D_h}{r^4 \rho K \delta}, \quad C_{15}^* = \frac{M_\odot}{L_\odot} \frac{c_p T \sqrt{\nu}}{l \delta \Delta t}$$

$$C_{12} = C_6^* - C_7^* + C_8^*, \quad C_6^* = \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho}, \quad C_7^* = \frac{M_\odot}{L_\odot} \frac{\varepsilon_{nuc} \varepsilon_T \sqrt{\nu}}{l}$$

$$\begin{aligned}
C_8^* &= \frac{\partial \ln l}{\partial \ln m} \frac{\chi_T + 1}{\nu}, & C_{13} &= \frac{8\pi R_\odot^2}{3M_\odot} \frac{\partial \ln l}{\partial \ln m} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}}, & C_{14} &= \frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \\
C_{15} &= C_9^* - C_{10}^*, & C_9^* &= \frac{\partial \ln l}{\partial \ln m} \frac{\chi_\mu}{\nu}, & C_{10}^* &= \frac{M_\odot}{L_\odot} \frac{\varepsilon_{\text{nuc}} \varepsilon_\mu \sqrt{\nu}}{l} \\
C_{16} &= -\frac{2}{3}(1 + \chi_T), & C_{17} &= -\frac{2}{3}, & C_{18} &= -\frac{2}{3}\chi_\mu
\end{aligned}$$

7.9.5.7 Expression vérifiée par $\Theta = y_3$

$$\Theta = \frac{\tilde{\rho}}{\bar{\rho}} = \frac{R^2}{3g} \frac{\partial \Omega^2}{\partial R} \quad (7.82)$$

$$\Theta = \frac{R_\odot^2 r^2 R_\odot^2 r^2 8\pi R_\odot^2 r^2 \rho 2\Omega}{3GM_\odot \nu^{\frac{3}{2}} 3M_\odot \sqrt{\nu}} \frac{\partial \Omega}{\partial \nu} = \frac{16\pi R_\odot^6 r^6 \rho \Omega}{9GM_\odot^2 \nu^2} \frac{\partial \Omega}{\partial \nu}$$

La forme utilisée pour l'intégration numérique est :

$$\begin{aligned}
\Theta - C_{19} \Omega \frac{\partial \Omega}{\partial \nu} &= 0 & (7.83) \\
C_{19} &= \frac{16\pi R_\odot^6 r^6 \rho}{9GM_\odot^2 \nu^2}
\end{aligned}$$

7.9.5.8 Fluctuation du poids moléculaire $\Lambda = y_4$

L'expression vérifiée par Λ est un problème de valeur initiale :

$$\frac{D\Lambda}{Dt} = \frac{\nabla_\mu U}{H_P} - \frac{6D_h \Lambda}{R^2}$$

La forme utilisée pour l'intégration numérique est :

$$\begin{aligned}
C_{20}\Lambda - C_{22} - C_{21}U &= 0 \iff \Lambda = \frac{C_{22} + C_{21}U}{C_{20}} & (7.84) \\
C_{20} = C_1^* + C_{13}^*, & C_1^* = \frac{1}{\Delta t}, & C_{13}^* &= \frac{6D_h}{R_\odot^2 r^2}, & C_{21} &= \frac{\nabla_\mu}{H_P}, & C_{22} &= \frac{\Lambda(t)}{\Delta t}
\end{aligned}$$

Expression substituée à Λ dans les équations précédentes.

7.9.5.9 Expression vérifiée par $\Psi = y_5$

$$\Psi = \frac{\varphi \Lambda - \Theta}{\delta}$$

La forme utilisée pour l'intégration numérique est :

$$\begin{aligned}
\Psi - C_{24}\Lambda + C_{23}\Theta &= 0 & (7.85) \\
C_{24} = \frac{\varphi}{\delta}, & C_{23} = \frac{1}{\delta}
\end{aligned}$$

7.9.6 Formalisme de Mathis & Zahn (2004)

7.9.6.1 Les variables

On allège les notations originales :

$$\Phi \Rightarrow \phi, \hat{\phi}_2 \Rightarrow \Phi, \Psi_2 \Rightarrow \Psi, U_2 \Rightarrow U, \Lambda_2 \Rightarrow \Lambda, \mathcal{A}_2 \Rightarrow A.$$

Les indications entre parenthèses, telles que (Equ. B4), renvoient aux équations de l'article de référence.

On note les variables principales par des Majuscules :

1. $x_i, i = 1, \dots, n_c$: abondances par mole
2. $y_1 \equiv \Omega$: vitesse angulaire s^{-1} ,
3. $y_2 \equiv U$: vitesse de circulation méridienne,
4. $y_3 \equiv \Psi \equiv \frac{\tilde{T}}{T} = \frac{\varphi\Lambda - \Theta}{\delta}$: taux de fluctuation de la température le long d'une isobare
5. $y_4 \equiv \Lambda = \frac{\tilde{\mu}}{\mu}$: taux de fluctuation du poids moléculaire moyen le long d'une isobare
6. $y_5 \equiv \mathcal{T}$: taux de transfert du moment cinétique par unité de masse.
7. $y_6 \equiv \Upsilon$: second membre de l'équation de la vitesse de circulation méridienne.
8. $y_7 \equiv \Phi$: taux de fluctuation du potentiel gravitationnel le long d'une isobare.
9. $y_8 \equiv \Pi$: second membre de l'équation de Poisson.

On ramène le système des équations à un système d'équations du premier ordre totalement implicite. Il est résolu par la méthode de collocation.

Les facteurs numériques sont notés par F_n pour les différencier de ceux du formalisme de Talon et al. (1997).

7.9.6.2 Transport du moment cinétique

Dans une zone radiative, le moment cinétique par unité de masse vérifie l'équation de transport vertical (Mathis & Zahn (2004), équation 19) :

$$\begin{aligned} \rho \frac{DR^2\Omega}{Dt} &= \frac{1}{R^2} \frac{\partial}{\partial R} \left[R^4 \rho \left(\frac{\Omega U}{5} + D_v \frac{\partial \Omega}{\partial R} \right) \right] - \rho \dot{\mathcal{M}}_\Omega \Rightarrow \\ R_\odot^2 \frac{Dr^2\Omega}{Dt} &= \frac{8\pi R_\odot^6 r^2 \rho}{3M_\odot R_\odot^2 r^2 \rho \sqrt{\nu}} \frac{\partial}{\partial \nu} \left[r^4 \rho \left(\frac{\Omega U}{5} + \frac{8\pi R_\odot^2 r^2 \rho D_v}{3M_\odot \sqrt{\nu}} \frac{\partial \Omega}{\partial \nu} \right) \right] - \dot{\mathcal{M}}_\Omega \Rightarrow \\ \frac{Dr^2\Omega}{Dt} &= \frac{8\pi R_\odot^2}{3M_\odot \sqrt{\nu}} \frac{\partial}{\partial \nu} \left[r^4 \rho \left(\frac{\Omega U}{5} + \frac{8\pi R_\odot^2 r^2 \rho}{3M_\odot \sqrt{\nu}} D_v \frac{\partial \Omega}{\partial \nu} \right) \right] - \frac{\dot{\mathcal{M}}_\Omega}{R_\odot^2} \Rightarrow \\ \sqrt{\nu} \frac{Dr^2\Omega}{Dt} &= \frac{\partial}{\partial \nu} \left(\frac{8\pi R_\odot^2}{15M_\odot} r^4 \rho \Omega U + \frac{64\pi^2 R_\odot^4}{9M_\odot^2} \frac{r^6 \rho^2 D_v}{\sqrt{\nu}} \frac{\partial \Omega}{\partial \nu} \right) - \frac{\dot{\mathcal{M}}_\Omega \sqrt{\nu}}{R_\odot^2} \end{aligned} \quad (7.86)$$

Avec les notations des algorithmes, les équations s'écrivent :

$$\begin{aligned} F_{26}\Omega - F_{30} &= \frac{\partial \mathcal{T}}{\partial \nu}, \quad \mathcal{T} = F_{31}\Omega U + F_{32} \frac{\partial \Omega}{\partial \nu} \\ F_{26} &= \frac{r^2 \sqrt{\nu}}{\Delta t}, \quad F_{30} = (F_{12}^* + F_{13}^*) \sqrt{\nu}, \quad F_{12}^* = \frac{r^2(t)\Omega(t)}{\Delta t}, \quad F_{13}^* = -\frac{\dot{\mathcal{M}}_\Omega}{R_\odot^2} \\ F_{31} &= \frac{8\pi R_\odot^2}{15M_\odot} r^4 \rho, \quad F_{32} = \frac{64\pi^2 R_\odot^4}{9M_\odot^2} \frac{r^6 \rho^2 D_v}{\sqrt{\nu}} \end{aligned} \quad (7.87)$$

7.9.6.3 Vitesse de circulation méridienne U (Equ. B4)

$$0 = \frac{Mg\rho c_P T}{LP} (\nabla_{\text{ad}} - \nabla) U \quad (7.88)$$

$$+ 2 \left\{ 1 - \frac{1}{6\pi G \rho R^2} \frac{\partial R^3 \Omega^2}{\partial R} - \frac{\epsilon + \epsilon_g}{\epsilon_m} \right\} \left\{ \frac{R^2 \Omega^2}{3g^2} \frac{dg}{dR} - \frac{2R\Omega^2}{3g} - \frac{d}{dR} \left(\frac{\Phi}{g} \right) \right\} \quad (7.89)$$

$$+ \frac{R\Omega}{3\pi G \rho} \frac{\partial \Omega}{\partial R} \quad (7.90)$$

$$- \frac{\rho_m}{\rho} \left\{ \frac{R}{3} \frac{\partial A}{\partial R} - \frac{2H_T}{R} \left(1 + \frac{D_h}{K} \right) \Psi \right\} \quad (7.91)$$

$$- \frac{\epsilon + \epsilon_g}{\epsilon_m} \{ A + [f_\epsilon (\epsilon_T - \delta) + \delta] \Psi + [f_\epsilon (\epsilon_\mu + \varphi) - \varphi] \Lambda \} \quad (7.92)$$

$$+ \frac{M c_P T}{L} \left(\frac{D\Psi}{Dt} + \phi \frac{D \ln \mu}{Dt} \Lambda \right) \quad (7.93)$$

On développe successivement chaque ligne.

7.9.6.3.1 Expressions dérivées de A (Equ. B5) Pour profiter de la réduction d'un ordre de dérivation (donc d'une variable principale) résultant de l'intégration par parties, comme il a été fait avec le groupe d'équations de Talon/Zahn, on divise les équations (Eq. B4) par ν et on remplace \mathcal{A} par son expression (Eq. B5).

$$\begin{aligned} A &= H_T \frac{\partial \Psi}{\partial R} - (1 - \delta + \chi_T) \Psi - (\varphi + \chi_\mu) \Lambda \\ A &= \frac{8\pi R_\odot^2}{3M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} - (1 - \delta + \chi_T) \Psi - (\varphi + \chi_\mu) \Lambda \\ \frac{2}{3} \frac{\partial A}{\partial \nu} &= \frac{\partial}{\partial \nu} \left(\frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}} \frac{\partial \Psi}{\partial \nu} \right) - \frac{\partial}{\partial \nu} \left(\frac{2}{3} (1 - \delta + \chi_T) \Psi \right) - \frac{\partial}{\partial \nu} \left(\frac{2}{3} (\varphi + \chi_\mu) \Lambda \right) = \\ &= \frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right) + \frac{\partial F_{18} \Psi}{\partial \nu} + \frac{\partial F_{11} \Lambda}{\partial \nu}, \\ F_{17} &= \frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}}, \quad F_{18} = -\frac{2}{3} (1 - \delta + \chi_T), \quad F_{11} = -\frac{2}{3} (\varphi + \chi_\mu), \\ \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} A &= \frac{8\pi R_\odot^2}{3M_\odot} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \ln l}{\partial \ln m} \frac{\partial \Psi}{\partial \nu} - \frac{1 - \delta + \chi_T}{\nu} \frac{\partial \ln l}{\partial \ln m} \Psi - \frac{\varphi + \chi_\mu}{\nu} \frac{\partial \ln l}{\partial \ln m} \Lambda \rightarrow \\ \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} A &= -F_{12} \frac{\partial \Psi}{\partial \nu} - \frac{1 - \delta + \chi_T}{\nu} \frac{\partial \ln l}{\partial \ln m} \Psi - \frac{\varphi + \chi_\mu}{\nu} \frac{\partial \ln l}{\partial \ln m} \Lambda, \\ F_{12} &= -\frac{8\pi R_\odot^2}{3M_\odot} \frac{r^2 \rho H_T}{\nu^{\frac{3}{2}}} \frac{\partial \ln l}{\partial \ln m}. \end{aligned}$$

7.9.6.3.2 Equation 7.83 (Page 138)

$$\frac{Mg\rho c_P T}{LP} (\nabla_{\text{ad}} - \nabla) U = \nu \left\{ \frac{M_\odot}{L_\odot} \frac{c_P T \sqrt{\nu}}{l H_P} (\nabla_{\text{ad}} - \nabla) U \right\}$$

7.9.6.3.3 Equation 7.84 (Page 138)

$$\begin{aligned}
& \frac{2}{\nu} \left\{ 1 - \frac{1}{6\pi G\rho R^2} \frac{\partial R^3 \Omega^2}{\partial R} - \frac{\epsilon + \epsilon_g}{\epsilon_m} \right\} = \frac{2}{\nu} - \frac{1}{3\pi G\rho R^2 \nu} \left(3R^2 \Omega^2 + 2R^3 \Omega \frac{\partial \Omega}{\partial R} \right) - \frac{2}{\nu} \frac{\partial \ln l}{\partial \ln m} = \\
& \frac{2}{\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) - \frac{\Omega^2}{\pi G\rho \nu} - \frac{2R}{3\pi G\rho \nu} \Omega \frac{\partial \Omega}{\partial R} \\
& \frac{R^2}{3g^2} \frac{dg}{dR} \Omega^2 - \frac{2R\Omega^2}{3g} - \frac{d}{dR} \left(\frac{\Phi}{g} \right) = \left(\frac{R}{3g} \frac{d \ln g}{d \ln R} - \frac{2R}{3g} \right) \Omega^2 + \frac{1}{Rg} \frac{d \ln g}{d \ln R} \Phi - \frac{1}{g} \frac{\partial \Phi}{\partial R} = \\
& \frac{R}{3g} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^2 + \frac{1}{gR} \frac{d \ln g}{d \ln R} \Phi - \frac{1}{g} \frac{\partial \Phi}{\partial R} \\
& 2\{-\}\{-\} = \nu \left(\frac{2}{\nu} \{-\}\{-\} \right) = \\
& \nu \left\{ \frac{2R}{3g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^2 + \frac{2}{g\nu R} \frac{d \ln g}{d \ln R} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \Phi \right. \\
& \left. - \frac{2}{g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \frac{\partial \Phi}{\partial R} - \frac{R}{3\pi Gg\rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^4 - \frac{1}{\pi Gg\rho \nu R} \frac{d \ln g}{d \ln R} \Omega^2 \Phi \right. \\
& \left. + \frac{1}{\pi Gg\rho \nu} \Omega^2 \frac{\partial \Phi}{\partial R} - \frac{2R^2}{9\pi Gg\rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^3 \frac{\partial \Omega}{\partial R} \right. \\
& \left. - \frac{2}{3\pi Gg\rho \nu} \frac{d \ln g}{d \ln R} \Omega \Phi \frac{\partial \Omega}{\partial R} + \frac{2R}{3\pi Gg\rho \nu} \Omega \frac{\partial \Omega}{\partial R} \frac{\partial \Phi}{\partial R} \right\}
\end{aligned}$$

7.9.6.3.4 Equation 7.85 (Page 139)

$$\frac{R\Omega}{3\pi G\rho} \frac{\partial \Omega}{\partial R} = \nu \left\{ \frac{8R_\odot^3}{9M_\odot G} \frac{r^3}{\nu^{\frac{3}{2}}} \Omega \frac{\partial \Omega}{\partial \nu} \right\}$$

7.9.6.3.5 Equation 7.86 (Page 139)

$$\begin{aligned}
& \frac{\rho_m}{\rho} \left\{ \frac{R}{3} \frac{\partial A}{\partial R} - \frac{2H_T}{R} \left(1 + \frac{D_h}{K} \right) \Psi \right\} = \\
& \frac{3M_\odot}{4\pi R_\odot^3} \frac{\nu^{\frac{3}{2}}}{r^3 \rho} \frac{R_\odot r}{3} \frac{8\pi R_\odot^2}{3M_\odot} \frac{r^2 \rho}{\sqrt{\nu}} \frac{\partial A}{\partial \nu} - \frac{3M_\odot}{4\pi R_\odot^3} \frac{\nu^{\frac{3}{2}}}{r^3 \rho} \frac{2H_T}{R_\odot r} \left(1 + \frac{D_h}{K} \right) \Psi = \\
& \nu \frac{2}{3} \frac{\partial A}{\partial \nu} - \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \nu^{\frac{3}{2}}}{r^4 \rho} \left(1 + \frac{D_h}{K} \right) \Psi = \\
& \nu \left\{ \frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right) + \frac{\partial F_{18} \Psi}{\partial \nu} + \frac{\partial F_{11} \Lambda}{\partial \nu} - \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_T \sqrt{\nu}}{r^4 \rho} \left(1 + \frac{D_h}{K} \right) \Psi \right\}
\end{aligned}$$

7.9.6.3.6 Equation 7.87 (Page 139)

$$\begin{aligned}
& \frac{\epsilon + \epsilon_g}{\epsilon_m} \{ A + [f_\epsilon (\epsilon_T - \delta) + \delta] \Psi + [f_\epsilon (\epsilon_\mu + \varphi) - \varphi] \Lambda \} = \\
& \nu \left\{ \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} A + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} [(f_\epsilon (\epsilon_T - \delta) + \delta)] \Psi + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} [(f_\epsilon (\epsilon_\mu + \varphi) - \varphi)] \Lambda \right\} =
\end{aligned}$$

$$\begin{aligned}
& \nu \left\{ -F_{12} \frac{\partial \Psi}{\partial \nu} + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (f_\epsilon (\epsilon_T - \delta) + 2\delta - 1 - \chi_T) \Psi + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (f_\epsilon (\epsilon_\mu + \varphi) - 2\varphi - \chi_\mu) \Lambda \right\} = \\
& = \nu \left\{ -F_{12} \frac{\partial \Psi}{\partial \nu} + \left(\frac{M_\odot \varepsilon_{\text{nuc}} \sqrt{\nu}}{L_\odot l} (\epsilon_T - \delta) + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\delta - 1 - \chi_T) \right) \Psi \right. \\
& \left. + \left(\frac{M_\odot \varepsilon_{\text{nuc}} \sqrt{\nu}}{L_\odot l} (\epsilon_\mu + \varphi) - \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\varphi + \chi_\mu) \right) \Lambda \right\}
\end{aligned}$$

7.9.6.3.7 Equation 7.88 (Page 139)

$$\begin{aligned}
& \frac{M_{\text{cP}} T}{L} \left(\frac{D\Psi}{Dt} + \phi \frac{D \ln \mu}{Dt} \Lambda \right) = \\
& \nu \left\{ \frac{M_\odot \sqrt{\nu} c_{\text{P}} T}{L_\odot l \Delta t} \Psi + \frac{M_\odot \sqrt{\nu} c_{\text{P}} T \phi}{L_\odot l \Delta t} (\ln \mu - \ln \mu^t) \Lambda - \frac{M_\odot \sqrt{\nu} c_{\text{P}} T \Psi^t}{L_\odot l \Delta t} \right\}
\end{aligned}$$

7.9.6.4 Regroupement

$$\begin{aligned}
0 &= \frac{M_\odot c_{\text{P}} T \sqrt{\nu}}{L_\odot l H_{\text{P}}} (\nabla_{\text{ad}} - \nabla) U \\
&+ \frac{2R}{3g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^2 + \frac{2}{g\nu R} \frac{d \ln g}{d \ln R} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \Phi \\
&- \frac{2}{g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \frac{\partial \Phi}{\partial R} - \frac{R}{3\pi G g \rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^4 - \frac{1}{\pi G g \rho \nu R} \frac{d \ln g}{d \ln R} \Omega^2 \Phi \\
&+ \frac{1}{\pi G g \rho \nu} \Omega^2 \frac{\partial \Phi}{\partial R} - \frac{2R^2}{9\pi G g \rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^3 \frac{\partial \Omega}{\partial R} \\
&- \frac{2}{3\pi G g \rho \nu} \frac{d \ln g}{d \ln R} \Omega \Phi \frac{\partial \Omega}{\partial R} + \frac{2R}{3\pi G g \rho \nu} \Omega \frac{\partial \Omega}{\partial R} \frac{\partial \Phi}{\partial R} + \frac{8R_\odot^3}{9M_\odot G} \frac{r^3}{\nu^{\frac{3}{2}}} \Omega \frac{\partial \Omega}{\partial \nu} \\
&- \frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right) - \frac{\partial F_{18} \Psi}{\partial \nu} - \frac{\partial F_{11} \Lambda}{\partial \nu} + \frac{3M_\odot}{2\pi R_\odot^4} \frac{H_{\text{T}} \sqrt{\nu}}{r^4 \rho} \left(1 + \frac{D_h}{K} \right) \Psi \\
&+ F_{12} \frac{\partial \Psi}{\partial \nu} - \left(\frac{M_\odot \varepsilon_{\text{nuc}} \sqrt{\nu}}{L_\odot l} (\epsilon_T - \delta) + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\delta - 1 - \chi_T) \right) \Psi \\
&- \left(\frac{M_\odot \varepsilon_{\text{nuc}} \sqrt{\nu}}{L_\odot l} (\epsilon_\mu + \varphi) - \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\varphi + \chi_\mu) \right) \Lambda \\
&+ \frac{M_\odot \sqrt{\nu} c_{\text{P}} T}{L_\odot l \Delta t} \Psi + \frac{M_\odot \sqrt{\nu} c_{\text{P}} T \phi}{L_\odot l \Delta t} (\ln \mu - \ln \mu^t) \Lambda - \frac{M_\odot \sqrt{\nu} c_{\text{P}} T \Psi^t}{L_\odot l \Delta t}
\end{aligned}$$

Ou encore :

$$\begin{aligned}
0 &= -\frac{R}{3\pi G g \rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^4 - \frac{2R^2}{9\pi G g \rho \nu} \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^3 \frac{\partial \Omega}{\partial R} \\
&+ \frac{2R}{3g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \left(\frac{d \ln g}{d \ln R} - 2 \right) \Omega^2 + \frac{8R_\odot^3}{9M_\odot G} \frac{r^3}{\nu^{\frac{3}{2}}} \Omega \frac{\partial \Omega}{\partial \nu} + \frac{1}{\pi G g \rho \nu} \Omega^2 \frac{\partial \Phi}{\partial R} \\
&- \frac{1}{\pi G g \rho \nu R} \frac{d \ln g}{d \ln R} \Omega^2 \Phi + \frac{2R}{3\pi G g \rho \nu} \Omega \frac{\partial \Omega}{\partial R} \frac{\partial \Phi}{\partial R} - \frac{2}{3\pi G g \rho \nu} \frac{d \ln g}{d \ln R} \Omega \Phi \frac{\partial \Omega}{\partial R}
\end{aligned}$$

$$\begin{aligned}
& + \frac{2}{g\nu R} \frac{d \ln g}{d \ln R} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \Phi \\
& - \frac{2}{g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \frac{\partial \Phi}{\partial R} - \frac{\partial F_{18} \Psi}{\partial \nu} - \frac{\partial F_{11} \Lambda}{\partial \nu} - \frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right) + F_{12} \frac{\partial \Psi}{\partial \nu} \\
& - \left(\frac{M_{\odot}}{L_{\odot}} \frac{\varepsilon_{\text{nuc}} \sqrt{\nu}}{l} (\epsilon_{\text{T}} - \delta) + \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\delta - 1 - \chi_{\text{T}}) - \frac{M_{\odot}}{L_{\odot}} \frac{\sqrt{\nu} c_{\text{P}} T}{l \Delta t} \right. \\
& - \left. \frac{3M_{\odot}}{2\pi R_{\odot}^4} \frac{H_{\text{T}} \sqrt{\nu}}{r^4 \rho} \left(1 + \frac{D_h}{K} \right) \right) \Psi \\
& - \left(\frac{M_{\odot}}{L_{\odot}} \frac{\varepsilon_{\text{nuc}} \sqrt{\nu}}{l} (\epsilon_{\mu} + \varphi) - \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\varphi + \chi_{\mu}) - \frac{M_{\odot}}{L_{\odot}} \frac{\sqrt{\nu} c_{\text{P}} T \phi}{l \Delta t} (\ln \mu - \ln \mu^t) \right) \Lambda \\
& - \frac{M_{\odot}}{L_{\odot}} \frac{\sqrt{\nu} c_{\text{P}} T \Psi^t}{l \Delta t} + \frac{M_{\odot}}{L_{\odot}} \frac{c_{\text{P}} T \sqrt{\nu}}{l H_{\text{P}}} (\nabla_{\text{ad}} - \nabla) U
\end{aligned}$$

La formulation utilisée s'écrit :

$$\begin{aligned}
& F_1 \Omega^4 + F_2 \Omega^3 \frac{\partial \Omega}{\partial \nu} + F_3 \Omega^2 + F_4 \Omega \frac{\partial \Omega}{\partial \nu} + F_5 \Omega^2 \frac{\partial \Phi}{\partial \nu} + F_6 \Omega^2 \Phi + F_7 \Omega \frac{\partial \Omega}{\partial \nu} \frac{\partial \Phi}{\partial \nu} \\
& + F_8 \Omega \Phi \frac{\partial \Omega}{\partial \nu} + F_9 \Phi + F_{10} \frac{\partial \Phi}{\partial \nu} - \frac{\partial F_{18} \Psi}{\partial \nu} - \frac{\partial F_{11} \Lambda}{\partial \nu} - \frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right) + F_{12} \frac{\partial \Psi}{\partial \nu} \\
& + F_{13} \Psi + F_{14} \Lambda + F_{15} + F_{16} U = 0 \implies \\
& (F_1 \Omega^2 + F_3) \Omega^2 + (F_2 \Omega^2 + F_8 \Phi + F_4) \Omega \frac{\partial \Omega}{\partial \nu} + (F_6 \Omega^2 + F_9) \Phi + F_7 \Omega \frac{\partial \Omega}{\partial \nu} \frac{\partial \Phi}{\partial \nu} \\
& + (F_5 \Omega^2 + F_{10}) \frac{\partial \Phi}{\partial \nu} + F_{12} \frac{\partial \Psi}{\partial \nu} + F_{13} \Psi + F_{14} \Lambda + F_{15} + F_{16} U = \frac{\partial \Upsilon}{\partial \nu} \quad (7.94)
\end{aligned}$$

$$\Upsilon = F_{17} \frac{\partial \Psi}{\partial \nu} + F_{11} \Lambda + F_{18} \Psi \quad (7.95)$$

Les coefficients ont pour expressions :

$$\begin{aligned}
& F_1 \Omega^4, F_1 = -\frac{R_{\odot}}{3\pi G} \frac{r}{\rho \nu g} \left(\frac{d \ln g}{d \ln R} - 2 \right), F_2 \Omega^3 \frac{\partial \Omega}{\partial \nu}, F_2 = -\frac{16R_{\odot}^4}{27M_{\odot} G} \frac{r^4}{g\nu^{\frac{3}{2}}} \left(\frac{d \ln g}{d \ln R} - 2 \right) \\
& F_3 \Omega^2, F_3 = \frac{2R_{\odot}}{3} \frac{r}{g\nu} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \left(\frac{d \ln g}{d \ln R} - 2 \right), F_4 \Omega \frac{\partial \Omega}{\partial \nu}, F_4 = \frac{8R_{\odot}^3}{9M_{\odot} G} \frac{r^3}{\nu^{\frac{3}{2}}} \\
& F_5 \Omega^2 \frac{\partial \Phi}{\partial \nu}, F_5 = \frac{8R_{\odot}^2}{3M_{\odot} G} \frac{r^2}{g\nu^{\frac{3}{2}}}, F_6 \Omega^2 \Phi, F_6 = -\frac{1}{\pi G R_{\odot}} \frac{1}{g\rho\nu r} \frac{d \ln g}{d \ln R}, F_7 \Omega \frac{\partial \Omega}{\partial \nu} \frac{\partial \Phi}{\partial \nu}, F_7 = \frac{128\pi R_{\odot}^5}{27M_{\odot}^2 G} \frac{r^5}{g\nu^{\frac{3}{2}}} \\
& F_8 \Omega \Phi \frac{\partial \Omega}{\partial \nu}, F_8 = -\frac{16R_{\odot}^2}{9M_{\odot} G} \frac{r^2}{g\nu^{\frac{3}{2}}} \frac{d \ln g}{d \ln R}, F_9 \Phi, F_9 = \frac{2}{R_{\odot}} \frac{1}{g\nu r} \frac{d \ln g}{d \ln R} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right) \\
& F_{10} \frac{\partial \Phi}{\partial \nu}, F_{10} = -\frac{16\pi R_{\odot}^2}{3M_{\odot}} \frac{r^2 \rho}{g\nu^{\frac{3}{2}}} \left(1 - \frac{\partial \ln l}{\partial \ln m} \right), \frac{\partial F_{11} \Lambda}{\partial \nu}, F_{11} = -\frac{2}{3} (\varphi + \chi_{\mu}) \\
& F_{12} \frac{\partial \Psi}{\partial \nu}, F_{12} = -\frac{8\pi R_{\odot}^2}{3M_{\odot}} \frac{r^2 \rho H_{\text{T}}}{\nu^{\frac{3}{2}}} \frac{\partial \ln l}{\partial \ln m} \\
& F_{13} \Psi, F_{13} = -(F_1^* + F_2^* + F_3^* + F_4^*), F_1^* = \frac{M_{\odot}}{L_{\odot}} \frac{\varepsilon_{\text{nuc}} \sqrt{\nu}}{l} (\epsilon_{\text{T}} - \delta), \\
& F_2^* = \frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\delta - 1 - \chi_{\text{T}}), F_3^* = -\frac{M_{\odot}}{L_{\odot}} \frac{\sqrt{\nu} c_{\text{P}} T}{l \Delta t}, F_4^* = -\frac{3M_{\odot}}{2\pi R_{\odot}^4} \frac{H_{\text{T}} \sqrt{\nu}}{r^4 \rho} \left(1 + \frac{D_h}{K} \right)
\end{aligned}$$

$$\begin{aligned}
F_{14}\Lambda, F_{14} &= -(F_5^* + F_6^* + F_7^*), F_5^* = \frac{M_\odot}{L_\odot} \frac{\varepsilon_{\text{nuc}}\sqrt{\nu}}{l} (\epsilon_\mu + \varphi), \\
F_6^* &= -\frac{1}{\nu} \frac{\partial \ln l}{\partial \ln m} (2\varphi + \chi_\mu), F_7^* = -\frac{M_\odot}{L_\odot} \frac{\sqrt{\nu} c_P T \phi}{l \Delta t} (\ln \mu - \ln \mu^t) \\
F_{15}, F_{15} &= -\frac{M_\odot}{L_\odot} \frac{\sqrt{\nu} c_P T \Psi^t}{l \Delta t}, F_{16}U, F_{16} = \frac{M_\odot}{L_\odot} \frac{c_P T \sqrt{\nu}}{l H_P} (\nabla_{\text{ad}} - \nabla) \\
\frac{\partial}{\partial \nu} \left(F_{17} \frac{\partial \Psi}{\partial \nu} \right), F_{17} &= \frac{16\pi R_\odot^2}{9M_\odot} \frac{r^2 \rho H_T}{\sqrt{\nu}}, F_{18}\Psi, F_{18} = -\frac{2}{3}(1 - \delta + \chi_T)
\end{aligned}$$

7.9.6.5 Relation barocline (Equ. B6)

$$\begin{aligned}
\Theta &= \frac{R^2}{3g} \frac{\partial \Omega^2}{\partial R} = \frac{2R^2}{3g} \Omega \frac{\partial \Omega}{\partial R} = \varphi\Lambda - \delta\Psi \implies \frac{16\pi R_\odot^4}{9M_\odot} \frac{r^4 \rho}{g\sqrt{\nu}} \Omega \frac{\partial \Omega}{\partial \nu} - \varphi\Lambda + \delta\Psi = 0 \implies \\
F_{19}\Omega \frac{\partial \Omega}{\partial \nu} + F_{20}\Lambda + F_{21}\Psi &= 0 \tag{7.96} \\
F_{19} &= \frac{16\pi R_\odot^4}{9M_\odot} \frac{r^4 \rho}{g\sqrt{\nu}}, F_{20} = -\varphi, F_{21} = \delta
\end{aligned}$$

7.9.6.6 Fluctuations du poids moléculaire (Equ. B7)

$$\begin{aligned}
\frac{D\Lambda}{Dt} + \left(\frac{6D_h}{R^2} - \frac{D \ln \mu}{Dt} \right) \Lambda - \frac{U \nabla_\mu}{H_P} &= 0 \implies F_{22}U + F_{23}\Lambda + F_{24} = 0 \tag{7.97} \\
F_{22} &= -\frac{\nabla_\mu}{H_P}, F_{23} = F_8^* + F_9^*, F_8^* = \frac{6}{R_\odot^2} \frac{D_h}{r^2}, F_9^* = -\frac{\ln \mu - \ln \mu^t - 1}{\Delta t}, F_{24} = -\frac{\Lambda^t}{\Delta t}
\end{aligned}$$

Pour la formation des algorithmes, cette expression a été substituée à Λ dans les équations précédentes.

7.9.6.7 Equation de Poisson (Equ. B8)

$$\begin{aligned}
0 &= \frac{1}{R} \frac{\partial^2 R\Phi}{\partial R^2} - \left(\frac{6}{R^2} + \frac{4\pi G}{g} \frac{\partial \rho}{\partial R} \right) \Phi + \frac{8\pi G}{3} \frac{\rho R}{g} \Omega^2 - \frac{4\pi G}{3g} \frac{\partial}{\partial R} (\rho R^2 \Omega^2) \\
\frac{1}{R} \frac{\partial^2 R\Phi}{\partial R^2} &= \frac{1}{R} \frac{\partial}{\partial R} \left(\Phi + R \frac{\partial \Phi}{\partial R} \right) \\
\frac{8\pi G}{3} \frac{\rho R}{g} \Omega^2 - \frac{4\pi G}{3g} \frac{\partial}{\partial R} (\rho R^2 \Omega^2) &= \\
\frac{4\pi G}{3g} 2R\rho \Omega^2 - \frac{4\pi G}{3g} \left(2R\rho + R^2 \frac{\partial \rho}{\partial R} \right) \Omega^2 - \frac{8\pi G}{3g} \rho R^2 \Omega \frac{\partial \Omega}{\partial R} &= \\
-\frac{4\pi G}{3} \frac{R^2 \rho}{g} \frac{\partial \ln \rho}{\partial R} \Omega^2 - \frac{8\pi G}{3} \frac{\rho R^2}{g} \Omega \frac{\partial \Omega}{\partial R} &\implies \\
0 &= \frac{8\pi R_\odot}{3M_\odot} \frac{r\rho}{\sqrt{\nu}} \frac{\partial}{\partial \nu} \left(\Phi + \frac{8\pi R_\odot^3}{3M_\odot} \frac{r^3 \rho}{\sqrt{\nu}} \frac{\partial \Phi}{\partial \nu} \right) - \left(\frac{6}{R_\odot^2 r^2} + \frac{32\pi^2 R_\odot^2 G}{3M_\odot} \frac{r^2 \rho^2}{g\sqrt{\nu}} \frac{\partial \ln \rho}{\partial \nu} \right) \Phi
\end{aligned}$$

$$-\frac{32\pi^2 R_\odot^4 G r^4 \rho^2}{9M_\odot} \frac{\partial \ln \rho}{g\sqrt{\nu}} \Omega^2 - \frac{64\pi^2 R_\odot^4 G r^4 \rho^2}{9M_\odot} \Omega \frac{\partial \Omega}{g\sqrt{\nu}}$$

En multipliant par $\frac{3M_\odot}{8\pi R_\odot} \frac{\sqrt{\nu}}{r\rho}$:

$$\frac{\partial}{\partial \nu} \left(\Phi + \frac{8\pi R_\odot^3 r^3 \rho}{3M_\odot} \frac{\partial \Phi}{\sqrt{\nu}} \right) - \left(\frac{9M_\odot}{4\pi R_\odot^3 r^3 \rho} \frac{\sqrt{\nu}}{g} + 4\pi R_\odot G \frac{r\rho}{g} \frac{\partial \ln \rho}{\partial \nu} \right) \Phi - \frac{4\pi R_\odot^3 G r^3 \rho}{3} \frac{\partial \ln \rho}{g} \Omega^2 - \frac{8\pi R_\odot^3 G r^3 \rho}{3} \Omega \frac{\partial \Omega}{g} = 0$$

L'équation de Poisson se ramène à :

$$F_{29}\Omega^2 + F_{28}\Omega \frac{\partial \Omega}{\partial \nu} + F_{27}\Phi = \frac{\partial \Pi}{\partial \nu} \quad (7.98)$$

$$\Pi = \Phi + F_{25} \frac{\partial \Phi}{\partial \nu} \quad (7.99)$$

$$F_{25} = \frac{8\pi R_\odot^3 r^3 \rho}{3M_\odot} \frac{1}{\sqrt{\nu}}, \quad F_{27} = F_{10}^* + F_{11}^*, \quad F_{10}^* = \frac{9M_\odot}{4\pi R_\odot^3 r^3 \rho} \frac{\sqrt{\nu}}{g}, \quad F_{11}^* = 4\pi G R_\odot \frac{r\rho}{g} \frac{\partial \ln \rho}{\partial \nu},$$

$$F_{28} = \frac{8\pi G R_\odot^3 r^3 \rho}{3} \frac{1}{g}, \quad F_{29} = \frac{4\pi R_\odot^3 G r^3 \rho}{3} \frac{\partial \ln \rho}{g}$$

7.9.7 Les conditions physiques dans les zones mélangées

Les zones mélangées, *i.e.* zones convectives éventuellement overshootées, apparaissent, disparaissent au cours de l'évolution. L'observation semble montrer que la vitesse angulaire y est constante, il y a **rotation solide**. Dans le milieu turbulent la vitesse de circulation méridienne U n'a pas de sens physique. De fait, le gradient de la vitesse angulaire n'est pas exactement nul, il correspond à un transfert de moment cinétique qui assure la quasi uniformité de la vitesse angulaire. Ce modèle correspond bien à l'image que l'on peut se faire d'un mélange qui, bien que rapide, ne peut être instantané. Son échelle de temps est toutefois petite en regard des autres échelles de temps en présence, celles de Kelving-Helmoltz en particulier.

Avec une rotation solide $\frac{\partial \Omega}{\partial \nu} = 0$. Le moment cinétique par unité de masse $R^2 \Omega$ n'est pas constant. Le moment cinétique total dans une zone convective varie au cours de l'évolution, à cause d'une part des variations des conditions physiques, température, densité composition chimique et, d'autre part en raison des modifications de son étendue. À l'image de ce qui est fait pour la composition chimique, CESAM2k propose deux formalismes pour assurer l'uniformisation de la vitesse angulaire.

Formalisme par diffusion : L'uniformisation de la vitesse angulaire dans les zones convectives est réalisée par une diffusion verticale de coefficient très grand devant l'unité, $D_v \gg 1$. Les équations de transport vertical du moment cinétique 7.71 (Page 133) et 7.82 (Page 138), dans les zone convective, deviennent respectivement :

$$C_1 \Omega - C_2 = \frac{\partial \mathcal{T}}{\partial \nu}, \quad \mathcal{T} = C_4 \frac{\partial \Omega}{\partial \nu} \quad (7.100)$$

$$F_{26} \Omega - F_{30} = \frac{\partial \mathcal{T}}{\partial \nu}, \quad \mathcal{T} = F_{32} \frac{\partial \Omega}{\partial \nu} \quad (7.101)$$

Dans l'équation 7.92 (Page 143) d'évolution des fluctuations du poids moléculaire, $D_h \gg 1$ entraîne $\Lambda = 0$ et, de là, $\Psi = 0$ (*cf.* Eq. 7.91 (Page 143)). Ces conditions

entraînent la discontinuité des gradients des fonctions inconnues aux limites entre zones radiatives et zones mélangées. Pour éviter d'introduire des limites internes mobiles et en nombre variable, ce qui compliquerait l'intégration numérique, des bases de B-splines continues non dérivables sont utilisées pour la diffusion du moment cinétique. Les discontinuités des gradients sont alors implicitement prises en compte par CESAM2k.

Ainsi dans les zones mélangées :

$$\frac{\partial \Omega}{\partial \nu} \sim 0, \quad U = 0, \quad \Theta = 0, \quad \Lambda = 0, \quad \Psi = 0, \quad \Upsilon = 0. \quad (7.102)$$

Dans une zone convective, le taux de transport vertical du moment cinétique se réduit à :

$$\mathcal{T} = \frac{64\pi^2 R_{\odot}^4}{9M_{\odot}^2} \frac{r^6 \rho^2 D_v}{\sqrt{\nu}} \frac{\partial \Omega}{\partial \nu}$$

et l'équation de transport vertical du moment cinétique devient :

$$\begin{aligned} C_1 \Omega - C_2 &= \frac{\partial \mathcal{T}}{\partial \nu}, & \mathcal{T} &= C_4 \frac{\partial \Omega}{\partial \nu} \\ F_{26} \Omega - F_{30} &= \frac{\partial \mathcal{T}}{\partial \nu}, & \mathcal{T} &= F_{32} \frac{\partial \Omega}{\partial \nu} \end{aligned}$$

Dans une zone radiative, la diffusion verticale du moment cinétique procède de deux processus, une advection liée à la vitesse de circulation méridienne et une diffusion proprement dite. Le processus d'échange de moment cinétique entre une zone convective et une zone radiative adjacentes est inconnu. On admet que la vitesse angulaire et le taux de transfert vertical du moment cinétique sont des fonctions continues sur les limites zone convective/zone radiative. En raison des mouvements turbulents à grande échelle dont la zone convective est le siège, le taux de transfert vertical de moment cinétique entre une zone convective et une zone radiative adjacentes est supposé correspondre au taux du processus d'advection affectant la zone radiative.

$$\mathcal{T} = C_3 U \Omega, \quad \mathcal{T} = F_{31} U \Omega$$

La résolution de ces équations conduit à la connaissance de la valeur de la vitesse de circulation méridienne coté radiatif des limites zone radiative / zone convective. La vitesse de circulation méridienne U n'intervient pas dans les équations vérifiées dans les zone convective, cf. équation 7.95 (Page 144). Du point de vue algorithmique, il est possible d'utiliser la variable attachée à U dans l'intégration de l'équation 7.95 (Page 144) et de supposer U comme une variable continue aux limites zone radiative/ zone convective. Ce dispositif, utilisé dans CESAM2k, simplifie les algorithmes. Dans les zones convectives, on définit une vitesse U par :

$$U = \frac{\mathcal{T}}{C_3 \Omega}, \quad U = \frac{\mathcal{T}}{F_{31} \Omega}$$

7.9.8 Les conditions physiques aux limites

Le problème différentiel, dans son ensemble est constitué d'un problème différentiel de conditions initiales pour Λ , d'une définition pour Ψ , de problèmes

problème aux limites de conditions initiales pour Ω , U et Φ . Pour sa résolution numérique, il est ramené à un problème aux limites de conditions initiales continué, suivant le formalisme retenu, de 7 ou 8 équations différentielles et scalaires. La méthode numérique utilisée nécessite la connaissance d'un nombre égal de conditions limites.

Le centre : ($\nu = \nu_0 = 0$), suivant l'état d'évolution le centre peut être situé dans une zone radiative ou dans une zone convective. Les conditions limites utilisées sont :

$$\begin{aligned} \frac{\partial \Omega}{\partial \nu} = 0, \quad U = 0, \quad \Lambda = 0, \quad \Upsilon = 0 \quad (\text{formalisme tz97}) \\ \frac{\partial \Omega}{\partial \nu} = 0, \quad U = 0, \quad \Lambda = 0, \quad \Upsilon = 0, \quad \Phi = 0 \quad (\text{formalisme mz04}) \end{aligned}$$

La surface : La partie externe de l'enveloppe, zone d'ionisation de l'hydrogène, est toujours une zone convective.

Limite externe pour l'équation de Poisson : Suivant Mathis & Zahn (2004) le potentiel gravitationnel Φ vérifie (correction faite d'une erreur d'homogénéité) :

$$\begin{aligned} 3\Phi + R \frac{\partial \Phi}{\partial R} = 0 \Rightarrow \Phi = -\frac{R}{3} \frac{\partial \Phi}{\partial R} \Rightarrow \Phi = -\frac{F_{25}}{3} \frac{\partial \Phi}{\partial \nu} \Rightarrow \\ \Pi = \Phi + F_{25} \frac{\partial \Phi}{\partial \nu} = \frac{2}{3} F_{25} \frac{\partial \Phi}{\partial \nu} = -2\Phi \end{aligned}$$

Les conditions limites utilisées sont :

$$\begin{aligned} U = 0, \quad \Psi = 0, \quad \Theta = 0 \quad (\text{formalisme tz97}) \\ \mathcal{T} = 0, \quad \Psi = 0, \quad \Pi + 2\Phi = 0 \quad (\text{formalisme mz04}) \end{aligned}$$

7.9.9 Les conditions initiales

Des conditions initiales doivent être définies pour la vitesse angulaire Ω_0 , la vitesse de circulation méridienne U_0 et le taux de fluctuation du poids moléculaire moyen Λ_0 . Les conditions initiales sont établies par la routine PRIVATE `initialise_rota` du module MOD_CESAM.

Vitesse angulaire initiale : on ne dispose pas de données observationnelles permettant d'inférer la répartition spatiale de la vitesse angulaire d'un modèle initial de pré-séquence principale ou de séquence principale d'âge zéro. Faute de mieux, CESAM2*k* suppose la rotation solide. La vitesse angulaire initiale est déduite de la variable `W_ROT` du fichier de données `mon_modelle.don`, cf. § 3.3 (Page 26).

Il est toutefois possible d'utiliser un profil de rotation grossièrement similaire à celui que l'on peut inférer des observations de sismologie solaire :

$$\Omega_0(r, 0) = \Omega_0 \left\{ 1 + q \left[1 + \cos \left(\pi \frac{r}{R_\star} \right) \right] \right\}$$

Ω_0 est la vitesse angulaire initiale, éventuellement déduite de la période de rotation et du rayon total initial, cf. § 3.3 (Page 26). $q \geq -1$ est un paramètre permettant

de fixer le rapport entre la vitesse angulaire au centre et à la surface, $q = 0$ correspond à une rotation rigide. q est la variable `w_form` du module `MOD_DONNEES` elle y est fixée à 0. Pour utiliser une valeur différente, il convient de la redéfinir à l'aide d'un fichier de "réglages", cf. § 4.9 (Page 60). Avec $q \neq 0$, le profil de Ω_0 n'est pas iniformisé dans les zone convective. Ω est initialisé dans la routine `w_initial` du module `MOD_CESAM`.

Vitesse de circulation méridienne initiale : cette quantité n'est pas accessible à l'observation. De façon euristique, dans la routine `u_initial` du module `MOD_CESAM`, U_0 est obtenu par la relation empirique :

$$U_0(R) = \frac{|w_{\text{rot}}|u_i}{2 - (R/R_*)^2}$$

où u_i est un paramètre fixé à 10^{-4} et w_{rot} la valeur de la variable `W_ROT` du fichier de données `mon_modele.don`.

Taux de fluctuation du poids moléculaire moyen : Λ est supposé nul à l'instant initial : $\Lambda_0 = 0$.

Conditions initiales pour Θ , et Ψ : elles sont nécessaires pour la résolution numérique, la formulation des équations étant implicite. `CESAM2k` les calcule à partir des relations les définissant, cf. équation 7.77 (Page 136) et équation 7.80 (Page 137), en utilisant les valeurs initiales retenues pour Ω et Λ .

Conditions initiales pour Φ : Le potentiel gravitationnel initial doit vérifier l'équation de Poisson. Il est obtenu en intégrant l'équation de Poisson du formalisme de Mathis & Zahn (2004) par éléments finis.

7.9.10 Pertes / gains de moment cinétique

Par l'intermédiaire de la routine générique `pertw` `CESAM2k` offre la possibilité d'utiliser différentes formes de pertes/gains de moment cinétique. Chacune d'elles dépend d'un paramètre libre, respectivement a , γ , ϖ , transmis par la variable `p_pertw` du fichier de données, cf. § 3.3 (Page 26). Une valeur négative/positive de ce paramètre correspond à une perte/gain de moment cinétique. Cette perte/gain de moment cinétique ne concerne que la zone convective externe. Les formes suivantes sont implantées dans `CESAM2k` :

- `pertw_sch` : La variation temporelle de moment cinétique par unité de masse et de temps, proportionnelle à Ω^3 (Schumanish), est modélisée par :

$$\dot{\mathcal{M}}_{\Omega} = aR^2\Omega_s^3, \quad |a| \sim 1.10^{-9}\text{s}.$$

- `pertw_loc` : Perte / gain de moment cinétique proportionnel à l'énergie cinétique de rotation locale. La variation temporelle de moment cinétique par unité de masse est modélisée par :

$$\dot{\mathcal{M}}_{\Omega} = \gamma R^2\Omega^2, \quad |\gamma| \sim 10^{-13} \sim 10^{-14}.$$

- `pertw_ptm` : Perte / gain de moment cinétique conséquence d'une perte / gain de masse $\dot{\mathcal{M}}$. Dans le cas d'une perte de masse, $\dot{\mathcal{M}} < 0$, on suppose qu'une

couche externe se détache et emporte avec elle son moment cinétique, alors $\varpi = 1$. La perte de moment cinétique par unité de masse est approchée par :

$$\dot{\mathcal{M}}_{\Omega} = \varpi \frac{\dot{\mathcal{M}}}{M_{ZC}} R_{ZC}^2 \Omega_{ZC}.$$

Dans le cas d'un gain de masse, $\dot{\mathcal{M}} > 0$, on suppose que l'apport de masse est animé d'une vitesse angulaire égale à celle de la couche externe avec un rayon de giration égal au rayon de l'étoile, $\varpi > 0$ peut être utilisé pour adapter ces dispositions.

- **pertw_0** : On ignore les perte / gain de moment cinétique. La variation temporelle de moment cinétique est modélisée par :

$$\dot{\mathcal{M}}_{\Omega} = 0.$$

Chutes de planètes : un gain/perte de moment cinétique de la zone convective externe résulte des chutes de planètes, cf. § 7.8.2 (Page 116). En notant Ω_{pl} la vitesse angulaire des planètes, $R_{pl} \geq R_{\star}$ le rayon de giration des planètes, l'apport de moment cinétique par unité de masse est :

$$\dot{\mathcal{M}}_{\Omega} = \frac{N_P M_{\oplus}}{M_{ZC}} \mathcal{P}(t) R_{pl}^2 \Omega_{pl}.$$

$N_P > 0$ est le nombre total de planètes de masse terrestre, M_{\oplus} , reçus par l'étoile, $\mathcal{P}(t)$ la fonction décrivant la dépendance temporelle, cf. § 4.6 (Page 57). Une vitesse angulaire positive des planètes accroît le moment cinétique total de la zone convective externe.

7.10 La convection

7.10.1 Critères de convection

L'énergie est transportée par convection lorsqu'est vérifié cf. Cox & Giuli (1968, p. 276) ou encore Kippenhahn & Weigert (1991, p. 39) :

- soit le critère de Ledoux, qui tient compte du gradient de composition chimique :

$$\nabla_{\text{rad}} < \nabla_{\text{ad}}^* + \frac{\varphi}{\delta} \nabla_{\mu} \sim \frac{4 - 3\beta}{\beta} (\nabla_{\text{ad}}^* - \nabla_{\text{rad}}) + \nabla_{\mu} < 0, \quad (\varphi \equiv \frac{d \ln \rho}{d \ln \mu}, \quad \nabla_{\mu} \equiv \frac{d \ln \mu}{d \ln P})$$

ici $\beta = P_{\text{gaz}}/P$, P_{gaz} est la pression gazeuse, $P = P_{\text{gaz}} + P_{\text{rad}}$ est la pression totale, $P_{\text{rad}} = a/3T^4$ est la pression de radiation et :

$$\mu^{-1} = \mu_i^{-1} + \mu_e^{-1} \sim 2X + \frac{3}{4}Y + \frac{Z}{2} = \frac{3 + 5X - Z}{4}, \quad \nabla_{\mu} = \frac{4\pi R^4 P}{GM} \frac{\partial \mu}{\mu} \frac{\partial X}{\partial M},$$

- soit le critère de Schwarzschild :

$$\nabla_{\text{ad}}^* - \nabla_{\text{rad}} < 0.$$

On a noté ∇_{ad}^* le gradient corrigé de la pression turbulente :

$$\nabla_{\text{ad}}^* = \frac{\Gamma_2 - 1}{\Gamma_2} \frac{d \ln P_{\text{gaz}}}{d \ln P}. \quad (7.103)$$

Le calcul de $\frac{\varphi}{\delta} \nabla_{\mu}$ nécessite la connaissance des taux d'ionisation partielle des divers éléments. Ces quantités ne sont pas explicitement accessibles avec les équations d'état tabulées dont on dispose. Pour le critère de Ledoux on utilise l'approximation "gaz parfait avec radiation" de l'équation d'état (Cox & Giuli, 1968, Eq.3-30, p. 276) :

$$P = \frac{\rho \mathcal{R} T}{\mu} + \frac{a}{3} T^4$$

mène à l'approximation fréquemment utilisée :

$$\frac{\varphi}{\delta} \sim \frac{\beta}{4 - 3\beta}.$$

Suivant la valeur `.TRUE.` ou `.FALSE.`, de la variable logique `ledoux` de la NAMELIST `NL_CONV` du fichier de données, l'un ou l'autre de ces deux critères est utilisé. Le critère de convection est formulé dans les routines `thermo`, cf. § 8.131 (Page 236) `thermo_atm`, cf. § 8.132 (Page 237) et `dgrad`, cf. § 8.28 (Page 182). Dans la restitution de la partie convective de l'atmosphère, le gradient radiatif sera modifié afin d'assurer la continuité du gradient de température, cf. § 7.3.3 (Page 94).

7.10.2 Calcul du gradient convectif

La routine `conv_jmj`, cf. § 8.22 (Page 179), utilise un formalisme proche de celui de Henyey et al. (1965) :

$$\nabla - \nabla_{\text{ad}}^* = \frac{\Gamma(\Gamma + 1)}{B}, \quad \nabla_{\text{ad}}^* = \frac{\Gamma_2 - 1}{\Gamma_2} \frac{d \ln P_{\text{gaz}}}{d \ln P}, \quad B = \xi \frac{l^4 g \delta (\rho c_p)^2}{H_p K^2}, \quad (7.104)$$

$$K = \frac{4acT^3}{3\kappa\rho}, \quad g = \frac{GM}{R^2}, \quad \xi = \frac{1}{72} \left(3 \frac{V}{Al}\right)^2 \left(1 + \frac{2Al}{3V\tau^2}\right)^2,$$

P_{gaz} est la pression gazeuse *i.e.* thermodynamique + radiative et P est la pression totale *i.e.* gazeuse + turbulente, cf. § 7.10.4 (Page 151), $P = P_{\text{gaz}} + P_{\text{tur}}$, τ est l'épaisseur optique Rosseland de l'élément convectif :

$$\tau = \kappa \rho l = \kappa \rho \alpha H_p,$$

$l = \alpha H_p$ est la longueur de mélange, $H_p = -\frac{dR}{d \ln P} = P/\rho g$ est l'échelle de hauteur de pression. Γ est l'efficacité de la convection, zéro réel de la cubique :

$$\Phi \Gamma^3 + \Gamma(\Gamma + 1) = B(\nabla_{\text{rad}} - \nabla_{\text{ad}}^*), \quad (7.105)$$

$$\Phi = \varphi \left(1 + \frac{2Al}{3V\tau^2}\right)^{-1}, \quad \varphi = \frac{3}{2} \left(3 \frac{V}{Al}\right)^{-1}, \quad \nabla_{\text{rad}} = \frac{3}{16\pi acG} \frac{\kappa LP}{mT^4}.$$

La quantité V/A est le rapport du volume/surface de l'élément convectif. Le terme correctif $\frac{\partial \ln P_{\text{gaz}}}{\partial \ln P}$ qui affecte le gradient adiabatique $\nabla_{\text{ad}} \equiv \frac{\Gamma_2 - 1}{\Gamma_2}$ dans l'expression de

∇^* , *cf.* Eq. 7.99 (Page 149), est introduit pour avoir une estimation du gradient de température lors d'une transformation adiabatique avec pression turbulente, ce qui est très approximatif et physiquement mal établi.

En un point d'une zone convective, il semble irréaliste que la longueur de mélange puisse être supérieure à la distance qui sépare ce point de la plus proche limite de la zone convective. Ainsi que Eggleton (1972) l'a proposé, dans la routine `conv_a0`, *cf.* § 8.19 (Page 179), la longueur de mélange est prise égale à $l = \alpha_0 H_p$ avec $\alpha_0 \equiv \alpha(1 - \nabla_{\text{ad}}^* / \nabla_{\text{rad}})$, la longueur de mélange utilisée devenant nulle à chaque limite zone radiative / zone convective. Cette approximation est justifiée par le fait que les grandes incertitudes inhérentes à la théorie ne justifient pas d'introduire une trop grande rigueur dans les détails, *cf.* Cox & Giuli (1968). Par ailleurs, l'introduction de la distance exacte change la nature du problème, il devient intégral-différentiel ce qui complique considérablement la résolution numérique. Introduire la distance exacte à la plus proche limite n'est pas immédiat, du moins algorithmiquement, car le problème devient globalement implicite. Ce qu'évite la formulation locale d'Eggleton.

Dans `conv_jmj` et `conv_a0` ces équations sont résolues avec $\xi = 1/162$, $\Phi = 9/4$ et $V/Al = 2/9$. La recherche du zéro de la cubique est faite à l'aide de l'algorithme de déflation.

La mise en œuvre de `conv_jmj` a bénéficié de la collaboration de J.Provost et de M.J.Goupil.

Dans la routine `conv_cm`, *cf.* § 8.20 (Page 179), le gradient de température dans les zones convectives est calculé selon les prescriptions de Canuto & Mazitelli (1991); l'expression (63) de ces auteurs, mise sous la forme implicite équivalente :

$$0 = \nabla + \Phi(\nabla - \nabla_{\text{ad}}^*) - \nabla_{\text{rad}}$$

est résolue itérativement. Φ est donné par Canuto & Mazitelli (1991, eq. 32) :

$$\Phi = a_1 \Sigma^m [(1 + a_2 \Sigma)^n - 1]^p, \quad \Sigma = 4A^2(\nabla - \nabla_{\text{ad}}^*), \quad A = \frac{l^2}{9\chi} \sqrt{\frac{g}{2H_p}}, \quad \chi = \frac{K}{\rho c_p}$$

ainsi que (5) et (6) avec $a_1 = 24.868$, $a_2 = 9.7666 \times 10^{-2}$, $m = 0.14972$, $n = 0.18931$ et $p = 1.8503$. Comme définie précédemment, la longueur de mélange est prise égale à $l = \alpha H_p$. L'efficacité de la convection est alors l'expression (8) :

$$\Gamma = \frac{\sqrt{\Sigma + 1} - 1}{2}.$$

Dans la routine `conv_cm_reza`, *cf.* § 8.21 (Page 179), similaire à `conv_cm`, il est tenu compte de la quantité thermodynamique $\delta = \left(\frac{\partial \ln \rho}{\partial \ln T}\right)_P$. Dans la routine `conv_cgm_reza`, *cf.* § 8.21 (Page 179) le calcul du gradient convectif est effectué selon les prescriptions de Canuto et al. (1996) avec la prescription de Bernkopf (Heiter et al. (2002)) Ces deux routines ont été mises en œuvre et mises à la disposition des utilisateurs de CESAM2k par Reza Samadi, LESIA, Observatoire de Paris.

7.10.3 Echelle de temps de la convection

Le temps de retournement d'un élément convectif est une estimation de l'échelle de temps de la convection. Il dépend de la zone convective à l'intérieur de laquelle il varie. Une possibilité est de le caractériser à rayon R_c distant d'une demi longueur de mélange l_m du fond R_0 de la zone convective considérée :

$$R_c = R_0 + l_m/2 = R_0 + \frac{\alpha}{2} H_p(R_c), \quad r_c R_\odot = r_0 R_\odot + \frac{\alpha}{2} \frac{Pr_c^2}{m\rho} \frac{R_\odot^2}{GM_\odot}$$

$$r_c = r_0 + \frac{\alpha R_\odot}{2GM_\odot} \frac{Pr_c^2}{m\rho}, \quad \implies \begin{cases} \wp P r_c^2 - m\rho r_c + m\rho r_0 = 0, & \text{si } r_0 > 0 \\ \wp P r_c - m\rho = 0, & \text{si } r_0 = 0 \end{cases}$$

Avec, pour la vitesse de convection v_c , (Kippenhahn & Weigert, 1991, eq. 7.11,20) :

$$v_c = \frac{6acT^3\Gamma}{\kappa\rho^2c_p l_m} = \frac{6acT^3\Gamma}{\kappa\rho^2c_p\alpha H_P}$$

pour une zone convective fixée, l'échelle de temps τ_c de la convection peut donc être estimée par :

$$\tau_c = l_m/v_c = \frac{(\alpha H_p)^2 \kappa\rho^2 c_p}{6acT^3\Gamma} = \frac{1}{6ac} \left(\frac{\alpha R_\odot^2}{GM_\odot} \right)^2 \left(\frac{Pr^2}{m} \right)^2 \frac{\kappa c_p}{T^3\Gamma}$$

les valeurs des diverses quantités étant prises en r_c et Γ étant l'efficacité de la convection en ce point.

7.10.4 Pression turbulente

La théorie de la pression turbulente est très approximative ; son expression est donnée par la relation phénoménologique :

$$P_{\text{turb}} = \rho \langle v^2 \rangle = a\rho v^2, \quad v^2 = \frac{\Gamma}{1+\Gamma} \frac{\delta\alpha^2 P}{8\rho} (\nabla - \nabla_{\text{ad}}^*),$$

où v est la "vitesse" des éléments convectifs ; la valeur du coefficient phénoménologique a est mal déterminée, et varie suivant les auteurs, $a \sim 0.21$ pour Canuto & Mazitelli (1991), $a \sim 0.50$ chez Henyey et al. (1965) ou encore, $a = \pi/2$.

Dans les conditions de la zone convective solaire, ce n'est que lorsque la température est inférieure à 50 000K que le rapport P_{turb}/P dépasse 1/10 000 ; ce rapport présente un maximum, de l'ordre de 16%, vers 8 000K, *i.e.* à l'intérieur de l'atmosphère. Cette correction de pression turbulente n'est supérieure à 1% que dans une zone localisée sous la surface où les températures se situent dans l'intervalle 6 500K, 15 500K ; ce qui correspond au début de la zone d'ionisation de l'hydrogène.

En tenant compte de la pression turbulente, le problème devient totalement implicite ; en effet, lorsque la pression totale, la température et la composition chimique sont connues, le calcul des grandeurs thermodynamiques nécessite la connaissance de la vitesse de convection, déduite elle même des grandeurs thermodynamiques calculées à partir de la pression gazeuse, donc de la pression totale et de la pression turbulente. Dans CESAM2k ce problème implicite est résolu en considérant la pression totale et la pression gazeuse comme deux variables dépendantes distinctes.

La mise en place de la pression turbulente a bénéficié de la collaboration de S. Brun.

PB La correction empirique $\frac{d \ln P_{\text{gaz}}}{d \ln P}$ affectant le gradient adiabatique rend le problème extrêmement instable. Dans CESAM2k cette correction est ignorée si la donnée CPTURB de la NAMELIST NL_CONV est positive ou nulle.

PB CESAM2k permet d'abandonner (*respt.* de réintroduire) la pression turbulente en cours et à *fortiori* au début d'une évolution, le nombre d'équations de structure effectivement résolues est alors 7, (*respt.* 8).

7.10.5 Localisation des limites des zones convectives

Hors atmosphère, cette localisation a pour but essentiel de fixer les limites de la zone à mélanger *i.e.* par mélange convectif, afin de localiser avec précision les discontinuités qui en résultent. Dans l'atmosphère, cette localisation est sans objet puisque la composition chimique et la vitesse angulaires y sont supposées constantes.

Les limites des zones convectives sont déterminées dans la routine `lim_zc`, *cf.* § 8.70 (Page 202). Dans un premier temps on localise en indice de couche, à l'aide du sous-programme `dgrad`, *cf.* § 8.28 (Page 182), les intervalles où se produisent un changement de signe de la quantité $\nabla_{\text{rad}} - \nabla_{\text{ad}}^*$. Dans une seconde étape, la localisation de chaque changement de signe est affinée, par dichotomie, jusqu'à la précision définie par la variable `loc_zc`. Celle-ci ayant été initialisée dans la routine `cesam`, *cf.* § 8.10 (Page 173), suivant le type de précision requis. Ainsi, avec la précision `sa` *cf.* Table 3.1 (Page 32), la limite de la zone convective solaire sera déterminée à mieux que `loc_zc=10-5` de la largeur de l'intervalle la contenant *e.g.* à mieux que ~ 30 m. C'est cette localisation améliorée qui sert de référence pour d'éventuelles extensions.

7.10.6 Extension des zones convectives

L'extension des zones convectives (`jpz=.FALSE.` dans la NAMELIST NL_CONV) et la pénétration convective (`jpz=.TRUE.`) ont pour effet d'étendre la zone de mélange et de fixer le gradient de température égal au gradient adiabatique dans les zones d'extension. Dans CESAM2k on distingue *undershooting*²¹ *i.e.* extension de la limite vers le centre, et *overshooting* *i.e.* extension de la limite vers la surface; leur importance est fixée, respectivement, par les paramètres `ovshts` et `ovshti` de la NAMELIST NL_CONV du fichier de données; les zones convectives sont respectivement étendues de `ovshts` $\times H_p$ et de `ovshti` $\times H_p$. En ce qui concerne l'overshoot d'un cœur convectif de rayon R_c , l'extension de la zone mélangée est limitée à $\min(\text{ovshts} \times H_p, R_c \times \text{ovshts})$ si `jpz=.FALSE.`; elle a pour valeur R_c , si `jpz=.TRUE.`. Pour la pénétration convective, obtenue avec `jpz=.TRUE.`, l'extension de la zone

21. Au sens d'Oxford, *undershooting* et *overshooting* sont utilisés de façon incorrecte, ce qui provoque des réactions épidermiques, parfois violentes, de collègues de pure expression anglo-saxonne; en français ils ont le sens imagé d'extension inférieure et supérieure.

mélangée L_p et du gradient adiabatique est déterminée par Zahn (1991) :

$$L_p = \frac{\zeta H_p}{K_p}, \quad K_p = \frac{\partial \ln K}{\partial \ln P},$$

$\zeta \sim 0.5$ = `ovshti`, étant un paramètre ajustable.

Dans une extension le gradient est pris égal au gradient adiabatique lorsque le paramètre d'overshooting est positif, au gradient radiatif avec une valeur négative, *cf.* § 3.9 (Page 37). La composition chimique et la vitesse angulaire sont homogénéisées, *cf.* § 7.10.7 (Page 153)

Les dispositions suivantes sont prévues :

- La pénétration convective n'a d'action que sur les limites inférieures des zones convectives.
- La pénétration convective et les overshoots n'ont d'effet que si la température est supérieure²² à 500 000K ; cette disposition a pour but d'éviter "d'overshooter" les zones convectives superficielles.
- Avec la présente version de `CESAM2k`, on ne peut donc pas étendre une limite zone radiative / zone convective située dans l'atmosphère.

7.10.7 Mélange convectif

Dans les zones mélangées *i.e.* les zones convectives et leurs extensions par pénétration convective, la composition chimique et la vitesse angulaire sont homogénéisées. Avec diffusion on utilise un coefficient de diffusion de mélange turbulent, typiquement, $d_M = 10^{13} \text{ cm s}^{-1}$, ordre de grandeur obtenu comme produit de valeurs typiques de la vitesse de convection et de l'échelle de hauteur de pression. En l'absence de diffusion, le mélange est effectué par moyennes spatiales couplées avec les réactions thermonucléaires *i.e.* par la résolution numérique simultanée du mélange et de l'évolution nucléaire, ce qui constitue un problème intégro-différentiel, *cf.* § 8.120 (Page 227).

Avec diffusion le mélange est effectué à l'aide d'un coefficient de diffusion D_{ZC} . Une estimation de sa valeur est $l_m v_c / 3$. Avec, pour la vitesse de convection v_c , (Kippenhahn & Weigert, 1991, eq. 7.11,20) :

$$v_c = \frac{6acT^3\Gamma}{\kappa\rho^2c_p l_m}, \quad D_{ZC} = \frac{2acT^3\Gamma}{\kappa\rho^2c_p}$$

7.10.8 Retrait d'un coeur convectif

L'étendue d'un coeur convectif varie au cours de l'évolution. Lors d'une extension la composition chimique, donc la densité, sont discontinues sur la limite externe du coeur. Lors d'une contraction, la composition chimique devient continue de la zone de retrait jusqu'au centre ; la dérivée spatiale du poids moléculaire moyen présente une discontinuité à la limite avec la zone radiative adjacente. Une description précise de ce retrait tenant compte des variations de composition chimique dues aux réactions thermonucléaires locales dans les parties devenant radiatives, est très complexe à traiter numériquement à cause des discrétisations spatiale et temporelle inévitables.

22. Pour modifier cette valeur il faut intervenir dans la routine `lim.zc`.

CESAM2*k* suppose que les abondances des éléments chimiques varient linéairement dans la partie radiative correspondant au retrait du coeur. La situation similaire qui se produit lors de l'augmentation de l'abscisse de la base d'une zone convective n'est pas traitée dans la présente version de CESAM2*k*. Les discontinuités fossiles²³ ne reçoivent pas de traitement particulier ; étant ignorées la diffusion numérique est chargée de les faire lentement disparaître. Avec diffusion, le retrait des coeurs convectifs ne reçoivent pas de traitement spécifique avec la présente version de CESAM2*k*.

7.10.9 Semi-convection

Dans les zones où le gradient radiatif vérifie :

$$\nabla_{\text{ad}} < \nabla_{\text{rad}} < \frac{\varphi}{\delta} \nabla_{\mu} \quad (7.106)$$

le milieu est *vibrationnellement* instable, (Kippenhahn & Weigert, 1991, §30.4.2, p.284). Le mélange des éléments chimiques n'y est pas aussi efficace qu'en cas de l'instabilité convective. CESAM2*k* traite cette situation par diffusion, le coefficient de diffusion étant calculé selon les prescriptions de Langer et al. (1983). Dans les parties concernées le gradient est pris égal au gradient radiatif.

PB Sans diffusion, la semi-convection n'est pas prise en compte.

7.10.10 Estimation de la fréquence de Brunt-Väissälä

Dans les fichiers de sortie en ASCII, CESAM2*k* donne la valeur de la quantité :

$$A = \frac{1}{\Gamma_1} \frac{\partial \ln P}{\partial \ln R} - \frac{\partial \ln \rho}{\partial \ln R} \quad (7.107)$$

pour chaque couche du modèle. A permet l'estimation de la fréquence de Brunt-Väissälä ; A , différence de deux dérivées premières, est équivalente à une dérivée seconde. Le calcul de la fréquence de Brunt-Väissälä est donc sensible aux discontinuités de la dérivée première de la composition chimique qui provoquent des oscillations ; on a indiqué au § 6.2 (Page 70), que la prise en compte d'une légère diffusion turbulente permettait d'obtenir des profils quasi théoriques pour la fréquence de Brunt-Väissälä.

Sur une limite zone radiative / zone convective, la composition chimique est une fonction discontinue si on ignore la diffusion microscopique. Avec diffusion, la densité est continue, non dérivable, A se comporte alors comme une fonction δ . Pour décrire finement le comportement de la fréquence de Brunt-Väissälä, dans les fichiers de sortie destinés aux exploitations sismologiques, *cf.* § 8.87 (Page 213), des points sont ajoutés au voisinage de ces discontinuités, *cf.* § 8.40 (Page 188). La dérivée spatiale de la densité étant accessible dans CESAM2*k*, l'équation 7.102 (Page 154) est utilisée pour le calcul de la fréquence de Brunt-Väissälä. Pour que cette relation soit aussi applicable il est nécessaire que l'approximation de la densité soit numériquement satisfaisante ; ce qui n'est réalisé que lorsque la densité est continue aux limites zones

23. Discontinuités de composition chimiques des pas temporels précédents.

radiatives / zones convectives. **Cette formulation n'est donc applicable que si la diffusion des éléments chimiques est prise en compte.** Si tel n'est pas le cas, comme par exemple avec la précision `np` cf. Table 3.1 (Page 32) ou sans diffusion.

Dans ces cas une formulation approchée est utilisée ; elle repose sur une approximation de $\varphi = \frac{\partial \ln \rho}{\partial \ln \mu}$:

$$\begin{aligned} d \ln \rho &= \alpha d \ln P - \delta d \ln T + \varphi d \ln \mu \implies \\ \frac{\partial \ln \rho}{\partial \ln P} &= \alpha - \delta \frac{\partial \ln T}{\partial \ln P} + \varphi \frac{\partial \ln \mu}{\partial \ln P} = \alpha - \delta \nabla + \varphi \nabla_{\mu} \end{aligned}$$

avec $\nabla_{\mu} = \frac{\partial \ln \mu}{\partial \ln P}$. φ n'est donné ni par les routines ni par les tabulations d'équation d'état. L'approximation "gaz parfait avec radiation" de l'équation d'état (Cox & Giuli, 1968, Eq.3-30, p. 276) :

$$P = \frac{\rho \mathcal{R} T}{\mu} + \frac{a}{3} T^4$$

mène à l'approximation fréquemment utilisée :

$$\frac{\varphi}{\delta} \sim \frac{\beta}{4 - 3\beta}.$$

En tenant compte de :

$$\frac{1}{\Gamma_1} = \alpha - \delta \nabla_{\text{ad}}, \quad H_p = - \frac{\partial R}{\partial \ln P}$$

on obtient la relation utilisée par `CESAM2k` :

$$A = \frac{R\delta}{H_p} \left(\nabla - \nabla_{\text{ad}} + \frac{\beta}{4 - 3\beta} \nabla_{\mu} \right) \quad (7.108)$$

Toutefois la formulation utilisée dans les versions précédentes de `CESAM` reste disponible. Pour la restituer, dans la routine `cesam.f`, cf. § 8.10 (Page 173), du sous-directory `SOURCE`, ligne 225, affecter `.FALSE.` au paramètre `new_bv`. L'expression suivante sera utilisée :

$$A = \frac{R\delta}{H_p} (\nabla_{\text{ad}} - \nabla) - 4\pi R^3 \sum_i \frac{\partial \rho}{\partial X_i} \frac{\partial X_i}{\partial M}. \quad (7.109)$$

Cette formulation présente des inconvénients :

- Elle nécessite les dérivées de la densité par rapport aux différentes espèces chimiques, quantités inaccessibles avec les équations d'état tabulées, si bien que la somme sur les espèces chimiques est nécessairement réduite au seul élément H (*i.e.* $i \equiv 1$). Une conséquence est que Eq. 7.102 (Page 154) conduit à des valeurs systématiquement inférieures à celles données par Eq. 7.104 (Page 155).

- Un autre inconvénient est la non dérivabilité de la densité aux limites zone radiative/zone convective. Comme d’une part, ces limites ne peuvent coïncider *parfaitement* avec les points de grille et que, d’autre part, les équations d’évolution de la composition chimique ne sont pas résolues simultanément avec les équations de la structure, il se produit nécessairement un petit décalage entre la discontinuité de la composition chimique, donc de la densité (ou seulement de leurs gradients), et le point limite où l’égalité des gradients est réalisée. De ces inconsistances, il résulte des oscillations, parfois violentes des valeurs de A obtenues par Eq. 7.102 (Page 154) et, de là, de la fréquence de Brunt-Väissälä.

Toutefois le gradient de composition chimique étant supposé nul dans l’atmosphère restituée, ces difficultés ne s’y présentent pas et Eq. 7.104 (Page 155) est utilisée.

On présente une autre formulation, qui bien qu’implantée dans CESAM2*k*, n’est pas utilisée. Comme Eq. 7.102 (Page 154), elle nécessite une oscularité suffisante des variables. La pression P , le rayon, la masse, étant approchés par une spline en fonction de la variable d’indice q , on a accès à leurs dérivées premières et secondes. L’idée est d’en déduire le gradient de densité puisque :

$$\rho = -\frac{1}{g} \frac{\partial P}{\partial R}, \quad g = \frac{GM}{R^2}.$$

Ainsi on tiendra compte *implicitement* et de façon cohérente, à travers la formulation numérique, des détails de l’équation d’état et des discontinuités. On a :

$$\frac{\partial \ln \rho}{\partial \ln R} = \frac{R}{\rho} \frac{\partial \rho}{\partial R} = \frac{R}{\rho g} \left(\frac{1}{g} \frac{\partial g}{\partial R} \frac{\partial P}{\partial R} - \frac{\partial^2 P}{\partial R^2} \right), \quad (7.110)$$

$$\frac{\partial g}{\partial R} = \frac{G}{R^2} \frac{\partial M}{\partial R} - \frac{2g}{R} = 4\pi\rho G - \frac{2g}{R}. \quad (7.111)$$

A des fins de référence et sous une forme adaptée à la programmation, on donne ci-après les expressions des deux termes de A (Eq. 7.102 (Page 154)) en fonction des quantités ξ , ζ et μ et de leurs dérivées par rapport à la variable d’espacement q .

- Avec les variables $\xi = \ln P$, $\zeta = \left(\frac{R}{R_\odot}\right)^2$ et $\mu = \left(\frac{M}{M_\odot}\right)^{2/3}$:

$$\frac{\partial \zeta}{\partial R} = \frac{2R}{R_\odot^2} = \frac{2\sqrt{\zeta}}{R_\odot}, \quad \frac{\partial}{\partial R} = \frac{\partial q}{\partial \zeta} \frac{\partial \zeta}{\partial R} \frac{\partial}{\partial q} = \frac{2\sqrt{\zeta}}{R_\odot} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q},$$

$$\frac{\partial \ln P}{\partial R} = \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta} \frac{\partial \zeta}{\partial R} = \frac{2\sqrt{\zeta}}{R_\odot} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta},$$

$$\frac{\partial \ln P}{\partial \ln R} = R \frac{\partial \ln P}{\partial R} = 2\zeta \left[\frac{\partial \xi}{\partial q} / \frac{\partial \zeta}{\partial q} \right]. \quad (7.112)$$

$$\frac{\partial P}{\partial R} = P \frac{\partial \ln P}{\partial R} = \frac{2RP}{R_\odot^2} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta}, \quad (7.113)$$

$$\begin{aligned} \frac{\partial}{\partial R} \frac{\partial q}{\partial \zeta} &= \frac{\partial}{\partial R} \left(\frac{\partial \zeta}{\partial q} \right)^{-1} = - \left(\frac{\partial \zeta}{\partial q} \right)^{-2} \frac{\partial}{\partial R} \frac{\partial \zeta}{\partial q} \\ &= - \left(\frac{\partial \zeta}{\partial q} \right)^{-2} \frac{2\sqrt{\zeta}}{R_\odot} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q} \frac{\partial \zeta}{\partial q} = - \frac{2\sqrt{\zeta}}{R_\odot} \left(\frac{\partial \zeta}{\partial q} \right)^{-3} \frac{\partial^2 \zeta}{\partial q^2}, \end{aligned}$$

$$\frac{\partial}{\partial R} \frac{\partial \ln P}{\partial q} = \frac{2\sqrt{\zeta}}{R_{\odot}} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q} \frac{\partial \ln P}{\partial q} = \frac{2\sqrt{\zeta}}{R_{\odot}} \left(\frac{\partial \zeta}{\partial q} \right)^{-1} \frac{\partial^2 \ln P}{\partial q^2},$$

$$\begin{aligned} \frac{\partial^2 P}{\partial R^2} &= \frac{\partial}{\partial R} \frac{\partial P}{\partial R} = \frac{\partial}{\partial R} \left\{ \frac{2RP}{R_{\odot}^2} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta} \right\} \\ &= \frac{1}{R} \frac{\partial P}{\partial R} + \frac{1}{P} \left(\frac{\partial P}{\partial R} \right)^2 + \left[\frac{\partial P}{\partial R} / \frac{\partial \zeta}{\partial q} \right] \frac{\partial}{\partial R} \frac{\partial q}{\partial \zeta} + \left[\frac{\partial P}{\partial R} / \frac{\partial \ln P}{\partial q} \right] \frac{\partial}{\partial R} \frac{\partial \ln P}{\partial q} \\ &= \frac{\partial P}{\partial R} \left\{ \frac{1}{R} + \frac{1}{P} \frac{\partial P}{\partial R} + \frac{2R}{R_{\odot}^2} \left[\frac{\partial^2 \xi}{\partial q^2} / \frac{\partial \xi}{\partial q} - \frac{\partial^2 \zeta}{\partial q^2} / \frac{\partial \zeta}{\partial q} \right] / \frac{\partial \zeta}{\partial q} \right\} \end{aligned} \quad (7.114)$$

L'expression de A donnée par Eq. 7.102 (Page 154) est obtenue par Eq. 7.107 (Page 156) et Eq. 7.105 (Page 156) calculée à l'aide de Eq. 7.106 (Page 156), Eq. 7.108 (Page 156) et Eq. 7.109 (Page 157).

– Avec les variables $\xi = \ln P$, $\zeta = \frac{R}{R_{\odot}}$ et $\mu = \frac{M}{M_{\odot}}$:

$$\frac{\partial \zeta}{\partial R} = \frac{1}{R_{\odot}}, \quad \frac{\partial}{\partial R} = \frac{\partial q}{\partial \zeta} \frac{\partial \zeta}{\partial R} \frac{\partial}{\partial q} = \frac{1}{R_{\odot}} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q},$$

$$\frac{\partial \ln P}{\partial R} = \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta} \frac{\partial \zeta}{\partial R} = \frac{1}{R_{\odot}} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta},$$

$$\frac{\partial \ln P}{\partial \ln R} = R \frac{\partial \ln P}{\partial R} = \frac{R}{R_{\odot}} \left[\frac{\partial \xi}{\partial q} / \frac{\partial \zeta}{\partial q} \right]. \quad (7.115)$$

$$\frac{\partial P}{\partial R} = P \frac{\partial \ln P}{\partial R} = \frac{P}{R_{\odot}} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta}, \quad (7.116)$$

$$\begin{aligned} \frac{\partial}{\partial R} \frac{\partial q}{\partial \zeta} &= \frac{\partial}{\partial R} \left(\frac{\partial \zeta}{\partial q} \right)^{-1} = - \left(\frac{\partial \zeta}{\partial q} \right)^{-2} \frac{\partial}{\partial R} \frac{\partial \zeta}{\partial q} \\ &= - \left(\frac{\partial \zeta}{\partial q} \right)^{-2} \frac{1}{R_{\odot}} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q} \frac{\partial \zeta}{\partial q} = - \frac{1}{R_{\odot}} \left(\frac{\partial \zeta}{\partial q} \right)^{-3} \frac{\partial^2 \zeta}{\partial q^2}, \end{aligned}$$

$$\frac{\partial}{\partial R} \frac{\partial \ln P}{\partial q} = \frac{1}{R_{\odot}} \frac{\partial q}{\partial \zeta} \frac{\partial}{\partial q} \frac{\partial \ln P}{\partial q} = \frac{1}{R_{\odot}} \left(\frac{\partial \zeta}{\partial q} \right)^{-1} \frac{\partial^2 \ln P}{\partial q^2},$$

$$\begin{aligned} \frac{\partial^2 P}{\partial R^2} &= \frac{\partial}{\partial R} \frac{\partial P}{\partial R} = \frac{\partial}{\partial R} \left\{ \frac{P}{R_{\odot}} \frac{\partial \ln P}{\partial q} \frac{\partial q}{\partial \zeta} \right\} \\ &= \frac{1}{P} \left(\frac{\partial P}{\partial R} \right)^2 + \left[\frac{\partial P}{\partial R} / \frac{\partial \zeta}{\partial q} \right] \frac{\partial}{\partial R} \frac{\partial q}{\partial \zeta} + \left[\frac{\partial P}{\partial R} / \frac{\partial \ln P}{\partial q} \right] \frac{\partial}{\partial R} \frac{\partial \ln P}{\partial q} \\ &= \frac{\partial P}{\partial R} \left\{ \frac{1}{P} \frac{\partial P}{\partial R} + \frac{1}{R_{\odot}} \left[\frac{\partial^2 \xi}{\partial q^2} / \frac{\partial \xi}{\partial q} - \frac{\partial^2 \zeta}{\partial q^2} / \frac{\partial \zeta}{\partial q} \right] / \frac{\partial \zeta}{\partial q} \right\}. \end{aligned} \quad (7.117)$$

L'expression de A donnée par Eq. 7.102 (Page 154), est obtenue par Eq. 7.110 (Page 157) et Eq. 7.105 (Page 156) calculée à l'aide de Eq. 7.106 (Page 156), Eq. 7.111 (Page 157) et Eq. 7.112 (Page 158).

PB Ces dernières relations doivent être adaptées si on tient compte de la rotation.

PB Des tests héliosismologiques ont mis en évidence que ces relations, bien que formellement plus cohérentes, faisaient ressortir les incohérences entre équation d'état et opacités. Il convient de ne les utiliser qu'avec prudence.

7.11 Les réactions thermonucléaires

7.11.1 Abondances initiales

Pour éviter des calculs inutiles, il est nécessaire d'optimiser le réseau nucléaire utilisé en fonction de la connaissance, à priori, de la physique des modèles à calculer. Il est par exemple inutile de tenir compte des réactions thermonucléaires concernant le silicium si la température maximum n'atteint pas le seuil du déclenchement de ces réactions. Il est donc inutile de suivre explicitement l'évolution des éléments chimiques qui ne sont pas concernés par le réseau nucléaire retenu, sauf, évidemment, si on désire en suivre la diffusion. Les isotopes de la mixture initiale n'étant pas retenus comme éléments chimiques ils ne doivent pas pour autant être totalement ignorés, d'autant que leur contribution à la masse totale, quelques 10^{-3} , est loin d'être négligeable. Dans CESAM2k, ces éléments sont représentés par l'isotope *fictif* dont la masse (*respt.* charge) égale la moyenne pondérée des masses (*respt.* charges) des éléments dont il n'est pas tenu compte explicitement. Par exemple cet isotope fictif, ici noté Ex, est souvent ^{24}Si avec une mixture initiale solaire. Cet isotope fictif complémentaire a l'indice `i_ex`, cf. § 8.2 (Page 168).

L'abondance initiale en *nombre par mole* de chaque isotope de la mixture utilisée dans CESAM2k est déterminée en fonction du réseau de réactions retenu, cf. § 8.3 (Page 169), à partir :

- Des abondances initiales *en masse* de l'hydrogène X , et de l'hélium Y , lues dans la NAMELIST `NL_CHIM` du fichier de données, cf. § 3.4 (Page 26).
- Des proportions initiales *en nombre* des éléments lourds dans $Z = 1 - X - Y$.
- Des rapports isotopiques initiaux.

De l'ensemble de ces données résulte un système d'équations linéaires dont la solution est la composition chimique initiale. Dans l'exemple qui suit, x_i désigne l'abondance par mole *i.e.* l'abondance par masse divisée par la masse atomique μ_i , de l'isotope d'ordre $i = 1, \dots, 10$ dans la liste : ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , Ex. Les rapports d'abondances initiaux en nombre, dans Z , du carbone, azote et oxygène seront respectivement désignés par C , N et O ; par exemple pour le soleil : $C = 0.24551$, $N = 0.06458$, $O = 0.51295$. Les rapports isotopiques initiaux en nombre seront désignés par le rapport de leurs symboles *e.g.* $^3\text{He}/^4\text{He} = 4.2 \cdot 10^{-4}$. Le système d'équations linéaires vérifiées par les abondances initiales en nombre, x_i , $i=1, \dots, 10$ s'écrit :

$$\begin{aligned} x_1 \nu_1 &= X \\ x_2 \nu_2 + x_3 \nu_3 &= Y \\ \sum_{i=4}^{10} x_i \mu_i &= 1 - X - Y \end{aligned}$$

$$\begin{aligned}
x_4 + x_5 - C \times \sum_{i=4}^{10} x_i &= 0 \\
x_6 + x_7 - N \times \sum_{i=4}^{10} x_i &= 0 \\
x_8 + x_9 - O \times \sum_{i=4}^{10} x_i &= 0 \\
x_2 - x_3 \text{ } ^3\text{He}/\text{}^4\text{He} &= 0 \\
x_5 - x_4 \text{ } ^{13}\text{C}/\text{}^{12}\text{C} &= 0 \\
x_7 - x_6 \text{ } ^{15}\text{N}/\text{}^{14}\text{N} &= 0 \\
x_9 - x_8 \text{ } ^{17}\text{O}/\text{}^{16}\text{O} &= 0
\end{aligned} \tag{7.118}$$

Pour chaque isotope, on fixe une abondance minimale au dessous de laquelle l'isotope est considéré comme inexistant, son évolution temporelle n'est alors pas contrôlée. La plupart des rapports isotopiques utilisés sont ceux de la nébuleuse solaire issus de la table 3 des nuclides de l'article de Anders & Grevesse (1989). Le rapport isotopique $^3\text{He}/^4\text{He} = 1.104 \times 10^{-4}$ est déduit d'une mesure dans Jupiter ; toutefois si ^2H est pris à l'équilibre, il convient d'ajouter à l'abondance initiale de ^3He celle de ^2H qui est transformée en ^3He pendant la pré-séquence principale, d'où la valeur du rapport isotopique initial $^3\text{He}/^4\text{He} = 4.2 \cdot 10^{-4}$.

L'ordre des éléments chimiques est en général celui des masses croissantes, l'isotope fictif venant en dernier, mais ce n'est pas une règle intangible. L'ordre est fixé dans la routine `tabul_nuc`, cf. § 8.127 (Page 233), *e.g.* dans `ppcno3a9` l'ordre des éléments est : ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , Ex. L'isotope fictif venant en dernier.

7.11.2 Cycle PP simplifié

Il s'agit de la routine `nuc_gong`, cf. § 8.98 (Page 218), du Solar Model Comparison Project de Christensen-Dalsgaard (1988). On ne tient compte que d'un seul élément chimique : l'hydrogène dont l'abondance par unité de masse est ici notée X . L'énergie thermonucléaire générée par unité de masse et de temps est :

$$\epsilon = Q_e a_{11} \frac{X^2}{2m_u} \rho T_9^{-\frac{2}{3}} \exp(-bT_9^{-\frac{1}{3}})$$

qui correspond à un taux :

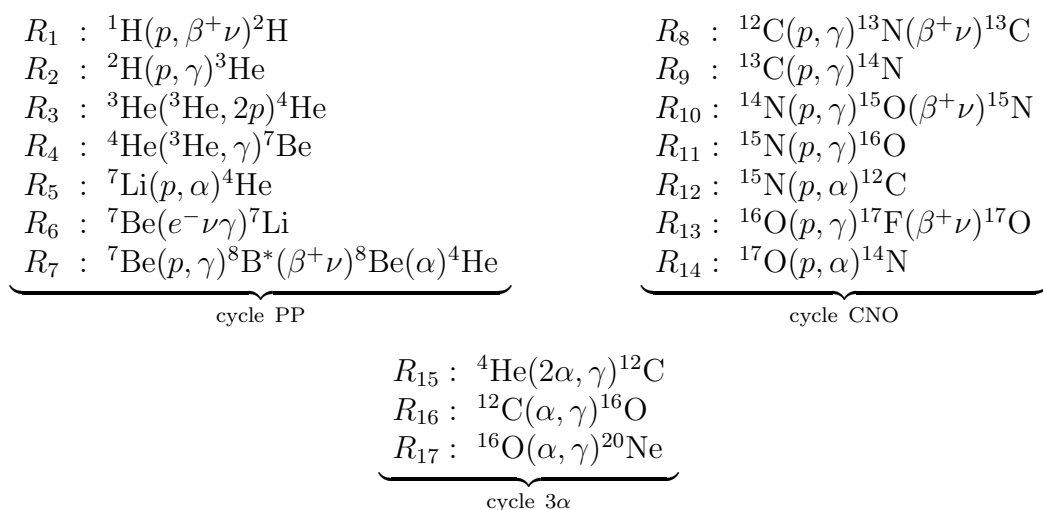
$$\dot{X} = \frac{\partial X}{\partial t} = -R_e a_{11} \frac{X^2}{2} \rho T_9^{-\frac{2}{3}} \exp(-bT_9^{-\frac{1}{3}})$$

où $R_e = 6.5$, $Q_e = 6.5 \times 10^{-5}$, $a_{11} = 4.21 \times 10^{-15}$, $b = 3.6$, m_u est l'unité de masse atomique et $T_9 = T \times 10^{-9}$.

7.11.3 Exemple de réseau nucléaire : cycles PP, CNO et 3α

A titre d'exemple, on détaille la construction d'un réseau nucléaire rassemblant les cycles PP, CNO et 3α

Pour simplifier les notations, on désigne indistinctement par un même symbole l'isotope et son abondance par mole. Les principales réactions thermonucléaires des cycles PP, CNO et 3α sont les suivantes (Clayton, 1968, p. 380,390) :



Compte non tenu de l'effet d'écran, le nombre R_i de réactions i , $i = 1, \dots, 17$, par $\text{sec}^{-1} \text{g}^{-1}$ est obtenu en multipliant le taux tabulé par :

- ρ pour les réactions binaires entre éléments différents,
- $\rho/2!$ pour les réactions binaires entre éléments identiques : R_1 et R_3 ,
- $\rho^2/3!$ pour la réaction ternaire R_{15} , Fowler et al. (1975, p. 86-87).

Le nombre d'électrons par mole e^- et les taux de réaction incluant l'effet d'écran sont des fonctions $R_{i, i=1, \dots, 17}(T, \rho, \mathcal{X})$, \mathcal{X} désignant le vecteur de composition chimique. Ils sont calculés ainsi que leur dérivées, par rapport à (T, ρ, \mathcal{X}) , dans la routine `rq_reac`, en utilisant les tables construites par le programme `tab_reac`.

Bien que les réactions nucléaires ne soient pas uniquement localisées dans un milieu totalement ionisé, *e.g.* le fer n'est pas totalement ionisé au centre du soleil, le nombre d'électrons libres par mole est pris égal à :

$$n_e = \sum_i z_i x_i.$$

7.11.4 Éléments à l'équilibre

Dans certaines phases de l'évolution, des éléments *e.g.* ${}^2\text{H}$, ${}^7\text{Li}$ et ${}^7\text{Be}$ pour le soleil, ont des temps caractéristiques d'évolution très courts par rapport à l'échelle de temps de l'évolution. Leur taux de création est égal à leur taux de destruction. De tels éléments sont dits "à l'équilibre". L'abondance de chaque élément à l'équilibre résulte directement de la relation d'équilibre, ce qui permet des simplifications importantes. ${}^2\text{H}$, ${}^7\text{Li}$ et ${}^7\text{Be}$ sont considérés à l'équilibre dans la routine `ppcno3a9`. On a alors :

$$0 = \frac{d {}^2\text{H}}{dt} = R_1 {}^1\text{H}^2 - R_2 {}^1\text{H} {}^2\text{H} \Rightarrow {}^2\text{H} = \frac{R_1 {}^1\text{H}}{R_2},$$

$$\begin{cases} 0 = \frac{d {}^7\text{Be}}{dt} = R_4 {}^3\text{He} {}^4\text{He} - R_6 {}^7\text{Be} e^- - R_7 {}^1\text{H} {}^7\text{Be} \\ 0 = \frac{d {}^7\text{Li}}{dt} = R_6 {}^7\text{Be} e^- - R_5 {}^1\text{H} {}^7\text{Li} \end{cases} \Rightarrow \begin{cases} {}^7\text{Be} = \frac{R_4 {}^3\text{He} {}^4\text{He}}{R_6 e^- + R_7 {}^1\text{H}} \\ {}^7\text{Li} = \frac{R_6 {}^7\text{Be} e^-}{R_5 {}^1\text{H}} \\ R_4 {}^3\text{He} {}^4\text{He} = R_5 {}^1\text{H} {}^7\text{Li} + R_7 {}^1\text{H} {}^7\text{Be} \end{cases}$$

7.11.5 Effet d'écran

L'effet d'écran est un abaissement de la barrière de potentiel coulombienne dû à la charge d'espace des ions. Il se traduit par l'augmentation des taux des réactions thermonucléaires. Pour une évolution jusque sur la branche des géantes, l'approximation "écran faible" est suffisante, son expression est la suivante (Clayton, 1968, eq. 4.215-221) :

$$f = \exp\left(-\frac{U_0}{kT}\right) = \exp\left(0.188z_1z_2\sqrt{\frac{\rho\zeta}{T_6^3}}\right), \quad \zeta = \sum_i z_i(1+z_i)x_i$$

z_i étant la charge du noyau i , x_i l'abondance, par mole, de l'élément chimique i et $T_6 = T \times 10^{-6}$. ζ représente l'effet des charges. L'approximation "écran faible", n'est valable que si f est proche de l'unité.

CESAM2k offre l'alternative d'utiliser l'écrantage de Mitler (1997), qui généralise, d'après cet auteur, les cas d'écrantage faible et intermédiaire; le cas écrantage fort n'est pas implémenté dans la version actuelle de CESAM2k. Pour son emploi, coder `mitler=.TRUE.` dans la NAMELIST `NLNUC` du fichier de données, cf. § 3.4 (Page 26). L'installation de l'écrantage de Mitler a bénéficié d'une collaboration avec S. Turck-Chièze.

PB Pour la capture électronique de la réaction ${}^7\text{Be}(e^- \nu \gamma){}^7\text{Li}$, l'effet d'écran est inclus dans l'expression du taux de réaction (Fowler et al., 1975).

7.11.6 Energie thermonucléaire et neutrinos

La quantité d'énergie thermonucléaire, $Q_i, i = 1, \dots, 17$, libérée par chaque réaction i , est déterminée pour chaque réaction par le programme `tab_reac` à partir des excès de masse et des énergies des neutrinos, Clayton (1968, p. 289). Ces quantités sont tabulées avec les taux des réactions.

La quantité d'énergie thermonucléaire libérée, est alors : $\sum_i Q_i R_i$ e.g. pour les cycles PP, CNO et 3α , on a :

$$\begin{aligned} \epsilon = & q_1 {}^1\text{H}^2 + q_2 {}^1\text{H} {}^2\text{H} + q_3 {}^3\text{He}^2 + q_4 {}^3\text{He} {}^4\text{He} + q_5 {}^1\text{H} {}^7\text{Li} + (q_6 e^- + q_7 {}^1\text{H}) {}^7\text{Be} \\ & + q_8 {}^1\text{H} {}^{12}\text{C} + q_9 {}^1\text{H} {}^{13}\text{C} + q_{10} {}^1\text{H} {}^{14}\text{N} + (q_{11} + q_{12}) {}^1\text{H} {}^{15}\text{N} + q_{13} {}^1\text{H} {}^{16}\text{O} + q_{14} {}^1\text{H} {}^{17}\text{O} \\ & + q_{15} {}^4\text{He}^3 + q_{16} {}^4\text{He} {}^{12}\text{C} + q_{17} {}^4\text{He} {}^{16}\text{O}, \end{aligned}$$

chacune des lignes représente l'énergie libérée respectivement par les cycles PP, CNO et 3α . Ces quantités apparaissent séparément dans la liste du modèle. Avec ${}^2\text{H}$, ${}^7\text{Li}$ et ${}^7\text{Be}$ à l'équilibre, il convient d'utiliser les valeurs à l'équilibre établies précédemment.

Le nombre de neutrinos, par unité de masse et de temps, émis respectivement par les réactions $R_1, R_6, R_7, R_8, R_{10}$ et R_{13} est :

$$\begin{aligned} H_n = & R_1 {}^1\text{H}^2, \quad B e_n = R_6 e^- {}^7\text{Be}, \quad B_n = R_7 {}^1\text{H} {}^7\text{Li}, \\ N_n = & R_8 {}^1\text{H} {}^{12}\text{C}, \quad O_n = R_{10} {}^1\text{H} {}^{14}\text{N}, \quad F_n = R_{13} {}^1\text{H} {}^{16}\text{O}. \end{aligned}$$

Le nombre de neutrinos reçus sur terre par les expériences du chlore et du galium est calculé en supposant la source distante d'une unité astronomique

7.11.7 Equations d'évolution

Pour l'exemple des cycles PP+CNO+3 α , avec ^2H , ^7Be , et ^7Li à l'équilibre, en désignant les abondances par mole par les symboles isotopiques, les équations d'évolution sont les suivantes :

$$\begin{aligned}
\frac{d^1\text{H}}{dt} &= -3R_1 \text{}^1\text{H}^2 + 2R_3 \text{}^3\text{He}^2 - R_4 \text{}^3\text{He} \text{}^4\text{He} \\
&\quad - R_8 \text{}^1\text{H} \text{}^{12}\text{C} - R_9 \text{}^1\text{H} \text{}^{13}\text{C} - R_{10} \text{}^1\text{H} \text{}^{14}\text{N} - (R_{11} + R_{12}) \text{}^1\text{H} \text{}^{15}\text{N} \\
&\quad - R_{13} \text{}^1\text{H} \text{}^{16}\text{O} - R_{14} \text{}^1\text{H} \text{}^{17}\text{O}, \\
\frac{d^3\text{He}}{dt} &= R_1 \text{}^1\text{H}^2 - 2R_3 \text{}^3\text{He}^2 - R_4 \text{}^3\text{He} \text{}^4\text{He}, \\
\frac{d^4\text{He}}{dt} &= R_3 \text{}^3\text{He}^2 + R_4 \text{}^3\text{He} \text{}^4\text{He} \\
&\quad + R_{12} \text{}^1\text{H} \text{}^{15}\text{N} + R_{14} \text{}^1\text{H} \text{}^{17}\text{O} \\
&\quad - 3R_{15} \text{}^4\text{He}^3 - R_{16} \text{}^4\text{He} \text{}^{12}\text{C} - R_{17} \text{}^4\text{He} \text{}^{16}\text{O}, \\
\frac{d^{12}\text{C}}{dt} &= -R_8 \text{}^1\text{H} \text{}^{12}\text{C} + R_{12} \text{}^1\text{H} \text{}^{15}\text{N} \\
&\quad + R_{15} \text{}^4\text{He}^3 - R_{16} \text{}^4\text{He} \text{}^{12}\text{C}, \\
\frac{d^{13}\text{C}}{dt} &= R_8 \text{}^1\text{H} \text{}^{12}\text{C} - R_9 \text{}^1\text{H} \text{}^{13}\text{C}, \\
\frac{d^{14}\text{N}}{dt} &= R_9 \text{}^1\text{H} \text{}^{13}\text{C} - R_{10} \text{}^1\text{H} \text{}^{14}\text{N} + R_{14} \text{}^1\text{H} \text{}^{17}\text{O}, \\
\frac{d^{15}\text{N}}{dt} &= R_{10} \text{}^1\text{H} \text{}^{14}\text{N} - (R_{11} + R_{12}) \text{}^1\text{H} \text{}^{15}\text{N}, \\
\frac{d^{16}\text{O}}{dt} &= R_{11} \text{}^1\text{H} \text{}^{15}\text{N} - R_{13} \text{}^1\text{H} \text{}^{16}\text{O} \\
&\quad + R_{16} \text{}^4\text{He} \text{}^{12}\text{C} - R_{17} \text{}^4\text{He} \text{}^{16}\text{O}, \\
\frac{d^{17}\text{O}}{dt} &= R_{13} \text{}^1\text{H} \text{}^{16}\text{O} - R_{14} \text{}^1\text{H} \text{}^{17}\text{O}, \\
\frac{d\text{Ex}}{dt} &= -\frac{1}{m_{\text{Ex}}} \left[m_{^1\text{H}} \frac{d^1\text{H}}{dt} + m_{^3\text{He}} \frac{d^3\text{He}}{dt} + m_{^4\text{He}} \frac{d^4\text{He}}{dt} + m_{^{12}\text{C}} \frac{d^{12}\text{C}}{dt} + m_{^{13}\text{C}} \frac{d^{13}\text{C}}{dt} \right. \\
&\quad \left. + m_{^{14}\text{N}} \frac{d^{14}\text{N}}{dt} + m_{^{15}\text{N}} \frac{d^{15}\text{N}}{dt} + m_{^{16}\text{O}} \frac{d^{16}\text{O}}{dt} + m_{^{17}\text{O}} \frac{d^{17}\text{O}}{dt} \right],
\end{aligned} \tag{7.119}$$

la dernière équation exprime la conservation du nombre de baryons, elle permet de tenir compte des éléments qui, comme le ^{20}Ne , n'est brûlé par aucune réaction, mais qui est créé, m_i est le nombre atomique de l'espèce chimique i .

Sans élément à l'équilibre, il faut remplacer les trois premières et la dernière équations par :

$$\begin{aligned}
\frac{d^1\text{H}}{dt} &= -2R_1 \text{}^1\text{H}^2 - R_2 \text{}^1\text{H} \text{}^2\text{H} + 2R_3 \text{}^3\text{He}^2 - R_5 \text{}^1\text{H} \text{}^7\text{Li} - R_7 \text{}^1\text{H} \text{}^7\text{Be} \\
&\quad - R_8 \text{}^1\text{H} \text{}^{12}\text{C} - R_9 \text{}^1\text{H} \text{}^{13}\text{C} - R_{10} \text{}^1\text{H} \text{}^{14}\text{N}
\end{aligned}$$

$$\begin{aligned}
& -(R_{11} + R_{12})^1\text{H } ^{15}\text{N} - R_{13} \text{}^1\text{H } ^{16}\text{O} - R_{14} \text{}^1\text{H } ^{17}\text{O}, \\
\frac{d^2\text{H}}{dt} &= R_1 \text{}^1\text{H } ^2 - R_2 \text{}^1\text{H } ^2\text{H}, \\
\frac{d^3\text{He}}{dt} &= R_2 \text{}^1\text{H } ^2\text{H} - 2R_3 \text{}^3\text{He}^2 - R_4 \text{}^3\text{He } ^4\text{He}, \\
\frac{d^4\text{He}}{dt} &= R_3 \text{}^3\text{He}^2 - R_4 \text{}^3\text{He } ^4\text{He} + 2R_5 \text{}^1\text{H } ^7\text{Li} + 2R_7 \text{}^1\text{H } ^7\text{Be} \\
& + R_{12} \text{}^1\text{H } ^{15}\text{N} + R_{14} \text{}^1\text{H } ^{17}\text{O} \\
& - 3R_{15} \text{}^4\text{He}^3 - R_{16} \text{}^4\text{He } ^{12}\text{C} - R_{17} \text{}^4\text{He } ^{16}\text{O}, \\
\frac{d^7\text{Li}}{dt} &= -R_5 \text{}^1\text{H } ^7\text{Li} + R_6 \text{}^7\text{Be } e^-, \\
\frac{d^7\text{Be}}{dt} &= R_4 \text{}^3\text{He } ^4\text{He} - R_6 \text{}^7\text{Be } e^- - R_7 \text{}^1\text{H } ^7\text{Be}. \\
\frac{d\text{Ex}}{dt} &= -\frac{1}{m_{\text{Ex}}} \left[m_{^1\text{H}} \frac{d^1\text{H}}{dt} + m_{^2\text{H}} \frac{d^2\text{H}}{dt} + m_{^3\text{He}} \frac{d^3\text{He}}{dt} + m_{^4\text{He}} \frac{d^4\text{He}}{dt} \right. \\
& + m_{^7\text{Li}} \frac{d^7\text{Li}}{dt} + m_{^7\text{Be}} \frac{d^7\text{Be}}{dt} + m_{^{12}\text{C}} \frac{d^{12}\text{C}}{dt} + m_{^{13}\text{C}} \frac{d^{13}\text{C}}{dt} \\
& \left. + m_{^{14}\text{N}} \frac{d^{14}\text{N}}{dt} + m_{^{15}\text{N}} \frac{d^{15}\text{N}}{dt} + m_{^{16}\text{O}} \frac{d^{16}\text{O}}{dt} + m_{^{17}\text{O}} \frac{d^{17}\text{O}}{dt} \right].
\end{aligned}$$

On dénommera par "jacobien" la matrice des dérivées partielles des équations d'évolution par rapport aux abondances :

$$J(i, j) \equiv \left(\frac{\partial \frac{dx_i}{dt}}{\partial x_j} \right)_{i,j}.$$

Cette matrice est la base de la résolution des équations implicites du problème différentiel raide, *cf.* §7.5.1 (Page 102), constitué par le réseau nucléaire. Il est nécessaire d'en tenir compte, du fait que l'effet d'écran est fonction des abondances. Une erreur dans l'expression du jacobien entraîne des difficultés de convergence de l'algorithme d'intégration numérique des équations d'évolution. Le programme `test_nuc` du sous-directory `TESTS` peut servir de canevas pour construire un programme permettant de vérifier l'exactitude du jacobien *cf.* §?? (Page ??).

7.12 Equation d'état gaz parfait

CESAM2*k* utilisant une grille adaptative, les dérivées de fonctions thermodynamiques par rapport aux abondances des isotopes sont requises. Elles ne sont pas données par les équations d'état de la littérature. Dans les stades avancés, lorsque les éléments lourds deviennent principalement constitués de magnésium et de silicium les erreurs de dérivées nuisent à la convergence. L'équation d'état `etat_gp`, construite à cet effet, est une équation d'état type "gaz parfait". Elle prend en compte la pression de radiation, les ionisations partielles des isotopes, et la dégénérescence électronique η .

On recherche la densité ρ "gaz parfait" satisfaisant l'équation implicite pression gaz parfait P_{gp} et pression quasi statique P_{qs} :

$$\begin{aligned}
P_{gp}(\rho) - P_{qs} &= 0, \quad P_{gp}(\rho) = P_0 + P_e + P_r \\
P_0 &= \frac{\rho \Re T}{\mu_0}, \quad \frac{\partial P_0}{\partial \rho} = \frac{P_0}{\rho}, \quad \frac{\partial P_0}{\partial T} = \frac{P_0}{T}, \quad \frac{\partial P_0}{\partial \mu_0} = -\frac{P_0}{\mu_0}, \quad \mu_0^{-1} = \sum x_i, \quad \frac{\partial \mu_0}{\partial x_i} = -\mu_0^2 \\
n_e &= \frac{\rho}{m_u \mu_e}, \quad P_e = n_e k T \frac{2 F_{3/2}(\eta)}{3 F_{1/2}(\eta)} = \frac{\rho \Re T}{\mu_e} \frac{2 F_{3/2}(\eta)}{3 F_{1/2}(\eta)}, \\
\frac{\partial P_e}{\partial \rho} &= \frac{P_e}{\rho}, \quad \frac{\partial P_e}{\partial T} = \frac{P_e}{T}, \quad \frac{\partial P_e}{\partial \mu_e} = -\frac{P_e}{\mu_e}, \quad \mu_e^{-1} = \sum \bar{z}_i x_i, \quad \frac{\partial \mu_e}{\partial x_i} = -\mu_e^2 \bar{z}_i \\
\bar{z}_i &= \sum_{j=1}^{Z_i} j x_i, \quad \sum_{j=1}^{Z_i} x_i = 1, \quad P_r = \frac{a}{3} T^4, \quad \frac{\partial P_r}{\partial T} = \frac{4}{3} a T^3 \\
\mu_0^{-1} &= \sum x_i, \quad \mu_e^{-1} = \sum \bar{z}_i x_i, \quad \mu^{-1} = \sum (1 + \bar{z}_i) x_i \\
\mu^{-1} &= \mu_0^{-1} + \mu_e^{-1}, \quad \mu = \frac{\mu_0 \mu_e}{\mu_0 + \mu_e} \\
\frac{\partial \mu_0}{\partial \mu} &= \frac{\mu_0(\mu_0 + \mu_e) - \mu_0 \mu_e}{(\mu_0 + \mu_e)^2} = \frac{\mu_0^2}{(\mu_0 + \mu_e)^2} = \left(\frac{\mu}{\mu_e} \right)^2, \quad \frac{\partial \mu_e}{\partial \mu} = \left(\frac{\mu}{\mu_0} \right)^2 \\
\frac{\partial \mu}{\partial x_i} &= \frac{\partial \mu}{\partial \mu_0} \frac{\partial \mu_0}{\partial x_i} + \frac{\partial \mu}{\partial \mu_e} \frac{\partial \mu_e}{\partial x_i} = - \left(\frac{\mu_0 \mu_e}{\mu} \right)^2 (1 + \bar{z}_i)
\end{aligned}$$

$F_{1/2}$ et $F_{3/2}$ étant respectivement les fonction de Fermi-Dirac d'ordres 1/2 et 3/2; on obtient :

$$\frac{\partial P_{gp}}{\partial \rho} = \Re T \left(\frac{1}{\mu_0} + \frac{1}{\mu_e} \frac{2 F_{3/2}}{3 F_{1/2}} \right).$$

A partir d'une valeur initiale ρ^0 la densité est obtenu itérativement par la relation (Newton-Raphson) :

$$\rho^{n+1} = \rho^n - (P_{gp}^n - P_{qs}) / \frac{\partial P_{gp}}{\partial \rho}.$$

CESAM2k utilise la pression et non la densité comme variable d'état. La formulation "gaz parfait" utilise la densité. On rappelle les formules de transformation de $P(\rho, T, \mu)$ en $\rho(P, T, \mu)$

$$\begin{aligned}
dP &= \left(\frac{\partial P}{\partial \rho} \right)_{T, \mu} d\rho + \left(\frac{\partial P}{\partial T} \right)_{\rho, \mu} dT + \left(\frac{\partial P}{\partial \mu} \right)_{\rho, T} d\mu \\
d\rho &= \left(\frac{\partial \rho}{\partial P} \right)_{T, \mu} dP + \left(\frac{\partial \rho}{\partial T} \right)_{P, \mu} dT + \left(\frac{\partial \rho}{\partial \mu} \right)_{P, T} d\mu \\
d\rho &= \left(\frac{\partial \rho}{\partial P} \right)_{T, \mu} \left[\left(\frac{\partial P}{\partial \rho} \right)_{T, \mu} d\rho + \left(\frac{\partial P}{\partial T} \right)_{\rho, \mu} dT + \left(\frac{\partial P}{\partial \mu} \right)_{\rho, T} d\mu \right] + \left(\frac{\partial \rho}{\partial T} \right)_{P, \mu} dT + \left(\frac{\partial \rho}{\partial \mu} \right)_{P, T} d\mu \\
1 &= \left(\frac{\partial \rho}{\partial P} \right)_{T, \mu} \left(\frac{\partial P}{\partial \rho} \right)_{T, \mu}
\end{aligned}$$

$$d\rho = d\rho + \left[\left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial T} \right)_{\rho,\mu} + \left(\frac{\partial \rho}{\partial T} \right)_{P,\mu} \right] dT + \left[\left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial \mu} \right)_{\rho,T} + \left(\frac{\partial \rho}{\partial \mu} \right)_{P,T} \right] d\mu$$

$$0 = \left[\left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial T} \right)_{\rho,\mu} + \left(\frac{\partial \rho}{\partial T} \right)_{P,\mu} \right] dT + \left[\left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial \mu} \right)_{\rho,T} + \left(\frac{\partial \rho}{\partial \mu} \right)_{P,T} \right] d\mu$$

équation devant être vérifiée $\forall dT, \forall d\mu$, d'où :

$$\left(\frac{\partial \rho}{\partial T} \right)_{P,\mu} = - \left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial T} \right)_{\rho,\mu}$$

$$\left(\frac{\partial \rho}{\partial \mu} \right)_{P,T} = - \left(\frac{\partial \rho}{\partial P} \right)_{T,\mu} \left(\frac{\partial P}{\partial \mu} \right)_{\rho,T}$$

d'où les dérivées de ρ :

$$\left(\frac{\partial \rho}{\partial P} \right)_{T,x_i} = \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1}$$

$$\left(\frac{\partial \rho}{\partial T} \right)_{P,x_i} = - \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1} \left[\left(\frac{\partial P_0}{\partial T} \right)_{\rho,x_i} + \left(\frac{\partial P_e}{\partial T} \right)_{\rho,x_i} + \left(\frac{\partial P_r}{\partial T} \right)_{\rho,x_i} \right] = - \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{\rho,x_i}^{-1} \frac{P_0 + P_e + 4P_r}{T}$$

$$\left(\frac{\partial \rho}{\partial \mu} \right)_{P,T} = - \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1} \left[\left(\frac{\partial P_0}{\partial \mu_0} \right)_{\rho,T} \frac{\partial \mu_0}{\partial \mu} + \left(\frac{\partial P_e}{\partial \mu_e} \right)_{\rho,T} \frac{\partial \mu_e}{\partial \mu} \right] = \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1} \frac{P_0 \mu_0 + P_e \mu_e}{(\mu_0 + \mu_e)^2}$$

$$\left(\frac{\partial \rho}{\partial x_i} \right)_{P,T} = \left(\frac{\partial \rho}{\partial \mu} \right)_{P,T} \frac{\partial \mu}{\partial x_i} = - \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1} \frac{P_0 \mu_0 + P_e \mu_e}{(\mu_0 + \mu_e)^2} \left(\frac{\mu_0 \mu_e}{\mu} \right)^2 (1 + \bar{z}_i) =$$

$$\left(\frac{\partial \rho}{\partial x_i} \right)_{P,T} = - \left(\frac{\partial P_{gp}}{\partial \rho} \right)_{T,x_i}^{-1} (P_0 \mu_0 + P_e \mu_e) (1 + \bar{z}_i)$$

Pour l'énergie interne, $U = 3/2P$ par volume, $u = U/\rho$ par unité de masse, avec U_i et u_i pour les énergies d'ionisation :

$$U = U_0 + U_e + U_r + U_i, \quad u = \frac{3}{2} \mathfrak{R}T \left[\frac{1}{\mu_0} + \frac{1}{\mu_e} \frac{2 F_{3/2}(\eta)}{3 F_{1/2}(\eta)} \right] + \frac{aT^4}{\rho} + u_i$$

Chapitre 8

Les routines de physique

Les grands diseurs ne sont pas les grands faiseurs.

Proverbe populaire.

La mise en œuvre de l'ensemble de ces routines a bénéficié de nombreuses collaborations avec, en particulier, J.Provost, G.Berthomieu et B.Pichon.

On fait d'abord mention des règles de programmation utilisées dans `CESAM2k`, puis on décrit l'aspect programmation des routines de physique regroupées dans les modules du sous-directory `SOURCE`. La plupart d'entre elles ne constituent que des applications au problème de la structure interne des méthodes numériques développées au chapitre 7 (Page 77). La description des routines purement numériques fait l'objet du chapitre 9 (Page 241).

Dans le but de faciliter l'exploitation de cette notice, les routines sont décrites en suivant l'ordre alphabétique de leur dénomination.

8.1 Routines génériques

La physique retenue est introduite par l'intermédiaire des mots clefs du fichier de données `mon_modele.don`, cf. § 3.4 (Page 26). A chaque mot clef est associée une **routine générique**, à laquelle sera affectée la **routine dépendante** où sera effectué le calcul. Par exemple, dans le fichier de données du § 3.4 (Page 26) `ppcno9` est affecté au mot clef `nom_nuc`. Ce mot clef associe à la routine générique `nuc` la fonction de la routine `ppcno9`. Ainsi, à chaque appel de `CALL nuc(...)`, le calcul des réactions thermonucléaires sera effectué par `ppcno9`, cf. § 8.112 (Page 222).

8.2 Unités

Les arguments des routines sont en **unité cgs sauf** : la masse, la luminosité et le rayon qui sont en unités solaires *i.e.* M_{\odot} , R_{\odot} et L_{\odot} . Si, dans un sous-programme,

l'évaluation de certaines variables de sortie fait intervenir des variables d'entrée dont les unités sont dans des systèmes différents, la cohérence entre les unités des variables d'entrée et de sortie est respectée. Par exemple, dans le sous-programme `thermo`, cf. § 8.131 (Page 236), qui calcule diverses grandeurs et leurs dérivées premières à partir des variables thermodynamiques, la variable `gradient` ($\nabla = \partial \ln T / \partial \ln P$) est sans dimension. Le calcul de la variable `dgradm` ($= (\partial \nabla / \partial m)$) nécessite la variable d'entrée `m` (masse) qui est sans dimension : $[m/M_{\odot}]$ ainsi, `dgradm` est sans dimension et non pas en g^{-1} .

Les noms des variables sont choisis de façon à être facilement identifiables. La composition chimique est un argument d'entrée de divers sous-programmes. Les proportions par mole – proportion par gramme / masse atomique – des divers éléments sont les coordonnées d'un vecteur très souvent appelé : `xchim` où les isotopes *effectivement utilisés* sont souvent rangés dans l'ordre des masses atomiques croissantes, ordre qui est défini dans le sous-programme `tabul_nuc`, cf. § 8.127 (Page 233), lors de la création des tables de réactions nucléaires, cf. § ?? (Page ??). Cet ordre n'est pas impératif, sauf en ce qui concerne ^1H qui doit être le premier isotope.

Des valeurs spécifiques sont utilisées pour :

- Indice de ^1H : 1.
- Indice de ^4He : `ihe4`
- Indice de ^3He : `ihe4-1`
- Indice de ^{56}Fe : `ife56`.
- Indice de l'élément fictif complémentaire : `i_ex`, cf. § 7.11.1 (Page 158).

Les variables sont toutes de *type déclaré*, sauf pour certains programmes de source externe, exemple : `z14xcotrin21` cf. § 8.137 (Page 240). La plupart des constantes physiques fondamentales utilisées par `CESAM2k` sont réunies dans le module `mod_donnees`, cf. § E.3 (Page 278). Elles sont initialisées au début de chaque exécution par un appel au sous-programme `ini_ctes`, cf. § 8.60 (Page 198), au début du sous-programme `cesam`, cf. § 8.10 (Page 173). Les constantes propres à chaque routine sont initialisées ou, si possible, recalculées à l'aide des constantes fondamentales au premier appel de cette routine, elles y sont conservées par l'utilisation de l'instruction `SAVE`.

Des commentaires et références permettent de suivre et de vérifier les algorithmes. Dans chaque sous-programme les quantités entrantes et sortantes de la liste d'arguments sont répertoriées, parfois un peu succinctement. Dans un sous-programme, les variables et les routines externes au module auquel appartient le sous-programme sont introduites par l'instruction :

`USE nom_module ONLY : routine, variable`

Cette contrainte permet l'identification des `MODULES` d'appartenance des quantités invoquées.

Le nombre de couches, les limites zones radiatives / zones convectives variant en fonction du temps, il est fait grand usage de tableaux de dimensions variables qui sont alloués/déalloués en fonction des besoins.

Les notations utilisées pour les variables sont aussi explicites que possible, compte tenu d'une nécessaire concision *e.g.* on utilise `gradad` pour le gradient adiabatique ∇_{ad} ; de même pour les dérivées *e.g.* les dérivées par rapport à $P_{\text{gaz}} \equiv p$ ont un

préfixe "d" et une extension¹ "p", exemple : $dgradadp \equiv \frac{\partial \nabla_{ad}}{\partial P_{gaz}}$.

8.3 Routine abon_ini

La routine `abon_ini` est une routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292). Sa fonction principale est d'effectuer l'initialisation des abondances, des rapports isotopiques et des charges des éléments chimiques utilisés par le modèle. A la suite de cette initialisation dans la routine `tabul_nuc`, cf. Fig. 8.3 (Page 207), les éléments chimiques utilisés par les réactions thermonucléaires seront seuls retenus. La source des données est principalement le site d'OPAL : `ftp://www-phys.llnl.gov/pub/opal`.

Description :

- Les charges, masses atomiques et les symboles des éléments chimiques jusqu'au nickel ($Z = 28$) sont initialisés puis un certain nombre de rapports isotopiques².
- Suivant la mixture initiale retenue, cf. § 3.4 (Page 26), les abondances en DeX (H=12) ou en nombre, sont initialisées puis, si besoin, transformées en abondances par nombre. Dans les cas où il y a utilisation d'une mixture initiale personnalisée, celle-ci est lue, en nombre, soit dans le fichier `mon_modele.mix`, soit dans le fichier `mixture` à défaut du précédent cf. § 4 (Page 53).
- Les abondances relatives des métaux dans Z sont ensuite évaluées.

Appel : `abon_ini` est appelée par la routine de calcul des réactions nucléaires lors de son initialisation, cf. § 8.77 (Page 206). `abon_ini` n'a **pas d'argument**

8.4 Routine add_ascii

Dans cette routine PRIVATE du module `mod_cesam`, cf. § E.13 (Page 295), on complète le tableau des quantités entrant éventuellement dans les fichiers ASCII de type `output`, cf. § 8.87 (Page 213).

Description :

En utilisant les valeurs du fichier des oscillations adiabatiques, il y a tabulation, en fonction du rayon, de X pour calcul de la dérivée $\frac{\partial X}{\partial r}$, ainsi que de $F_{rad} = f(r)$ pour celui de $\frac{\partial F_{rad}}{\partial r} = B - J$ et de $\int_0^r \frac{3}{4} \kappa \rho dr = f_{edd} J(r)$. L'ordre des interpolations $m_x \geq 2$ fixé dans la routine peut être adapté (ne pas dépasser 4 *i.e.* splines naturelles). Le fichier d'oscillations adiabatiques est ensuite augmenté, soit pour le calcul des inversions (`ivar=25`) soit pour celui des oscillations non adiabatiques (`ivar=44`). Les dérivées partielles du second ordre sont déduites numériquement en utilisant la routine `df_rotx` décrite au § 8.27 (Page 181). L'identification des quantités est décrite au § D.1 (Page 267).

Appel : `add_ascii` est appelée par `cesam`.

```
SUBROUTINE add_ascii(var, glob, itot, ivar)
```

1. Cette règle est souvent transgressée pour raison de concision.
2. Ces initialisations seraient plus justifiées dans la routine `ctes`.

- Entrées :
 - `glob` : Tableau des quantités globales.
 - `itot` : Nombre total de points.
 - `ivar` : Nombre de variables.
- Entrées/Sorties :
 - `var` : Tableau des variables.

8.5 Routine `alecian1`

La routine interne `alecian1` est une routine PRIVATE du module `mod_evol`, cf. §E.11 (Page 294), sa fonction est le calcul des accélérations radiatives suivant le premier formalisme de G.Alécian. `alecian1` utilise les tables : `phi_psi2_100.dat` ... `phi_psi2_200.dat` du sous-directory `SUN.STAR.DATA`. Chaque fichier correspond à une valeur de la masse clairement identifiée. Cette routine est le regroupement et l'adaptation à CESAM2*k* des données calculées par les sous-programmes `modul_ppxi` et `modul_grad` de G.Alécian.

Description :

- Lors du premier appel : lecture des options, des données et préparation de tableaux nécessaires au calcul des accélérations. Les éléments chimiques utilisés sont identifiés et les isotopes sont regroupés. L'accélération radiative de chaque élément non identifié peut, soit être mise à 0, soit être fixée égale à $-g_e$ (gravité), l'élément ne subit alors aucune force externe. Pour chaque élément identifié, bien que le calcul de l'accélération radiative puisse être effectué, l'une ou l'autre des possibilités décrites précédemment peut être éventuellement utilisée. Pour ce faire, il convient de suivre, *avant compilation*, les intructions données en commentaires.

PB Dans la présente version de CESAM2*k*, les seuls éléments identifiables sont : C, N, O, Ne, Na, Mg, Al, Si, S, Ar, Ca, et Fe. Il n'est pas possible de calculer des accélérations radiatives pour H, He, Li, Be par exemple.

- Puis, pour chaque appel : calcul des accélérations radiatives et calcul de la gravité effective sur chaque ion.

Appel : `alecian1` est appelée par la routine générique `f_rad`, cf. §8.55 (Page 196).

SUBROUTINE `alecian1(lum,ray,t,kap,dkapx,nel,ychim,ioni,grav,g_rad,dg_rad)`

- Entrées :
 - `lum, ray, t, kap, dkapx` : luminosité, rayon, température, opacité, dérivée/X (mole).
 - `nel, ychim, ioni, grav` : nombre d'électrons libres par volume, composition chimique par mole, taux d'ionisation, gravité.
- Entrées/Sorties : `g_rad(i)` : vecteur des accélérations radiatives, la gravité est `grav+g_rad` sur l'élément d'indice `i`
- Sorties : `g_rad(i,j)` : matrice des dérivées des accélérations radiatives sur l'élément `i` / abondance par mole de l'élément `j`.

8.6 Routine ascii

Cette routine PRIVATE du module `mod_cesam`, cf. §E.13 (Page 295), permet la création d'un fichier de sortie ASCII personnalisé. Pour ce faire, coder `nom_output='ascii'` dans le fichier de données `mon_modele.don`. Les quantités à écrire, leur ordre et la structure souhaitée sont définis dans un fichier de nom `sortie_ascii` qui doit se trouver dans l'environnement, cf. §4.8 (Page 58). Les lignes d'identification sont adaptées aux Sorties effectuées. Le fichier ASCII ainsi créé aura l'extension `-ascii`, Exemple : `mon_modele-ascii` dont un exemple se trouve dans le sous-directory `EXPLOIT`.

Diverses possibilités sont offertes :

- Écriture de la fraction de masse en réel ou en DeX cf. §D.1 (Page 267).
- Tabulation du centre vers la surface ou l'inverse.

Les variables et quantités globales à écrire sont **nécessairement** à prendre parmi celles des tableaux `glob` et `var` calculés par `cesam` cf. §B.1 (Page 261) et §B.2 (Page 261).

Description :

- Mise en correspondance des tableaux des quantités globales et des tableaux des variables créés par `cesam` et à écrire.
- Transformation éventuelle des masses en DeX \Rightarrow fraction de masse.
- Inversion éventuelle de l'ordre d'écriture des variables.
- Écriture des en-têtes.
- Écriture des variables.

Appel : Le sous-programme `ascii` est appelé par la routine générique `output`, cf. §8.87 (Page 213).

```
SUBROUTINE ascii(nglob,tglob,totvar,tvar,var,glob,itot,
1 logm,reverse,titre_ascii)
```

- Entrées :
 - `nglob` : Nombre de quantités globales à écrire.
 - `tglob` : Tableau des indices des quantités globales à écrire.
 - `totvar` : Nombre de variables à écrire.
 - `tvar` : Tableau des indices des variables à écrire.
 - `var` : Tableau des valeurs des variables créé par `cesam`.
 - `glob` : Tableau des valeurs des quantités globales créé par `cesam`.
 - `itot` : Nombre de points du tableau `var`.
 - `logm=.TRUE.` : Tabulation de la masse en DeX, en fraction de masse sinon.
 - `reverse=.TRUE.` : Tabulation du centre vers la surface, de la surface au centre sinon.
 - `titre_ascii` : Nom à attribuer au fichier ainsi créé.

8.7 Routine générique atm

Cette routine PUBLIC du module `mod_atm`, cf. §E.8 (Page 291), gère la restitution de l'atmosphère. Elle fait appel à différentes routines PRIVATE, suivant le type d'atmosphère requis défini par la variable `nom_atm` du fichier de données.

Dans ces routines y sont définies les conditions limites externes du problème différentiel de la structure interne; ces conditions limites doivent porter sur deux des trois variables thermodynamiques P , T et ρ . Elles se trouvent localisées dans la région optiquement mince de l'atmosphère. Les équations de la structure interne supposent que le flux d'énergie radiative est transporté par diffusivité radiative *i.e.* approximation de diffusion, ce qui est inexact dans le milieu optiquement mince où sont définies les conditions limites externes. Il faut donc transporter ces limites du milieu mince dans le milieu épais. La source de CESAM2*k* offre la possibilité d'utiliser deux types de méthodes : l'approximation monocouche et la reconstitution de l'atmosphère. Voir, pour plus de détails, Morel et al. (1994).

Appel : atm est appelée par les routines `cesam`, `static.m` et `static.r`.

```
SUBROUTINE atm(list,l_rac,r_rac,xchim,pt_rac,dptsdl,dptsdr,
1 t_rac,dtsdl,dtsdr,m_rac,dmsdl,dmsdr,p_rac,dpsdl,dpsdr,t_eff)
```

- Entrées :
 - `list=.TRUE.` : calcul réduit pour une liste (sans les dérivées),
 - `l_rac` : luminosité au raccord,
 - `r_rac` : rayon au raccord,
 - `xchim` : composition chimique par gramme.
- Sorties :
 - `pt_rac` : pression totale au raccord,
 - `dptsdl` : dérivée / L de la pression totale au raccord,
 - `dptsdr` : dérivée / R de la pression totale au raccord,
 - `t_rac` : température au raccord,
 - `dtsdl` : dérivée / L de la température au raccord,
 - `dtsdr` : dérivée / R de la température au raccord,
 - `m_rac` : masse au raccord,
 - `dmsdl` : dérivée / L de la masse au raccord,
 - `dmsdr` : dérivée / R de la masse au raccord,
 - `p_rac` : pression gazeuse au raccord,
 - `dpsdl` : dérivée / L de la pression gazeuse au raccord,
 - `dpsdr` : dérivée / R de la pression gazeuse au raccord,
 - `t_eff` : température effective.

8.8 Routine subordonnée `base_chim`

Cette routine subordonnée de `diffus`, génère le vecteur nodal de la base de B-splines utilisée pour la résolution des équations de la diffusion des éléments chimiques, *cf.* §7.8 (Page 113).

Description :

- Allocation et initialisation du tableau des multiplicités.
- Définition des multiplicités aux limites zone radiative / zone convective.
- Formation du vecteur nodal.

Appel : Le sous-programme `base_chim`, n'a pas d'argument.

8.9 Routine subordinnée base_rotata

Cette routine subordinnée de `diffus` et de `evol`, génère le vecteur nodal de la base de B-splines utilisée pour la résolution des équations de la diffusion du moment cinétique, *cf.* § 7.9 (Page 129).

Description :

- Allocation du tableau des multiplicités.
- Initialisation du tableau des multiplicités selon la présence ou l'absence de limites zone radiative / zone convective.
- Formation du vecteur nodal.

Appel : Le sous-programme `base_rotata`, n'a pas d'argument.

8.10 Routine cesam

Cette routine PUBLIC, sans argument, du module `mod_cesam`, *cf.* § E.13 (Page 295), constitue en fait le programme principal. Elle gère l'ensemble des calculs ; elle est appelée par les programmes `cesam2k` ou `cesam2k.dbg`, *cf.* § 8.11 (Page 175). Son organigramme général est présenté Figure 1.1 (Page 7). Elle est divisée en 5 parties :

- *Menu d'entrée identifiant l'option de CESAM2k à utiliser :*
 - Reprise d'une évolution en cours.
 - Initialisation d'un modèle de séquence principale d'âge zéro homogène.
 - Initialisation d'un modèle de pré-séquence principale homogène.
- Puis lecture du fichier de données `mon_modele.don`, enfin, initialisation des paramètres de calcul.
- *Cas de la reprise d'une évolution en cours.* Bien qu'il y ait séparation des cas, modèle repris en binaire ou en ASCII, le déroulement des calculs est assez similaire :
 - Lecture du modèle à reprendre, soit en binaire, soit en ASCII.
 - Vérification de la cohérence des paramètres utilisés dans le modèle repris et ceux définis par le fichier de données.
 - Initialisations diverses, en particulier les réactions thermonucléaires et détermination de la valeur du premier pas temporel à utiliser.
 - Pour un modèle repris en ASCII, initialisation des développements sur des bases de B-splines des variables principales et de la composition chimique.
- *Cas de l'initialisation d'un modèle de séquence principale d'âge zéro homogène.*
 - Séparation des cas : modèle d'initialisation en binaire, en ASCII.
 - Pour un modèle repris en ASCII, initialisation des développements sur les bases de B-splines des variables principales.
 - Formation de la fonction de répartition.
 - Initialisations diverses, en particulier les réactions thermonucléaires.
 - calcul du modèle de ZAMS homogène par appel à la routine `resout`.
- *Cas de l'initialisation d'un modèle de pré-séquence principale homogène.*
 - Séparation des cas : modèle repris en binaire ou en ASCII.
 - Pour un modèle repris en ASCII, initialisation des développements sur des bases de B-splines des variables principales.

- Formation de la fonction de répartition.
- Initialisations diverses, en particulier les réactions thermonucléaires.
- Lecture du facteur de contraction, et calcul d'un premier modèle initial par appel à la routine `resout`.
- Dialogue de validation du modèle calculé avec, éventuellement, reprise du calcul avec un facteur de contraction amélioré.
- Calcul d'un second modèle initial avec un facteur de contraction augmenté de 10%.
- Après validation, calcul du pas temporel initial.
- *Calcul de l'évolution.*
- Ecriture du modèle en binaire pour les reprises : fichier `mon_modele_B.rep`. En affectant une valeur négative au paramètre `NB_MAX_MODELES` du fichier de données, cf. § 3.7 (Page 33), tous les fichiers binaires seront écrits dans l'environnement. Pour identification le numéro du modèle est indiqué dans le nom du fichier, exemple : `mon_modele0146_B.rep`.
- Mise à jour du fichier `mon_modele.HR` pour dessin du diagramme HR; ce dernier est réinitialisé dans le cas d'une nouvelle évolution.
- Formation du listing `mon_modele.lis`.
- Calcul du modèle au pas temporel suivant par appel à la routine `resout`.
- Gestion du nouveau modèle :
 - Arrêt du calcul, formation du listing, du fichier ASCII s'il est demandé. On trouve au § B.1 (Page 261) et § B.2 (Page 261) la description des tableaux `glob` et `var` créés par `cesam` pour les Sorties ASCII.
 - Poursuite du calcul de l'évolution.

Appel : `cesam`, appelée par le programme principal, cf. § 8.11 (Page 175), *n'a pas d'argument*.

8.11 Programmes `cesam2k`, `cesam2k_dbg`

Ces deux programmes permettent l'exploitation et le debug de `CESAM2k`. Ils ne comportent que l'instruction d'appel à la routine `cesam`, cf. § 8.10 (Page 173). Ils ne sont différenciés que par des options de compilation qui optimisent l'exécution ou permettent le debug. (Les deux programmes `cesamT` et `cesamT_dbg` identiques aux précédents et utilisés pour des tests, ne sont laissés dans la distribution que pour en faciliter la gestion.)

8.12 Routine `chim_gram`

La fonction de cette routine PUBLIC du module `mod_variables` effectue la transformation des valeurs des rapports d'abondances par mole et de leurs dérivées en rapports d'abondances par masse.

Appel : `chim_gram` est appelée de divers endroits, depuis `cesam` particulièrement.

SUBROUTINE `chim_gram(xchim,dxchim)`

- Entrées/Sorties :

- `xchim`, `dxchim` : composition chimique et dérivée par mole \rightarrow par gramme.

8.13 Routine générique `coeff_rota`

Cette routine PRIVATE du module `mod_evol`, cf. §E.11 (Page 294), est la gestion du calcul des coefficients de la diffusion du moment cinétique.

Description : Le calcul est orienté vers la routine définie dans la routine `lit_n1.f`, cf. §8.74 (Page 205), suivant le nom du formalisme à utiliser, cf. §3.4 (Page 26).

Appel : `coeff_rota` est appelée par `eq_diff_rota`, cf. §8.46 (Page 191).

SUBROUTINE `coeff_rota(dt,nu,y,frl,ddf1)`

- Entrées :
 - `dt`, `nu`, `y` : pas temporel, abscisse, tableau des variables de la rotation.
- Sorties :
 - `f1`, `ddf1` : coefficients des équations de la rotation et dérivées par rapport aux variables de la rotation.

8.14 Routines `coeff_rota3/4`

Ces routines ont pour fonction le calcul des coefficients des équations de la diffusion du moment angulaire suivant le formalismes de Talon et al. (1997); Mathis & Zahn (2004) respectivement.

Description :

- Interpolation au point de calcul des variables de l'équilibre quasi-statique et calcul de divers coefficients.
- Interpolation de la composition chimique, appel à l'équation d'état.
- Estimation des variables de la rotation et de l'équilibre quasi-statique au pas temporel précédent.
- Calcul de l'opacité, des taux d'ionisation, de l'énergie nucléaire et calcul de divers coefficients.
- Interpolation au point de calcul des variables de la rotation.
- Calcul des coefficients de diffusion et des pertes de moment cinétique.
- Calcul des coefficients non identiquement nuls.

Appel : `coeff_rota3/4` sont appelées par `coeff_rota` `eq_diff_rota3/4`. Les arguments sont ceux de `coeff_rota`.

8.15 Routine `coll_atm`

La fonction de cette routine PRIVATE du module `mod_atm`, cf. §E.8 (Page 291), est la résolution, dans l'espace des B-splines, du système des équations de la restitution de l'atmosphère Eq. 7.11 (Page 98).

Description :

- Initialisation du vecteur nodal et des limites, initialisations et allocations diverses lors du premier appel.
- Vérification, qu’au point de raccord, l’accélération centrifuge ne dépasse pas la gravité.
- Pour chaque point de collocation et chaque point limite, formation par appels à `eq_atm` des Eq. 7.11 (Page 98) dans l’espace des B-splines.
- Résolution du système et évaluation des corrections de la méthode itérative, retour à `lim_atm` ou poursuite des itérations (boucle infinie).

Appel : `coll_atm` est appelée par la routine `lim_atm` cf. § 8.68 (Page 201).

```
SUBROUTINE coll_atm(r_rac,l_rac,xchim,ord_atm,knot_atm,dim_atm,ms_atm)
```

- Entrées :
 - `r_rac`, `l_rac`, `xchim` : rayon, luminosité composition chimique au raccord avec l’enveloppe.
 - `ord_atm`, `knot_atm`, `dim_atm`, `ms_atm` : ordre des équations différentielles, nombre de points du vecteur nodal, dimension de la base de B-splines, ordre des B-splines.

8.16 Routine `coll_qs`

La fonction de cette routine PRIVATE du module `mod_static`, cf. § E.12 (Page 295), effectue la résolution, dans l’espace des B-splines, du système Eq. ?? (Page ??) des équations de la structure interne (équilibre quasi-statique).

Description :

- Initialisation du vecteur nodal et des limites, initialisations et allocations diverses lors du premier appel.
- Changement de la base de B-splines si le nombre de couches a varié depuis l’appel précédent.
- Pour chaque point de collocation et chaque point limite, formation par appels à `static`, cf. § 8.125 (Page 231), des équations ?? (Page ??) dans l’espace des B-splines.
- Résolution du système et évaluation des corrections et de la précision de la méthode itérative, retour à `resout` ou poursuite des itérations (boucle infinie).

Appel : `coll_qs` est appelée par `resout`, cf. § 8.116 (Page 223).

```
SUBROUTINE coll\_qs(dt,compt,repnd,err,vare,qmax,corr)
```

- Entrées :
 - `dt`, `compt`, `repnd=.TRUE.` : pas temporel, compteur du nombre d’itérations, il faut réinitialiser la solution.
- Sorties :
 - `err`, `vare`, `qmax`, `corr` : erreur maximale, variable présentant cette erreur, indice de la couche où l’erreur est maximale, correction.

8.17 Routine collision

Cette routine PRIVATE du module `mod_evol`, cf. § E.11 (Page 294), calcule les intégrales de collision $\Omega_{ij}^{(kl)}$, $k, l = 1, 2$ en utilisant les tabulations de Paquette et al. (1986). Ces tables ont été scannées de l'ApJS. Les quantités qui figurent dans les `data` sont les valeurs des intégrales de collision calculées par les formules d'interpolation pour les $\psi_{st}(n)$ (notations de Paquette et al. (1986)), et prolongées jusqu'à $\psi_{st} = 4$; au delà de $\psi_{st} = 4$, on utilise les approximations analytiques.

Au premier appel de la routine, on calcule les coefficients d'interpolation (appel à `bsp1dn`, cf. § 9.2.2 (Page 244)) et on effectue quelques autres initialisations. De la longueur de Debye λ , on déduit les valeurs des ψ_{ij} d'où l'interpolation des intégrales de collision et le calcul de leurs dérivées par rapport à chacune des abondances; distinction est faite entre potentiels attractifs et répulsifs. On économise des ressources de temps calcul en exploitant les symétries.

Appel : `collision` est appelée par la routine `diffm_br` cf. § 8.30 (Page 183).

SUBROUTINE `collision(nb, zb, mij, xi, ro, drox, t, omega11, zij, zpij, zsjj,`
`1 domega11, dzij, dzpij, dzsjj)`

- Entrées :
 - `nb, zb, mij, xi` : nombre de particules (1 : indice de ^1H , `nb` : indice pour les électrons), charges, masses réduites, abondances par mole.
 - `ro, drox, t` : densité, dérivée logarithmique par mole, température.
- Sorties :
 - `omega11, zij, zpij, zsjj` : tables des intégrales de collision.
 - `domega11, dzij, dzpij, dzsjj` : tables des dérivées des intégrales de collision.

8.18 Routine générique conv

Cette routine PUBLIC du module `mod_conv`, cf. § E.7 (Page 290), gère le calcul du gradient de température $\nabla \equiv \frac{\partial \ln T}{\partial \ln P}$ dans les zones convectives et de ses dérivées partielles par rapport aux variables locales. Suivant la formulation de la convection définie dans le fichier de données, cf. § 3.4 (Page 26), un appel ciblé dirige le calcul vers la routine *ad-hoc*.

Description :

La structure des routines de convection est la même pour toutes. Seules diffèrent les formulations cf. § 7.10.2 (Page 149). Des grandeurs spécifiques aux conditions physiques du calcul sont initialisées au premier appel. Puis la racine réelle de la cubique est recherchée en utilisant l'algorithme de Newton-Raphson. Enfin le gradient convectif, l'efficacité de la convection et leurs diverses dérivées sont calculés.

Appel : Le sous-programme `conv` est appelé par les routines `thermo`, cf. § 8.131 (Page 236), et `thermo_atm`, cf. § 8.132 (Page 237).

SUBROUTINE `conv(r, krad, gravite, delta, cp, ro, hp, taur, gradrad,`
`1 gradad, der, gradconv, dgradkra, dgradgra, dgradel, dgradcp, dgradro,`

2 dgradhp, dgradtaur, dgradgrad, dgradgad,
 3 gam, dgamkra, dgamgra, dgandel, dgamcp, dgamro,
 4 dgamhp, dgamtaur, dgamgrad, dgamgad)

- Entrées :
 - r, krad, gravite, delta, cp : rayon, conductivité radiative, gravité, δ , c_p .
 - ro ,hp, taur, gradrad, gradad : densité, échelle de hauteur de pression, épaisseur optique de la bulle convective, gradient radiatif, gradient adiabatique – éventuellement corrigé du terme $\frac{d \ln P_{\text{gaz}}}{d \ln P}$, cf. § 7.10.4 (Page 151).
 - der=.TRUE. : le calcul des dérivées est requis.
- Sorties :
 - gradconv, dgradkra, dgradgra, dgradel, dgradcp, dgradro, dgradhp, dgradtaur, dgradrad, dgradad : gradient de température et dérivées.
 - gam, dgamkra, dgamgra, dgandel, dgamcp, dgamro, dgamhp, dgamtaur, dgamgrad, dgamgad : efficacité de la convection et dérivées.

8.19 Routine conv_a0

En un point d'une zone convective, il semble irréaliste que la longueur de mélange puisse être supérieure à la distance qui sépare ce point de la plus proche limite de la zone convective. Ainsi que Eggleton (1972) l'a proposé, avec la routine PRIVATE conv_a0 du module mod_conv, cf. § E.7 (Page 290), la longueur de mélange utilisée devient nulle à chaque limite zone radiative / zone convective. L'introduction de la distance exacte change la nature du problème qui devient intégral-différentiel ce qui complique considérablement la résolution numérique. Ce qu'évite la formulation locale d'Eggleton, cf. § 7.10.2 (Page 149).

8.20 Routine conv_cm

Dans la routine PRIVATE conv_cm du module mod_conv, cf. § E.7 (Page 290), le gradient de température dans une zone convective est calculé selon les prescriptions de Canuto & Mazitelli (1991) avec, pour longueur de mélange, $l = \alpha H_p$, cf. § 7.10.2 (Page 149).

8.21 Routines conv_cgm_reza, conv_cm_reza

Dans la routine PRIVATE conv_cm_reza du module mod_conv, cf. § E.7 (Page 290), similaire à conv_cm, il est tenu compte de la quantité thermodynamique $\delta = (\frac{\partial \ln \rho}{\partial \ln T})_P$. Dans la routine conv_cgm_reza, le calcul du gradient convectif est effectué selon les prescriptions de Canuto et al. (1996) avec la prescription de Bernkopf (Heiter et al. (2002)), cf. § 7.10.2 (Page 149).

Ces deux routines ont été mises en œuvre et mises à la disposition des utilisateurs de CESAM2k par Reza Samadi, LESIA, Observatoire de Paris

8.22 Routine conv_jmj

Les spécificités de la routine de convection PRIVATE conv_jmj du module mod_conv, cf. §E.7 (Page 290), ont été décrites au §7.10.2 (Page 149). La mise en œuvre de conv_jmj a bénéficié de la collaboration de J.Provost et de MJ.Goupil.

8.23 Routine coulomb

Cette routine PRIVATE du module mod_evol effectue le calcul du logarithme de Coulomb et de ses dérivées par rapport à la densité et à l'abondance en ^1H ou ^4He , suivant le formalisme simplifié de Michaud & Proffitt (1993, p. 250).

Appel : coulomb est appelée par la routine diffm_mp.

```
SUBROUTINE coulomb(zi,zj,thetae,ro,x,t,lnlambij,
1 lnlambijr,lnlambijx,cij,cijr,cijx)
```

- Entrées :
 - zi, zj, thetae : charges des éléments i et j, thetae=1 pour un mélange hélium/hydrogène ;
 - ro, x, t : densité, abondance par volume de ^1H ou ^4He , température.
- Sorties :
 - lnlambij, lnlambijr, lnlambijx, cij, cijr, cijx : logarithmes de Coulomb et dérivées par rapport à ρ et ^1H ou ^4He .

8.24 Routines ctes_85, ctes_94, ctes_94m

Ces routines PRIVATE du module mod_donnees permettent l'initialisation de la plupart des éléments du module mod_donnees, cf. §E.3 (Page 278), qui regroupe les principales constantes physiques à utiliser.

- ctes_85 : Routine de référence, dont la source principale des données de physique est le CRC Handbook of chemistry and physics 75th edition 1994, D.R. Lide *et al.* CRC press Boca Raton, Ann Arbor, London, Tokyo.
- ctes_94 : Identique à ctes_85 avec des valeurs plus récentes pour quelques constantes.
- ctes_94m : Identique à ctes_94 avec des valeurs entières pour les masses atomiques.

Appel : Suivant la variable nom_ctes du fichier de données cf. §3.4 (Page 26), l'une ou l'autre est appelée par la routine générique ini_ctes cf. §8.60 (Page 198), au début de l'exécution du sous-programme cesam cf. §8.10 (Page 173). Ces routines **n'ont pas d'argument**.

8.25 Routine générique des

Cette routine PRIVATE du module mod_cesam, cf. §E.13 (Page 295), est la *routine générique* de dessin, elle utilise le logiciel PGLOT cf. §A.2 (Page 250).

Au fur et à mesure du calcul, le diagramme HR, la répartition en zone radiative / zone convective, les variables de structure et les abondances en fonction de la masse ou du rayon sont tracés.

Appel : `des` qui n'a que des arguments d'entrée est appelée par `cesam`.

SUBROUTINE `des(fin,dt,teff)`

- `fin=.TRUE.` : dernier appel, fermeture du dessin,
- `dt, teff` : pas temporel, température effective.

8.26 Routines `des_m`, `des_r`

L'une ou l'autre de ces routines PRIVATE du module `mod_cesam`, cf. §E.13 (Page 295), est appelée par la routine générique `des` suivant le type de dessin défini par la variable `nom_des` du fichier de données, cf. §3.4 (Page 26). Le dessin est effectué respectivement en fonction de la masse (`nom_des='des_m'`) ou du rayon (`nom_des='des_r'`), ou encore n'est pas effectué (`nom_des='no_des'`)

Description : Les algorithmes de ces deux routines sont similaires :

- Initialisations diverses relatives au logiciel PGPLOT.
- Tracé des abondances par unité de masse, ^1H et ^4He sont dessinées en valeurs réelles, les abondances des autres éléments sont normalisées sur $[0,1]$, les coefficients de normalisation sont indiqués sur l'écran pour chaque variable.
- Tracé des variables de structure normalisées, pression, température, rayon ou masse et luminosité, les coefficients de normalisation sont indiqués sur l'écran pour chaque variable.
- Tracé du diagramme HR, sur les deux axes, les échelles sont déterminées lors du premier appel en fonction de la valeur de la masse initiale ; si le fichier `mon_modele.TeffL` existe dans l'environnement, tracé de la boîte d'erreur cf. §4.11 (Page 63).
- Tracé des zones convectives, à l'issue de chaque pas temporel, un trait plein indique les intervalles de masse affectés par la convection.

Particularité de `des_r` :

Le rayon d'une étoile change au cours de l'évolution ; il arrive que les graphes de la représentation des variables de structure en fonction du rayon utilisent mal le domaine spatial de l'épure. Lorsque le rayon total devient, soit supérieur aux $9/10$, soit inférieur aux $3/4$, du domaine, on effectue un changement d'échelle qui a pour effet de ramener la surface aux $3/4$ du domaine. Tant que l'échelle en rayon reste fixée, un tiret vertical – aux $3/4$ du domaine – matérialise la valeur antérieure du rayon qui a servi à définir l'échelle. Cette disposition permet de suivre l'évolution du rayon en restant dans le cadre de l'épure.

8.27 Routine `df_rotx`

Cette routine PUBLIC du module `mod_etat`, cf. §E.5 (Page 290), constitue un opérateur qui permet de transformer les dérivées par rapport à P, T, X d'une

fonction thermodynamique $f(P, T, X)$ en celles par rapport à ρ, T, X . En appliquant plusieurs fois l'opérateur on obtient les dérivées d'ordre supérieur.

Appel : `df_rotx` est appelée par `add_ascii`, cf. § 8.4 (Page 169).

```
SUBROUTINE df_rotx(p,ro,t,alfa,delta,dfdpx,dfdt_px,dfdx_pt,
1 drodx_pt,df_dro,df_dt,df_dx)
```

– Entrées :

- `p, ro, t, alfa, delta` : pression, densité, température, $\frac{\partial \ln \rho}{\partial \ln P}$, $-\frac{\partial \ln \rho}{\partial \ln T}$,
- `dfdpx, dfdt_px, dfdx_pt, drodx_pt` : $\frac{\partial f}{\partial P}$, $\frac{\partial f}{\partial T}$, $\frac{\partial f}{\partial X}$, $\frac{\partial \rho}{\partial X}$.

– Sorties :

- `df_dro, df_dt, df_dx` : $\frac{\partial f}{\partial \rho}$, $\frac{\partial f}{\partial T}$, $\frac{\partial f}{\partial X}$

8.28 Fonction dgrad

Cette fonction PUBLIC du module `mod_static`, cf. § E.12 (Page 295), forme le critère de convection de Schwarzschild ou de Ledoux suivant les prescriptions décrites au § 7.10.1 (Page 148).

Description :

- Après quelques initialisations, appel à l'équation d'état et calcul de l'opacité
- Calcul du gradient radiatif avec le cas particulier du centre.
- Formation du critère de convection de Schwarzschild ou de Ledoux.

PB Pour une raison inconnue, l'utilisation du critère de Ledoux est délicate.

Appel : `dgrad` est appelée de `lim_zc`, cf. § 8.70 (Page 202).

```
FUNCTION dgrad(pt,p,t,dlpp,xchim,m,l,r,dxchim,w)
```

- `pt, p, t, dlpp, xchim, dxchim` : Pression totale et gazeuse, température, $\frac{\partial \ln P_{\text{gaz}}}{\partial \ln P}$, composition chimique et dérivée;
- `m, l, r, w` : masse, luminosité, rayon, vitesse angulaire.

8.29 Routine générique diffm

Cette routine PRIVATE du module `mod_evol`, cf. § E.11 (Page 294), est la *routine générique* de calcul de la diffusion microscopique. Elle gère les descriptions `diffm_br`, `diffm_mp` et `diffm_0`, cette dernière, formelle, fixe à zéro les coefficients de diffusion microscopique.

Description :

Pour chaque élément du vecteur de composition chimique, l'équation de diffusion, avec les termes nucléaires, est formulée dans la routine `eq_dif_chim`. Les coefficients de diffusion microscopique sont calculés dans les routines de type `diffm`, et des coefficients de diffusion turbulente et de mélange (moment angulaire inclus) sont introduits par l'intermédiaire des routines de type `diffm` cf. § 8.32 (Page 184).

Appel : `diffm` est appelée par `eq_diff_chim`, cf. § 8.44 (Page 190).

SUBROUTINE diffm(p,t,r,l,m,ro,drox,kap,dkapx,w,gradrad,dgradradx,xi,d,dd,v,dv)

- Entrées :
 - p, t, r, lum, ltot, m : pression, température, rayon, luminosité locale et totale, masse.
 - ro, drox, kap, dkapx : densité, dérivée/ X, opacité, dérivée / X,
 - gradrad, dgradradx, xi : gradient radiatif, dérivée / X. composition chimique par mole.
- Sorties :
 - d, dd : matrice des coefficients de diffusion microscopiques d_{ij} de $\frac{\partial x_j}{\partial m}$ et dérivées $\frac{\partial d_{ij}}{\partial x_k}$.
 - v, dv : vecteur des vitesses v_i des x_i et dérivées $\frac{\partial v_i}{\partial x_k}$.

8.30 Routine diffm_br

La routine PRIVATE diffm_br du module mod_evol, cf. § E.11 (Page 294), permet de calculer les coefficients de diffusion microscopique en résolvant les équations de transport selon le formalisme de Burgers, en tenant compte, éventuellement, des accélérations radiatives cf. § 8.5 (Page 170). La formulation du système des équations de Burgers utilisée dans CESAM2k, est en grande partie originale. Elle est décrite § ?? (Page ??).

Description :

- Au premier appel, initialisation de constantes, des masses réduites, allocation diverses, initialisation des accélérations radiatives.
- Calcul des taux d'ionisation, calcul du poids moléculaire moyen, de la gravité corrigée des accélérations radiatives et de leurs dérivées.
- Calcul des intégrales de collision, des coefficients de résistance Eq. 7.45 (Page 120) et de leurs dérivées.
- Calcul des coefficients \bar{q}_{ij} Eq. 7.57 (Page 125), d_i Eq. 7.58 (Page 125), f_{ij} Eq. 7.59 (Page 125) et de leurs dérivées.
- Formation des matrices Δ Eq. 7.54 (Page 125), \mathcal{M} Eq. 7.55 (Page 125), A_w Eq. 7.56 (Page 125), A_r Eq. 7.60 (Page 126), A_c Eq. 7.62 (Page 126), A_e Eq. 7.61 (Page 126).
- Calcul de ω par inversion du système linéaire, Eq. 7.63 (Page 126).
- Formation de la matrice des coefficients de diffusion Eq. 7.66 (Page 127), et du jacobien Eq. 7.68 (Page 127).
- Calcul du vecteur des vitesses de diffusion Eq. 7.64 (Page 126) et du jacobien Eq. 7.67 (Page 127).

8.31 Routine diffm_mp

La routine PRIVATE diffm_mp du module mod_evol, cf. § E.11 (Page 294), utilise la formulation simplifiée de Michaud & Proffitt (1993) pour l'estimation des coefficients de diffusion microscopique. On suppose que les éléments lourds sont *éléments traces*, i.e. un élément autre que ^1H et ^4He ne diffuse que par rapport aux

protons. Ce formalisme, plus simple mais moins général que celui de Burgers, **ne s'applique pas** dans un milieu privé de noyaux d'hydrogène, *i.e.* un cœur d'hélium.

Description :

- Au premier appel, initialisation de constantes, des masses réduites, allocations diverses.
- Calcul des logarithmes de Coulomb pour ^1H et ^4He .
- Calcul des coefficients de diffusion de ^1H et de ^4He et dérivées.
- Calcul des coefficients de diffusion pour les éléments tests par rapport à ^1H et dérivées.

8.32 Routine générique difft

Cette routine PRIVATE du module `mod_evol`, *cf.* §E.11 (Page 294), est la *routine générique* de calcul de la diffusion turbulente.

Appel : `difft` est appelée par `eq_diff_chim`, *cf.* §8.44 (Page 190).

SUBROUTINE `difft(melange,t,ro,drox,kap,dkapx,deff,gradad,gradrad,m,d,dd)`

- Entrées :
 - `melange` : `melange=.TRUE.` on est dans une zone à mélanger,
 - `t`, `ro`, `drox`, `kap`, `dkapx`, `deff` : température, densité, opacité, et dérivées par rapport à l'abondance de ^1H **par gramme**, coefficient de diffusion turbulente due à la rotation,
 - `gradad`, `gradrad`, `m` : gradients adiabatique et radiatifs, abscisse de calcul.
- Sorties :
 - `d`, `dd`, `v`, `dv` : tableaux $d(i,j)$, $dd(i,j,k)$, $v(i)$, $dv(i,k)$, $i,j,k = 1, \dots, n_{\text{elem}}$.

8.33 Routine difft_gab

Cette routine PRIVATE du module `mod_evol`, *cf.* §E.11 (Page 294), effectue le calcul des coefficients de diffusion turbulente. Elle exploite une suggestion de M.Gabriel : pour éviter la sédimentation de l'hélium et des éléments lourds dans la partie supérieure de l'enveloppe, il suffit de mélanger la partie de l'enveloppe jusqu'à la fin de la zone d'ionisation de l'hélium, localisée vers 10^6K .

Description :

- Quelques initialisations sont effectuées au premier appel.
- S'il y a mélange, ou si la température est inférieure à 10^6K la diagonale de la matrice de diffusion est affectée du coefficient de mélange convectif $d_M \gg 1$ *cf.* §7.10.7 (Page 153).
- Sans mélange, on ajoute aux termes diagonaux de la matrice de diffusion, les coefficients de diffusion turbulente `d_turb` *cf.* §3.4 (Page 26), d_{eff} *cf.* §7.9.4 (Page 132).

SUBROUTINE `difft_gab(melange,t,deff,d)`

- Entrées :
 - `melange` : `melange=.TRUE.` on est dans une zone à mélanger,
 - `t,deff` : température, coefficient de diffusion turbulente due à la rotation,
- Sortie :
 - `d` : tableau $d(i, j)$.

8.34 Routine `difft_nu`

Cette routine PRIVATE du module `mod_evol`, cf. § E.11 (Page 294), effectue le calcul des coefficients de diffusion turbulente.

Description :

- Quelques initialisations sont effectuées au premier appel.
- En cas de mélange, la diagonale de la matrice de diffusion est affectée du coefficient de mélange convectif $d_M \gg 1$ cf. § 7.10.7 (Page 153).
- Sans mélange, on ajoute aux termes diagonaux de la matrice de diffusion, les coefficients de diffusion turbulente `d_turb` cf. § 3.4 (Page 26), d_{eff} cf. § 7.9.4 (Page 132) et de viscosité radiative Re_ν , cf. Morel & Thévenin (2002).

Appel : `difft_nu`, est appelée par `difft` :

SUBROUTINE `difft(melange,t,ro,drox,kap,dkapx,deff,d,dd)`

- Entrées :
 - `melange` : `melange=.TRUE.` on est dans une zone à mélanger,
 - `t, ro, drox, kap, dkapx, deff` : température, densité, opacité, et dérivées par rapport à l'abondance de ^1H **par gramme**, coefficient de diffusion turbulente due à la rotation,
- Sorties :
 - `d, dd` : tableaux $d(i, j)$, $dd(i, j, k)$, $i, j, k = 1, \dots, n_{\text{elem}}$.

8.35 Routine `difft_sun`

Cette routine PRIVATE du module `mod_evol`, cf. § E.11 (Page 294), calcule un coefficient de diffusion turbulente, sous la ZC externe solaire, suivant Gabriel (1997).

Appel : `difft_sun` est appelée par `eq.difft`,

SUBROUTINE `difft_sun(melange,deff,gradad,gradrad,m,d)`

- Entrées :
 - `melange` : `melange=.TRUE.` on est dans une zone à mélanger,
 - `deff, gradad, gradrad, m` : coefficient de diffusion turbulente due à la rotation, gradients adiabatique et radiatifs, abscisse de calcul.
- Sorties :
 - `d` : tableau $d(i, j)$ $i, j, \dots, n_{\text{elem}}$.

8.36 Routine diffus

Cette routine PRIVATE du module `mod_evol`, *cf.* §E.11 (Page 294), gère l'intégration numérique des équations de diffusion des espèces chimiques *cf.* §7.8 (Page 113) et du moment cinétique *cf.* §7.9 (Page 129).

Ces équations, du type intégro-différentiel, sont couplées. Pour des raisons d'instabilité numérique, il s'est avéré extrêmement délicat de les résoudre simultanément, force a été de les traiter séquentiellement, au prix d'une convergence médiocre. Leur résolution numérique utilise la méthode des éléments finis de Galerkin, *cf.* §7.7.1 (Page 110). Bien que la structure des algorithmes de résolution utilisés pour chacun des deux systèmes différentiels soient identiques deux routines distinctes ont été créées, pour conserver un maximum de souplesse.

Description :

- Au premier appel, initialisation et diverses écritures.
- Recherche du nombre et des indices des limites zone radiative / zone convective.
- Création du vecteur nodal pour la diffusion des éléments chimiques.
- Création du vecteur nodal pour la diffusion du moment cinétique.
- Résolution des équations de la diffusion du moment cinétique.
- Résolution des équations de la diffusion des éléments chimiques.
- Déallocations diverses.

Appel : `diffus` est appelée par `evol`, *cf.* §8.54 (Page 195).

SUBROUTINE `diffus(ok,dt,mc_tmp,nc_tmp)`

- Entrées :
 - `dt` : pas temporel,
 - `mc_tmp,nc_tmp` : masses temporaires (abscisses langrangiennes), nombre de masses temporaires.
- Sorties :
 - `ok`: `ok=.TRUE.` les algorithmes de résolution de la diffusion du moment cinétique et des espèces chimiques ont convergé au niveau de précision requis.

8.37 Routine générique diffw

Cette routine PRIVATE du module `mod_evol`, *cf.* §E.11 (Page 294), est la *routine générique* de calcul des coefficients de diffusion turbulente horizontale, verticale et effective du moment cinétique, *cf.* §7.9 (Page 129). Dans la présente version de CESAM2k elle ne gère que la description `diffw_mpz`.

Appel : `diffw` est appelée par `eq_diff_rota`, *cf.* §8.46 (Page 191).

Arguments de diffw

SUBROUTINE `diffw(dlnro,grand_k,nu,n2mu,n2t,r,ro,dfrot,frot,1 deff,dh,dv)`

- Entrées :

- `dlnro, grand_k, nu, n2mu, n2t` : gradient $\ln \rho$, K , ν , variables liées à la fréquence de Brunt-Väissälä, *cf.* § 7.9.4 (Page 132).
- `r, ro, dfrot, frot` : rayon, densité, vecteur des variables de la rotation et dérivées.
- Sorties :
 - `deff, dh, dv` : coefficients de diffusion turbulente effective, horizontale, verticale.

8.38 Routine `diffw_mpz/p03`

Ces routines PRIVATE du module `mod_evol`, *cf.* § E.11 (Page 294), effectue le calcul des coefficients de diffusion horizontale, verticale et effective du moment cinétique, selon les prescriptions de Mathis et al. (2004); Palacios et al. (2003) respectivement.

Description de `diffw_mpz` :

- Au premier appel, initialisation et diverses écritures.
- Calcul de D_h
- Calcul de D_v
- Calcul de D_{eff}

Appel : `diffw_mpz/p03`, appelées par `diffw`, utilisent la même liste d'appel.

8.39 Routine `dnun1`

La fonction de cette routine PRIVATE du module `mod_cesam`, *cf.* § E.13 (Page 295), est le calcul approximatif, valable uniquement pour le cas solaire, de ν_0 , $\delta\nu_{02}$ et $\delta\nu_{13}$ suivant Schatzman & Praderie Schatzman & Praderie (1990, eq. 101-102).

Description de `dnun1` :

- Tabulation de la vitesse du son en fonction de R^2 , calcul de ν_0 par intégration numérique de type Gauss (on utilise le fait que $rdr = 1/2dr^2$).
- Calcul de $\delta\nu_{02}$ et $\delta\nu_{13}$.

Appel : `dnun1` est appelée de `list`, *cf.* § 8.71 (Page 203).

SUBROUTINE `dnun1(r, c, n, rtot, nu0, dnu02, dnu13, a)`

- Entrées :
 - `r, c, n, rtot` : tables des rayons, des vitesses du son, nombre de points, rayon total.
- Sorties :
 - `nu0, dnu02, dnu13, a` : ν_0 , $\delta\nu_{02}$, $\delta\nu_{13}$, le A de la formule 100.

8.40 Routine subordonnée `ecrit_ascii`

Cette routine subordonnée de la routine `cesam` a pour fonctions la formation la mise en forme et l'écriture des fichiers de sortie ASCII, *cf.* § 3.5 (Page 29). Pour

décrire finement le comportement de la fréquence de Brunt-Väissälä, dans les fichiers de sortie destinés aux exploitations sismologiques, *cf.* § 7.10.10 (Page 154), des points sont ajoutés au voisinage des limites zones radiatives / zones convectives ; typiquement, avec la valeur du paramètre $l_0 = 3$, 20 points supplémentaires sont ajoutés de part et d'autre de chaque limite.

Description :

- Au premier appel, allocation du tableau des quantités globales.
- Allocations de tableaux et création des abscisses supplémentaires permettant de décrire finement le profil de la fréquence de Brunt-Väissälä au voisinage des limites zones radiatives / zones convectives. Par défaut, on dispose 10 abscisses supplémentaires de part et d'autre de chaque limite. Insertion de ces abscisses dans le tableau des abscisses connues.
- Formation de la table des quantités globales.
- Allocation du tableau des quantités variables et formation de la partie atmosphère.
- Formation de la partie structure interne et composition chimique. Les quantités correspondants aux abscisses déjà connues sont reprises ; celles correspondant aux point ajoutés sont calculées.
- Déallocations de tableaux.
- Appel à la routine `add_ascii`, *cf.* § 8.4 (Page 169) pour complément de données.
- Appel à la routine générique `output`, *cf.* § 8.87 (Page 213), qui oriente vers le type de sortie ASCII désirée.
- Déallocation du tableau des variables.

Appel : `ecrit_ascii`, qui n'a pas d'argument, est appelée de `cesam` à l'occasion de la sortie de fin de calcul, ou encore à l'issue de chaque pas temporel si on désire conserver tous les fichiers ASCII, *cf.* § 3.5 (Page 29).

8.41 Routine subordinnée `ecrit_rotata`

Cette routine subordinnée de `diffus` et de `evol`, a pour fonction l'écriture, dans l'évironnement du calcul, du fichier des variables et des coefficients des équations de la diffusion du moment cinétique, *cf.* § 7.9 (Page 129). Ce fichier est destiné à être exploité par le programme de dessin `des_rotata.f` du sous-directory `EXPLOIT`, *cf.* § 3.18 (Page 50). Ce fichier de travail n'est pas destiné à des exploitations mais aux mises au point. Il n'est créé que si la variable logique `ecrit_rot`, *cf.* § 4.9 (Page 60), du module `mod_donnees` est vraie, il est alors formé à l'issue de chaque appel à la routine `resout_rotata`, *cf.* § 8.118 (Page 226).

Description :

- Définition du nombre de variables et allocation du tableau à écrire.
- Calcul des coefficients des équations de la diffusion du moment cinétique, aux abscisses du vecteur de la rotation (le centre est omis en raison de la singularité de l'échelle de hauteur de presion).
- Formation du tableau des variables et écriture sur le fichier `mon_modele_coeff_rotata.dat`. Ce dernier est écrasé par chaque nouvelle

écriture.

Appel : Le sous-programme `ecrit_rota`, n'a pas d'argument.

8.42 Routine edding

Cette routine PRIVATE du module `mod_atm`, cf. § E.8 (Page 291), calcule la température dans une atmosphère purement radiative, pour une profondeur optique grise donnée suivant la loi $T(\tau)$ d'Eddington.

Appel : `edding`, appelée par la routine générique `tdetau` cf. § 8.130 (Page 235), utilise les mêmes arguments.

8.43 Routine eq_atm

La fonction de cette routine PRIVATE du module `mod_atm`, cf. § E.8 (Page 291), est de former les équations de restitution de l'atmosphère Eq. 7.11 (Page 98).

Description de `eq_atm` :

- Au premier appel, après quelques initialisations, on projette sur la base de B-spline l'atmosphère solaire, connue sous forme de DATA, on obtient ainsi la solution initiale, on définit ensuite les limites et on détermine les points de collocation.
- Extraction des grandeurs physiques, calcul de la T_{eff} , de la gravité et loi $T(\tau)$.
- Calcul des grandeurs thermodynamiques, de ∇ en particulier, et dérivées, formation des équations aux points de collocation et aux points limites.

Appel : `eq_atm` est appelée de `coll_atm`, cf. § 8.15 (Page 176) :

SUBROUTINE `eq_atm(fait,li,xchim,cx,y,be,ae,r_rac,l_rac,xcoll_atm)`

- Entrées :
 - `fait`, `li`, `xchim`, `cx` : `fait=1` point de collocation, `fait=2` point limite, indice de la limite, composition chimique/gramme, indice du point de collocation,
 - `y` : variables et dérivées,
 - `r_rac`, `l_rac`, `xcoll_atm` : rayon au fond de l'atmosphère, luminosité, abscisses des points de collocation.
- Sorties :
 - `be`, `ae` : seconds membres et jacobien.

8.44 Routine eq_diff_chim

La routine PRIVATE `eq_diff_chim` du module `mod_evol`, cf. § E.11 (Page 294), a pour fonction de calculer, dans l'espace physique, les résidus des équations 7.36 (Page 115), les quantités figurant à gauche des produits scalaires et leurs dérivées par rapport aux abondances.

Pour un élément chimique d'indice i fixé, les coefficients de l'équation de diffusion calculés dans les routines de type `diffm`, cf. § 8.29 (Page 182) sont de trois sortes :

- Les coefficients v_i de x_i , dans l'équation 7.29 (Page 113), codés $v(i)$.
- Les coefficients de diffusion microscopique $d_{i,j}^*$, de $\frac{\partial x_j}{\partial m}$, dans l'équation 7.30 (Page 113), codés $d(i,j)$.
- Les coefficients $d_{i,i}$, de $\frac{\partial x_j}{\partial m}$, dans l'équation 7.30 (Page 113), sont ajoutés à $d(i,i)$ dans les routines de type `diff`, cf. §8.32 (Page 184).

les dérivées par rapport à x_k , $k = 1, n_{\text{elem}}$ sont respectivement notées $dv(i,k) = dv(nbelem(k-1)+i)$ et $dd(i,j,k) = dd(nbelem(nbelem(k-1)+j-1)+i)$.

Description de `eq_diff_chim`

- Au point courant :
 - Initialisations diverses,
 - extraction des variables de structure au point de calcul `cx`, appel à l'équation d'état, détermination des opacités, calcul des réactions thermonucléaires et des coefficients de l'équation de diffusion,
 - détermination des résidus des équations 7.36 (Page 115), des quantités figurant à gauche des produits scalaires et de leurs dérivées par rapport aux abondances.
- Au point limite externe, évaluation et exploitation de la perte de moment angulaire.

Appel : `eq_diff_chim` est appelée par `resout_chim`, cf. §8.117 (Page 225).

SUBROUTINE `eq_diff_chim(nu,y,dt,ad,as,bd,bs)`

- Entrées :
 - `nu`, `dt`, `y` : point de calcul en $m^{\frac{2}{3}}$, pas temporel, variables et dérivées.
- Sorties :
 - `as`, `ad`, `bs`, `bd` : dérivées par rapport aux abondances $x_{i,j}$ des arguments des produits scalaires avec S_j^{mc} et $\frac{\partial x_{i,j}}{\partial \nu}$ et résidus.

8.45 Routine `eq_diff_poisson`

La routine PRIVATE `eq_diff_poisson` du module `mod_evol`, cf. §E.11 (Page 294), a pour fonction de calculer, dans l'espace physique, les coefficients des équations, à résoudre dans `poisson_initial.f`, cf. §8.97 (Page 218), pour l'initialisation du potentiel gravitationnel.

Description de `eq_diff_poisson` :

- initialisation de constantes et allocations.
- calcul des coefficients au temps $t + dt$.
- formation des coefficients au point courant.
- formation des coefficients aux points limites.

Appel : `eq_diff_poisson` est appelée par `poisson_initial`, cf. §8.97 (Page 218).

SUBROUTINE `eq_diff_poisson(fait,nu,ad,as,bs)`

- Entrées :
 - `fait`, `nu` : `fait=0` : point courant, `fait=1, 2` : limite, point de calcul en $m^{\frac{2}{3}}$, variables et dérivées.

- Sorties :
- `as`, `ad`, `bs` : dérivées par rapport à la variable Ψ des arguments des produits scalaires avec $S_j^{m_c}$ et $\frac{\partial x_{i,j}}{\partial \nu}$, résidus.

8.46 Routine `eq_diff_rota3/4`

La routine PRIVATE `eq_diff_rota` du module `mod_evol`, cf. § E.11 (Page 294), a pour fonction de calculer, dans l'espace physique, les coefficients et les résidus des équations de la diffusion du moment cinétique, § 7.9 (Page 129).

Description de `eq_diff_rota3/4` :

- Interpolation au point d'abscisse lagrangienne `nu` des coefficients des équations.
- Détermination de la nature du milieu convectif ou radiatif par interpolation du vecteur de convection.
- Au point courant (`fait=1`) :
 - Formation des équations dans le milieu convectif.
 - Formation des équations dans le milieu radiatif.
- Sur les limites (`fait=2, 3`) :
 - Partie intégrée des équations pour un noyau convectif et une zone convective externe.

Appel : `eq_rota` est appelée par `resout_rota`, cf. § 8.118 (Page 226).

SUBROUTINE `eq_diff_rota(fait,nu,y,ad,as,bd,bs)`

- Entrées :
 - `fait`, `nu`, `y` : `fait=1` : point courant, `fait=2, 3` : limite, point de calcul en $m^{\frac{2}{3}}$, variables et dérivées.
- Sorties :
 - `as`, `ad`, `bs`, `bd` : dérivées par rapport aux variables Ω , U , θ , Λ , ψ des arguments des produits scalaires avec $S_j^{m_c}$ et $\frac{\partial x_{i,j}}{\partial \nu}$, résidus.

8.47 Routine générique `etat`

Les routines de type `etat` constituent les interfaces entre CESAM2k et la routine de calcul de l'équation d'état qui a souvent une origine externe. Ces routines sont des routines PRIVATE du module `mod_etat`, cf. § E.5 (Page 290).

Pour une pression gazeuse P , une température T et une composition chimique X_i données, `etat` calcule la densité ρ , l'énergie interne spécifique U , c_p , δ , ∇_{ad} , et leurs dérivées premières, α , β , Γ_1 . Certaines routines d'équation d'état utilisent des tables que l'on doit créer dans le sous-directory `SUN_STAR_DATA`, cf. § 2.2 (Page 13). Le nom de ces tables doit être indiqué dans le fichier de données. La **routine générique** de calcul de l'équation d'état est `etat`, routine PUBLIC du module `mod_etat`.

Appel : `etat` est appelée de divers endroits, en particulier depuis `evol`, `thermo`, `thermo_atm`, `eq.dif_chim`, `cesam`.


```

SUBROUTINE etat(p,t,xchim,deriv,
1 ro,drop,drot,drox,u,dup,dut,dux,
2 delta,deltap,deltat,deltax,cp,dcpp,dcpt,dcp,
3 gradad,dgradadp,dgradadt,dgradadx,alfa,beta,gamma1)

```

- Entrées :
 - p, t, xchim, deriv : pression, température, composition chimique par gramme, deriv=.TRUE. calcul des dérivées;
- Sorties :
 - ro, drop, drot, drox : densité et dérivées;
 - u, dup, dut, dux : énergie interne et dérivées;
 - delta, deltap, deltat, deltax : δ et dérivées;
 - cp,dcpp,dcpt,dcp : c_p et dérivées;
 - gradad, dgradadp, dgradadt, dgradadx : ∇_{ad} et dérivées;
 - alfa, beta, gamma1 : α, β, Γ_1 .

8.48 Routine etat_ceff

La routine PRIVATE `etat_ceff` du module `mod_etat`, cf. § E.5 (Page 290), utilise le formalisme CEFF. C'est une adaptation à CESAM2k des programmes originaux de J.Christensen-Dalsgaard par A.Baglin, M.Auvergne, Y. Lebreton, P.Morel & B.Pichon.

Les routines qui relèvent de ce package sont les suivantes :

```

bilinc, clmnew, cvmgrp, cvmgz, cvmgn, cvmgt, cvmgm, dmpeqs, derive,
eostst, eqstfc, eqstpc, eqstrh, eosder, f4der, f4n, f4mhd, hheion,
hmnion, hvionac, inteffc, ismax, isamax, ismin, leq, neweta, omegac,
phderc, prthvi, setgm1, setf4, seteqs, blockdata bleqstc, setcns,
sdot, ssum, sscal, scopy, saxpy, storec, zeroc.

```

Ces routines sont regroupées dans un même package et compilées simultanément avec `etat_ceff`; elles sont donc des routines PRIVATE du module `mod_etat`, cf. § E.5 (Page 290). Diverses modernisations concernant la programmation ont été effectuées par B.Pichon et P.Morel. Le BLOCKDATA initial a été transformé en un tableau de paramètres. D'origine externe, ces routines ne sont pas détaillées.

PB Cette routine qui demande beaucoup de ressources est peu robuste, son usage est délicat.

8.49 Routine etat_eff

La routine PRIVATE `etat_eff` du module `mod_etat`, cf. § E.5 (Page 290), utilise le formalisme EFF, la source, rassemblée dans le package EFF, est une adaptation à CESAM2k des programmes originaux de J. Christensen-Dalsgaard par A. Baglin, M. Auvergne & P.Morel.

Les routines qui relèvent de ce package sont les suivantes :

```

bilin, eqstf, eqstp, hviona, inteff, omega, phder, setcns, store,
theffp, zero.

```

Elles sont exploitées sous forme de routines subordonnées de la routine `etat_eff`. Le BLOCKDATA initial a été transformé en un tableau de paramètres.

8.50 Routine `etat_gong1`

Selon les prescriptions du "Solar Model Comparison Project" Christensen-Dalsgaard (1988), la routine PRIVATE `etat_gong1` du module `mod_etat`, *cf.* §E.5 (Page 290), suppose le milieu totalement ionisé, mais ne tient compte ni de la dégénérescence, ni de l'énergie interne d'ionisation, ni de la pression de radiation.

8.51 Routine `etat_gong2`

Suivant "Solar Model Comparison Project" Christensen-Dalsgaard (1988), la routine PRIVATE `etat_gong2` du module `mod_etat`, *cf.* §E.5 (Page 290), utilise le formalisme EFF simplifié, sans tenir compte, ni de la pression de radiation, ni de la dégénérescence. La mise en œuvre de `etat_gong2` a bénéficié de la collaboration de J. Provost.

8.52 Routine `etat_mhd`

La routine PRIVATE `etat_mhd` du module `mod_etat`, *cf.* §E.5 (Page 290), assure l'interface avec l'équation d'état solaire tabulée MHD de W.Däppen *cf.* Mihalas et al. (1988). Les tables, en ASCII compressé, sont distribuées dans le package de l'équation d'état MHD, par W.Däppen. Les plus récentes *i.e.* avec "correction τ ", sont disponibles à l'adresse FTP anonymous `usc.edu` directory `pub/astro-physics/mhd-oc/wd-evo`; leur mise en place est décrite au §2.2.5 (Page 19).

Les routines qui relèvent de ce package sont regroupées et compilées simultanément avec `etat_mhd`; ce sont donc des routines PRIVATE du module `mod_etat`. Il s'agit de :

`fmttob`, `mhdpx`, `mhdpx1`, `mhdpx2`, `intpt`, `mhdst`, `mhdst1`, `rtab`, `rabu`,
`quint`, `quintd`, `lir`.

Les noms des tables à utiliser sont indiqués directement dans `etat_mhd` il n'y a donc pas lieu d'en indiquer les noms dans le fichier de données. Ces tables doivent être créées par l'utilisateur, *cf.* §2.2.5 (Page 19), et placées dans le sous-directory `SUN_STAR_DATA`, *cf.* §2.2 (Page 13).

8.53 Routines `etat_opal`, `etat_opalX`, `etat_opalZ`

Ces routines PRIVATE du module `mod_etat`, *cf.* §E.5 (Page 290), assurent l'interface avec les diverses versions de l'équation d'état tabulée OPAL (Iglesias & Rogers, 1991). Les tables d'équation d'état, disponibles sur le site WEB `ftp://www-physics.llnl.gov/pub/opal/eos/` sont très grandes.

Suivant les valeurs de l'abondance en métaux requise, il faut contruire les tables ainsi qu'il est décrit au §2.2.4 (Page 16), et les placer dans le sous-directory

FIGURE 8.1 – Environnement de la routine `evol` de gestion de l'évolution de la composition chimique et du moment cinétique.

`SUN_STAR_DATA`, cf. § 2.2 (Page 13). Les routines utilisent les tables OPAL en binaire. La transformation des `BLOCKDATA` de la distribution sous la forme de paramètres, a permis des gains de place et de temps calcul appréciables. La mise en œuvre de ces routines a bénéficié de collaborations de S.Brun, J.M. Marques & L.Piau.

Bien qu'identiques, les routines utilisées pour la lecture et l'interpolation des tables utilisent des paramétrages différents. Il s'agit de :

`esac`, `gmass`, `quad`, `r_opal_bin`, `rad_sub ro_new`, `t6rinterp`.

Elles sont introduites sous la forme de routines subordonnées et reproduites pour chacune des routines d'équation d'état OPAL.

8.54 Routine evol

La routine PUBLIC `evol` du module `mod_evol`, cf. § E.11 (Page 294), gère le mélange convectif, l'intégration temporelle de la composition chimique, éventuellement du moment cinétique. Un organigramme schématique est reproduit Figure 8.1 (Page 194).

Description

- Initialisations diverses, éliminations éventuelles pour le mélange convectif *uniquement* de limites zones radiatives / zones convectives trop externes, détermination si le modèle est totalement radiatif ou convectif.
- Détermination de la répartition en masse – en $(m/M_{\odot})^{2/3}$ – utilisée pour l'interpolation de la composition chimique et du moment cinétique. Les points retenus sont, ou bien identiques aux points de la grille utilisée pour les variables globales, ou bien situés aux centres des intervalles entre deux points de cette grille, ou bien encore ceux de la *grille fixe*, cf. § 3.8 (Page 35). A défaut d'une description physique de l'évolution des discontinuités aux limites zones radiatives / zones convectives l'utilisation alternative des milieux de grille comme points d'interpolation crée une légère diffusion numérique destinée à stabiliser l'évolution. Aux voisinages du centre et des limites zones radiatives / zones convectives, des points supplémentaires sont ajoutés pour améliorer la précision. Cette dernière disposition n'est appliquée qu'avec l'utilisation d'une *grille fixe*.
- Les indices du début et de la fin de chaque zone convective sont ensuite identifiés dans la grille pour l'interpolation de la composition chimique ; en enlevant les "c" de commentaires, des algorithmes de test permettent des vérifications.
- Une alternative se présente alors : :
 - On tient compte de la diffusion microscopique avec ou sans diffusion du moment cinétique, il y a appel à la routine `diffus`, cf. § 8.36 (Page 186), sauf si le modèle est complètement convectif depuis le début de l'évolution – ou de la reprise du calcul. Au retour de la routine `diffus`, la précision du calcul de l'évolution des abondances est estimée.
 - Il y a intégration et mélange convectif classiques, en tenant compte d'éventuels

gains ou pertes de masse, et en estimant la précision du calcul de l'évolution des abondances :

- d'abord pour les zones convectives où l'intégration de la composition chimique est réalisée de façon simultanée avec le mélange convectif,
- ensuite pour les zones radiatives.
- La composition chimique est ensuite ajustée en fonction des retraits ou avancées des limites zones radiatives / zones convectives.
- Enfin, on calcule les coefficients d'interpolation de la composition chimique en tenant compte des discontinuités aux limites zones radiatives / zones convectives.
- Une renormalisation de la composition chimique de façon à assurer $X+Y+Z = 1$ est effectuée dans l'espace des splines. Cette renormalisation est nécessaire pour des arguments :
 - **Physiques**: les masses atomiques n'étant pas des nombres entiers, les équations de l'évolution thermonucléaire de la composition chimique ne peuvent conserver *exactement* $X + Y + Z = 1$. Pour ce faire, il faudrait tenir compte de corrections relativistes.
 - **Numériques**: les équations implicites de l'évolution de la composition chimique étant résolues de façon itérative, il y a nécessairement une erreur numérique résultant de l'appréciation de la convergence. Les erreurs d'arrondi sont aussi à prendre en ligne de compte.
- Une estimation du pas temporel à venir est faite en fonction des valeurs, requise et obtenue, de la précision de l'intégration. En cas de non convergence d'algorithmes itératifs, ou si la précision requise n'est pas atteinte, le modèle est réinitialisé lors du retour à `resout`, et le calcul est alors repris avec un pas temporel divisé par 2. *L'estimation du pas temporel ne porte que sur l'estimation de la précision de l'intégration de la composition chimique.*

Appel : `evol` est appelée par `resout`, cf. § 8.116 (Page 223).

SUBROUTINE `evol(compt,dt,dts,repnd)`

- Entrées :
 - `compt` : compteur des itérations globales, `compt=0` pour la première itération globale de chaque nouveau pas temporel.
 - `dt` : pas temporel.
- Sorties :
 - `dts` : estimation du pas temporel à utiliser pour le pas temporel suivant.
 - `repnd=.TRUE.` : il faut réduire le pas temporel à cause de TdS ou de non convergence.

8.55 Routine générique `f_rad`

La routine PRIVATE `f_rad` du module `mod_evol`, cf. § E.11 (Page 294), constitue la *routine générique* du calcul des accélérations radiatives. Le calcul de l'accélération radiative est effectué avec la routine dont le nom `NOM_FRAD` est indiqué dans le fichier de données, cf. § 3.10 (Page 38).

Appel : `f_rad` est appelée par `diffm_br`, cf. § 8.30 (Page 183).

SUBROUTINE `f_rad(lum,ray,t,kap,dkapx,nel,ychim,ioni,grav,g_rad,dg_rad)`

- Entrées :
 - `lum, ray, t, kap, dkapx` : luminosité, rayon, température, opacité, dérivée/ X (mole).
 - `nel, ychim, ioni, grav` : nombre d'électrons libres par volume, composition chimique par mole, taux d'ionisation, gravité.
- Entrées/Sorties : `g_rad(i)` : vecteur des accélérations radiatives, la gravité est `grav+g_rad` sur l'élément d'indice `i`
- Sorties : `dg_rad(i,j)` : matrice des dérivées des accélérations radiatives sur l'élément `i` par rapport à l'abondance par mole de l'élément `j`.

8.56 Routine subordonnée fcmx

Cette routine subordonnée de `resout` a pour fonction d'indiquer s'il sera nécessaire d'utiliser le nombre maximum de couches pour les modèles à calculer.

Description :

- Définition de la variable logique `logic` suivant la réalisation d'un des critères d'utilisation du nombre maximum de couches.
- Définition de la variable logique de sortie `cmx` suivant les critères de sortie ASCII définis au § D.1.4 (Page 271).

Appel : `fcmx` est appelé de `resout` à l'issue de la convergence du modèle.

SUBROUTINE `fcmx(clogic)`

- Sortie :
 - `clogic = .TRUE.` : il faudra utiliser le nombre maximum de couches.

8.57 Routine from_alecian

La subroutine PRIVATE `from_alecian` du module `mod_bp_for_alecian`, cf. § E.10 (Page 294), regroupe une partie des routines de calcul des accélérations radiatives, selon la seconde version du formalisme développé par G.Alécian. D'origine externe les fonctions des routines de ce regroupement ne sont pas détaillées.

8.58 Routine hopf

Cette routine PRIVATE du module `mod_atm`, cf. § E.8 (Page 291), calcule la température dans une atmosphère purement radiative pour une profondeur optique grise donnée suivant la loi $T(\tau)$ de Hopf, cf. Mihalas (1978).

Description :

- Mise en forme des données pour l'interpolation.
- Interpolation de la température et dérivées.

Appel : `hopf`, appelée par la routine générique `tdetau` cf. § 8.130 (Page 235), utilise les mêmes arguments.

8.59 Routine `iben`

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), de type `nuc`, cf. § 8.77 (Page 206), calcule l'énergie gravifique pour dl'initialisation de l'évolution avec pré-séquence principale, cf. § 7.4.3 (Page 100). Cette initialisation est effectuée dans la routine `cesam`, cf. § 8.10 (Page 173) ; à cet effet, le nom de la routine de réactions thermonucléaires à utiliser est temporairement redéfini : `nom_nuc=iben`.

Appel : `iben`, appelée par la routine générique `nuc`. Les arguments sont ceux d'une routine de type `nuc`.

8.60 Routine générique `ini_ctes`

Cette routine PUBLIC du module `mod_donnees`, cf. § E.3 (Page 278), est la *routine générique* pour l'initialisation de la plupart des constantes physiques. Suivant le nom indiqué dans le fichier de données, cf. § 3.4 (Page 26), il est fait appel à la routine d'initialisation correspondante, cf. § 8.24 (Page 180).

Appel : Cette routine générique est appelée par `lit_n1`, cf. § 8.74 (Page 205), elle n'a pas d'argument.

8.61 Routine `initialise_rota`

La fonction de cette routine PRIVATE du module `mod_cesam`, cf. § E.13 (Page 295), est l'initialisation des fonctions décrivant la diffusion du moment cinétique.

Description :

- Détermination du rayon de l'étoile R_* .
- Appels aux routines `w_initial`, cf. § 8.63 (Page 199), et `u_initial`, cf. § 8.62 (Page 198), pour l'initialisation de la vitesse angulaire Ω et de la vitesse de circulation méridienne U .
- Déduction des valeurs initiales de θ , Λ et ψ de celles de Ω et U .

Appel : `initialise_rota`, appelée par `cesam`, cf. § 8.10 (Page 173), n'a pas d'argument.

8.62 Fonction `initialise_u`

Cette fonction PRIVATE du module `mod_cesam`, cf. § E.13 (Page 295), initialise la vitesse de circulation méridienne U du formalisme de la diffusion du moment cinétique, cf. § 7.9.5.1 (Page 133).

Description : La valeur initiale de la vitesse de circulation méridienne est initialisée par :

$$|\Omega|U_0 \left(2 - \frac{R}{R_*}\right)^{-1}, \quad U_0 \sim 0.01.$$

Appel : `initialise_u` est appelée par `initialise_rota`, cf. § 8.61 (Page 198).

FUNCTION initialise_u(r,rstar)

- r, rstar : rayon local, rayon total.

8.63 Fonction initialise_w

Cette fonction PRIVATE du module `mod.cesam`, cf. § E.13 (Page 295), initialise la vitesse angulaire Ω du formalisme de la diffusion du moment cinétique, cf. § 7.9.5.1 (Page 133).

Description : La valeur locale de la vitesse angulaire est donnée par :

$$\Omega = |\Omega_0| \left\{ 1 + \Omega_f \left[1 + \cos \left(\pi \frac{R}{R_*} \right) \right] \right\},$$

Ω_0 et Ω_f sont respectivement la vitesse angulaire initiale et le facteur de forme définis dans le fichier de données `mon_modele.don`, cf. § 3.4 (Page 26).

Appel : `initialise_w` est appelée par `initialise_rota`, cf. § 8.61 (Page 198).

FUNCTION initialise_w(r,rstar)

- r, rstar : rayon local, rayon total.

8.64 Routine inter

La fonction de cette routine PUBLIC du module `mod.variables`, cf. § E.4 (Page 286), est de déterminer par interpolation inverse, à partir de son développement sur une base de B-splines, l'indice de la couche correspondant à une masse ou à un rayon donné. Puis, par interpolation directe, de déterminer en ce point les valeurs des variables et de leurs dérivées premières, par rapport à la variable de masse ou de rayon. En variables lagrangiennes, l'interpolation est effectuée en $(m/M_\odot)^{2/3}$; en variables eulériennes l'interpolation est effectuée en (r/R_\odot) . La variable d'indice 6 *i.e.* $f(6)$, affectée à la fonction de répartition, cf. § 7.2.3 (Page 86), est utilisée pour transmettre l'indice décimal de la couche où s'effectue l'interpolation.

Appel : `inter` est appelée par divers programmes, en particulier `evol`.

SUBROUTINE inter(m23_ou_r2,bp,q,qt n,knot,x,f,dfdx,r2,m23)

- Entrées :
 - m23_ou_r2 : chaîne de **3 caractères**, si m23_ou_r2='m23' interpolation en masse, si m23_ou_r2='r2' interpolation en rayon.
 - bp, q, qt, n, knot, x : éléments pour l'interpolation.
 - r2, m23 : points d'interpolation.
- Sorties :
 - f, dfdx : fonctions ($f(6)$: indice fractionnaire d'interpolation), dérivées.

8.65 Routine `inter_atm`

Pour l'atmosphère, cette routine PUBLIC du module `mod_exploit`, cf. §E.14 (Page 296), a une fonction similaire à la routine `inter` précédente. Elle est utilisée par le programme `for037_2k` du module `mod_exploit`, cf. §3.18 (Page 50), qui permet l'interpolation d'un modèle en des masses ou des rayons définis à l'avance.

Description :

- Localisation de l'intervalle d'interpolation en masse, rayon ou indice de couche.
- Détermination de l'indice fractionnaire d'interpolation inverse par algorithme de Brent.
- Interpolation des variables de l'atmosphère.
- L'indice fractionnaire est conservé.

Appel : `inter_atm` est appelée par le programme `for037_2k`.

```
SUBROUTINE inter_atm(m_ou_r,ne_atm,bp_atm,x_atm,xt_atm,n_atm,
1 ord_atm,knot_atm,m_atm,r_atm,x,f,dfdx)
```

- Entrées :
 - `m23_ou_r2` : chaîne de 3 caractères, interpolation en masse si `m23_ou_r2='m23'`, ou interpolation en rayon si `m23_ou_r2='r2□'`.
 - `ne_atm`, `bp_atm`, `x_atm`, `xt_atm`, `n_atm`, `ord_atm`, `knot_atm` : éléments pour l'interpolation par B-spline.
 - `m_atm`, `r_atm` : masses et rayons de la discrétisation de l'atmosphère.
- Sorties :
 - `f`, `dfdx` : fonctions, dérivées, rayon, masse.
 - `f(ne_atm+1)=x_int` : indice fractionnaire de l'abscisse d'interpolation.

8.66 Routines `k5750`, `k5777`

Ces deux routines PRIVATE du module `mod_atm`, cf. §E.8 (Page 291), calculent la température dans une atmosphère non purement radiative pour une profondeur optique grise donnée, suivant les lois $T(\tau)$ solaires $T_{\text{eff}} = 5750$ K et $T_{\text{eff}} = 5777$ K calculées par C. Van't Veer avec le programme Atlas 9 de Kurucz.

Description :

- Mise en forme des données pour l'interpolation.
- Interpolation de la température et dérivées.

Appel : `k5750`, `k5777`, sont appelées par la routine générique `tdetau`, cf. §8.130 (Page 235). Elles utilisent les mêmes arguments.

8.67 Routine `kappa_cond`

Cette routine PRIVATE du module `mod_opacite`, cf. §E.6 (Page 290), calcule l'opacité conductive en utilisant les ajustements polynômiaux de Iben (1975), et complète l'opacité radiative.

Description : Bien que remaniée pour son adaptation à CESAM2k, cette routine, en grande partie d'origine externe, n'est pas détaillée. La moyenne harmonique avec l'opacité radiative est effectuée à l'issue du calcul.

Appel : Cette routine est appelée par la plupart des routines de calcul d'opacité à l'issue du calcul de l'opacité radiative.

SUBROUTINE `kappa_cond(xh,t,ro,kappa,dkapdt,dkapdr,dkapdx)`

- Entrées :
 - `xh` : composition chimique en fraction de masse.
 - `t` : température.
 - `ro` : densité.
- Entrées/Sorties :
 - `kap` : opacité.
 - `dkapdt`, `dkapdr`, `dkapdx` : dérivées de l'opacité en fonction de la température, densité, abondance d'hydrogène.

8.68 Routine `lim_atm`

Cette routine PUBLIC du module `mod_atm`, cf. §E.8 (Page 291), permet de restituer l'atmosphère à partir d'une loi $T(\tau)$ grise, cf. §7.3.2 (Page 92).

Description :

- Au premier appel :
 - Initialisation de constantes.
 - Recherche, dans l'environnement, d'un fichier binaire d'atmosphère pour reprise, et tests de cohérence.
 - Reprise de l'atmosphère en binaire ou initialisation en ASCII.
 - Recherche du rayon bolométrique et initialisations diverses.
 - Projection sur la base de B-splines.
- Pour chaque appel :
 - Commentaires divers.
 - Appel à `coll_atm` pour l'intégration.
 - Au cas où une liste complète du modèle est requise, formation des tableaux des quantités nécessaires.
 - Ecriture du fichier binaire d'atmosphère `mon_modele_atm`.

Appel : `lim_atm` appelée par la routine générique d'atmosphère `atm`, cf. §8.7 (Page 172), a la même liste d'appel.

8.69 Routines `lim_gong1`, `lim_tau1`

Ces deux routines PUBLIC du module `mod_atm`, cf. §E.8 (Page 291), sont deux formes similaires de la restitution d'une atmosphère dans l'approximation mono-couche, cf. §7.3.1 (Page 91). Dans la routine `lim_gong1`, deux facteurs $\beta = 7.22$ et $\lambda = 6$, affectent respectivement les équations de la pression et de la luminosité, leur effet étant, avec une équation d'état supposant l'ionisation totale, d'obtenir un modèle solaire qui ressemble au soleil actuel.

FIGURE 8.2 – Organigramme de la routine `lim_zc` de détermination des limites zones radiatives / zones convectives.

PB Avec l'approximation monocouche, la pression turbulente est ignorée dans l'atmosphère.

Appel : `lim_gong1`, `lim_tau1`, appelées par la fonction générique d'atmosphère `atm`, cf. §8.7 (Page 172), ont la même liste d'appel.

8.70 Routine `lim_zc`

Cette routine PRIVATE du module `mod_static`, cf. §E.12 (Page 295), a trois fonctions principales :

- Initialisation, si nécessaire, de la répartition des couches en fonction du nombre de points à utiliser.
- Localisation des limites des zones radiatives et convectives, au besoin avec overshoot.
- Détermination des coefficients de répartition.

`lim_zc` est une routine importante de CESAM2k, son organigramme est présenté Figure 8.2 (Page 202).

Description de `lim_zc`

- Initialisation :
 - Initialisations diverses effectuées au premier appel, en particulier, détermination des points limites et calcul des coefficients d'intégration correspondants à cette répartition.
 - Au cas où le nombre de couches doit être modifié, détermination de la nouvelle fonction d'espacement, changement de la base de B-splines, redéfinition des points limites et calcul des coefficients d'intégration. Avec l'option `grille_fixe`, cf. §3.8 (Page 35), la grille d'interpolation de la composition chimique n'est modifiée que si le nombre de couches a varié de plus de la quantité `dn_fixe` fixée dans `resout`, cf. §8.10 (Page 173).
- Localisation :
 - Calcul, en chaque point de la grille, de la différence $\nabla_{\text{rad}} - \nabla_{\text{ad}}^*$, localisation des changements de signe et identification de la nature de la limite à l'aide de la variable logique `lconv` qui est `.TRUE.` lorsqu'on passe d'une zone radiative à une zone convective, en traversant la limite dans le sens croissant de la variable d'espace *i.e.* masse, rayon ou encore, indice.
 - Elimination des zones convectives mal définies *i.e.* trop proches du centre, de la limite externe ou encore n'affectant que très peu de couches.
 - Affinement, par dichotomie, de la localisation des limites retenues.
 - Si besoin, extensions des zones convectives et analyse des chevauchements éventuels.
- Détermination :
 - Des coefficients de répartition qui permettent d'amener chacune des limites sur un point de la grille.

Appel : `lim_zc`, appelée par `resout`, n'a que des arguments d'entrée.

SUBROUTINE `lim_zc(no_rep,new_nqs)`

`no_rep=.TRUE.` : Il n'y a pas eu d'initialisation ou de réinitialisation du modèle.

`new_nqs=.TRUE.` : Il faut effectuer une redistribution des couches du modèle quasi-statique, la constante de répartition ayant été modifiée.

8.71 Routine list

Cette routine PRIVATE du module `mod_cesam`, cf. § E.13 (Page 295), n'a que des arguments d'entrée. Sa fonction est de construire le listing permettant de suivre le déroulement du calcul et de produire, pour différentes époques de l'évolution, une liste détaillée des variables du modèle, des abondances, des rapports isotopiques à la surface et des flux de neutrinos – l'implantation de ces derniers a bénéficié d'une collaboration avec G.Berthomieu.

Description :

- Initialisations diverses, en particulier des indices d'identification des éléments chimiques.
- Construction d'un cartouche indiquant : l'âge du modèle, la température effective, la luminosité, le rayon, la gravité, la pression, la température, les abondances par fraction de masse de ^1H et ^4He centrales, les proportions d'énergie libérées par les cycles thermonucléaires et la gravité, la nature du modèle, la variation relative de masse, ses valeurs présente et initiale, les abondances et les rapports isotopiques en surface, éventuellement la période et la vitesse de rotation de surface.
- *Si le listing complet est requis*, le modèle est listé couche par couche du sommet de l'enveloppe vers le centre, un cartouche étant disposé toutes les 10 couches pour faciliter l'identification des quantités écrites. Les limites zones radiatives / zones convectives sont matérialisées. L'atmosphère est ensuite reproduite couche par couche, en partant de l'enveloppe vers la surface ; on indique ensuite la déplétion des éléments en surface, les abondances relatives en nombre à la surface, les flux de neutrinos, et une estimation, pour le cas solaire, des grandeurs astérosismologiques : ν_0 , $\delta\nu_{02}$, $\delta\nu_{13}$, A .

Appel : `list` est appelée par `cesam`, cf. § 8.10 (Page 173).

SUBROUTINE `list(alfa,anub8,anube7,anun13,anuo15,anupep,anupp,beta,`
`1 compg,cp,delta,dcapdr,dcapdt,depsdr,depsdt,d2p,d2ro,`
`2 chaine,convec,ecritout,epsilon,gamma,gamma_atm,gradad,grada_atm,`
`3 gradconv,gradc_atm,gradrad,gradr_atm,hp,i_cno,i_gr,i_pp,i_3a,`
`4 kap,l,m,mu,mue,m_atm,p,pt,pt_atm,p_atm,r,ro,ro_atm,r_atm,t,tau,`
`5 teff,tx_ioni,t_atm,u,vaissala,w,z,degene)`

- `anub8`, `anube7`, `anun13`, `anuo15`, `anupep`, `anupp` : Tables des nombres de neutrinos des diverses sources.
- `alfa`, `beta`, `delta` : α , β , δ .

- `comp`, `cp`, `hp` : Composition chimique par unité de masse, chaleur spécifique à pression constante, échelle de hauteur de pression.
- `kap`, `dcapdr`, `dcapdt` : Opacité, dérivées par rapport à la densité et à la température.
- `epsilon`, `depsdr`, `depsdt` : Energie nucléaire des différents cycles et énergie graviphique, dérivées par rapport à la densité et à la température.
- `d2p`, `d2ro` : Dérivées secondes de la pression et de la densité centrales.
- `chaîne`, `convec`, `ecritout` : Statut du modèle, nature radiative ou convective de la couche, écriture du listing complet.
- `gradad`, `gradconv`, `gradrad`, `gamma` : Gradient adiabatique, gradient convectif, gradient radiatif, Γ .
- `grada_atm`, `gradc_atm`, `gradr_atm`, `gamma_atm` : Gradient adiabatique, gradient convectif, gradient radiatif, Γ pour l'atmosphère.
- `i_cno`, `i_pp`, `i_3a`, `i_gr` : Pourcentages d'énergie d'origine nucléaire PP, CNO, 3α et gravifique.
- `mu`, `mue`, `vaissala` : Poids moléculaire moyen, poids moléculaire moyen par électron libre, fréquence de Brunt-Väissälä.
- `p`, `pt`, `t`, `r`, `l`, `m`, `ro`, `u` : Pression gazeuse, pression totale, température, rayon, luminosité, masse, densité, énergie interne spécifique.
- `p_atm`, `pt_atm`, `t_atm`, `r_atm`, `m_atm`, `ro_atm` : Pression gazeuse, pression totale, température, rayon, masse, densité dans l'atmosphère.
- `tau`, `teff` : Epaisseur optique, température effective
- `tx_ioni`, `w`, `z`, `degene` : Taux d'ionisation, vitesse angulaire, abondance des métaux, facteur de dégénérescence.

8.72 Routine `lit_binaire`

Cette routine PUBLIC du module `mod_exploit`, cf. § E.14 (Page 296), permet la lecture des fichiers binaires, cf. § C (Page 265), à l'exception des fichiers d'atmosphère.

Description :

- Recherche et lecture du fichier de données `mon_modele.don`, cf. § 3.4 (Page 26).
- Lecture des paramètres du modèle, calcul des dimensions et allocation des tableaux.
- Lecture du modèle.

Appel : `lit_binaire` appelée par des programmes d'exploitation, cf. § 3.18 (Page 50), du module `mod_exploit`, cf. § E.14 (Page 296), n'a que des arguments d'entrée.

SUBROUTINE `lit_binaire(chaine,dt)`

- `chaîne`, `dt` : Nom du modèle, pas temporel.

8.73 Routine `lit_hr`

Cette routine PUBLIC du module `mod_exploit`, cf. § E.14 (Page 296), permet la lecture des fichiers ASCII de diagramme HR `mon_modele.HR`, cf. § D.2 (Page

272).

Description :

- Lecture du premier enregistrement du fichier ASCII `mon_modele.HR`, pour allocations.
- Lecture du fichier dans son intégralité.
- Formation des tests de lecture.

Appel : `lit_hr` est appelée par des programmes d'exploitation, *cf.* § 3.18 (Page 50), du module `mod_exploit`, *cf.* § E.14 (Page 296), en particulier par le programme `des_hr`.

SUBROUTINE `lit_hr`(`init`,`chaine`,`fin`,`erreur`,`log_l`,`log_r`,`log_teff`,
1 `vrot`,`wrot`)

- Entrées :
 - `init=.TRUE.`, `chaine` : Initialisation, nom du modèle.
- Sorties :
 - `fin=.TRUE.`, `erreur=.TRUE.` : Fin de fichier, erreur de lecture.
 - `log_l`, `log_r`, `log_teff` : Logarithmes décimaux de la luminosité, du rayon, de la température effective.
 - `vrot`, `wrot` : vitesses linéaire et angulaire de la couche externe.

8.74 Routine `lit_nl`

La fonction de cette routine PUBLIC des modules `mod_donnees`, *cf.* § E.3 (Page 278) et `mod_exploit`, *cf.* § E.14 (Page 296), est de lire les NAMELISTS du fichier d'entrée `mon_modele.don`, *cf.* § 3.4 (Page 26).

Description :

- Recherche et lecture des NAMELISTS du fichier de données `mon_modele.don`, *cf.* § 3.4 (Page 26). En cas d'échec de lecture, tentative de trouver et d'utiliser des fichiers de données de versions précédentes de CESAM.
- Vérification de la cohérence des données.
- Détermination des abondance par masse de l'hydrogène X , hélium Y et métaux Z , mise en place de tests.
- Initialisation des principales constantes de physique par appel à la routine `ini_ctes`, *cf.* § 8.60 (Page 198).
- Définition du type de rotation à utiliser.

Appel : `lit_nl` est appelée par `cesam`, *cf.* § 8.10 (Page 173).

SUBROUTINE `lit_nl`(`wrot`)

- Sortie :
 - `wrot` : Vitesse angulaire.

FIGURE 8.3 – Environnement de la routine de calcul des réactions thermonucléaires. La routine `nuc` est appelée de `cesam` pour l’initialisation des abondances. Lors d’une évolution, elle est appelée i) de `eq_dif_chim` ou de `rkimps` pour le calcul de l’évolution des abondances, suivant qu’il y ait ou non suivi de la diffusion des éléments chimiques, ii) de `thermo` pour le calcul de l’énergie thermonucléaire libérée, iii) de `list` pour le calcul des flux de neutrinos. Au premier appel, il y a initialisation des abondances initiales et tabulation des réactions nucléaires suivant le cycle et la compilation des réactions à utiliser, *cf.* § 3.4 (Page 26). Lors des appels suivants, il y a calcul des variations des abondances ou/et de l’énergie thermonucléaire libérée, ou encore des flux de neutrinos.

8.75 Fonction logique `lmix`

La valeur de cette fonction PRIVATE du module `mod_evol` est `.TRUE.` dans les zones convectives.

Description :

- Un premier test détermine si on se trouve dans la zone convective externe.
- Une fois la localisation de l’abscisse obtenue, on affecte à la fonction la valeur de la table de mélange pour l’indice obtenu.

Appel : `lmix` est appelée principalement des routines concernées par la diffusion du moment cinétique et des éléments chimiques.

LOGICAL FUNCTION `lmix(nu)`

- `nu` : (masse) ^{$\frac{2}{3}$} .

8.76 Routine `modif_mix`

Cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), permet de personnaliser une mixture, *cf.* § 4.2 (Page 54).

Description :

- Recherche du fichier de la nouvelle mixture `mon_modele.modif_mix`, *cf.* § 4.2 (Page 54).
- Lecture des modifications en DeX des abondances initiales.
- Modification des abondances des éléments lourds.

Appel : `modif_mix`, appelée par `abon_ini`, *cf.* § 8.3 (Page 169), n’a pas d’argument.

8.77 Routine générique `nuc`

La routine PUBLIC `nuc` du module `mod_nuc`, *cf.* § E.9 (Page 292) est la **routine générique** qui, par l’intermédiaire de la routine spécifique dont le nom est défini dans le fichier de données *cf.* § 3.4 (Page 26), gère le calcul des réactions et de l’énergie nucléaire, *cf.* § 7.11 (Page 158). Les routines de type `nuc` sont disponibles dans 4 catégories différentes, suivant le domaine de température couvert :

- **test** : réseaux de réactions simplifiés destinés à des tests.
- **pré-3 α** : utilisables depuis la pré-séquence principale jusqu'à l'amorçage des réactions de l'hélium, intervalle de température couvert : 0.5-100 MK. Les réseaux de réactions utilisés décrivent les cycles PP et CNO.
- **3 α** : utilisables depuis la pré-séquence principale jusqu'à l'amorçage des réactions du carbone, intervalle de température couvert : 0.5-200 MK. Les réseaux de réactions utilisés décrivent les cycles PP, CNO et 3 α .
- **co** : utilisables depuis la pré-séquence principale jusqu'au cycle de l'oxygène, intervalle de température couvert : 0.5-2000 MK. Les réseaux de réactions utilisés décrivent les cycles PP, CNO, 3 α , la combustion du carbone et de l'oxygène. Au delà l'hypothèse d'équilibre quasi-statique n'est plus justifiée.

Les isotopes utilisés dans les réseaux nucléaires dépendent, d'une part, du type de modèle à calculer et, d'autre part, des éléments chimiques auxquels on s'intéresse plus particulièrement.

La mise en œuvre de ces routines a bénéficié, à divers titres, de collaborations de B. Pichon, Y. Lebreton, S. Brun et D. Cordier. La source de CESAM2k contient diverses routines du type `nuc` spécialement adaptées à des cas spécifiques de l'évolution stellaire.

Description : On décrit l'organigramme d'une routine de type `nuc`, à l'exception de `iben`, cf. § 8.59 (Page 197), et de `pp1`, cf. § 8.98 (Page 218), dont les algorithmes diffèrent quelque peu du cadre général. Un argument d'entrée, `fait`, définit la partie du calcul à effectuer :

- `fait=1`
 - Initialisation du nombre d'isotopes du réseau nucléaire.
 - Initialisation du réseau nucléaire par appel à `rq_reac`, cf. § 8.122 (Page 229).
 - Initialisation des abondances par appel à `abon_ini`, cf. § 8.3 (Page 169).
 - Détermination du type et des caractéristiques de l'isotope fictif et calcul des abondances isotopiques initiales.
 - Initialisations diverses et écritures.
- `fait=2`
 - Calcul des taux de réaction.
 - Formation du jacobien.
- `fait=3`
 - Calcul de l'énergie nucléaire.
- `fait=4`
 - Calcul des flux de neutrinos. Les "flux sur terre" sont calculés en supposant que ces neutrinos sont produits à 1UA de la terre, même si le modèle n'est pas un modèle solaire.

Appel : `nuc` est appelée de divers endroits, en particulier de `rkimps`, `thermo`, `eq_dif_chim`, `cesam`.

```
SUBROUTINE nuc(t,ro,comp,dcomp,jac,deriv,fait,
1 epsilon,et,ero,ex,hhe,be7e,b8e,n13e,o15e,f17e)
```

- Entrées :
 - `t`, `ro` : Température, densité.

- `deriv=.TRUE.` : On calcule le jacobien et les dérivées.
- `fait` : `fait=1` : initialisation de la composition chimique, `fait=2` : calcul des taux et du jacobien si `deriv=.TRUE.` `fait =3` : énergie nucléaire et si `deriv=.TRUE.` : dérivées / température, densité et abondances, `fait=4` : calcul des neutrinos.
- Entrées/Sorties :
 - `comp` : Abondances des isotopes par mole.
- Sorties :
 - `dcomp`, `jac` : Dérivées temporelles des abondances, jacobien.
 - `epsilon`, `et`, `ero`, `ex` : Energie thermonucleaire et dérivées.
 - `hhe`, `be7e`, `b8e`, `n13e`, `o15e`, `f17e` : Nombres de neutrinos pour les réactions ${}^1\text{H}(p, \beta^+\nu){}^2\text{H}$, ${}^7\text{Be}(e^-, \nu\gamma){}^7\text{Li}$, ${}^8\text{B}(\beta^+\nu){}^8\text{Be}$ ${}^{13}\text{N}(\beta^+\nu){}^{13}\text{C}$, ${}^{15}\text{O}(\beta^+\nu){}^{15}\text{N}$, ${}^{17}\text{F}(\beta^+\nu){}^{17}\text{O}$.

8.78 Routine générique opa

La routine PUBLIC `opa` du module `mod_opa`, cf. § E.6 (Page 290), est la **routine générique** qui, par l'intermédiaire de la routine spécifique dont le nom, `NOM_OPA`, est défini dans le fichier de données cf. § 3.4 (Page 26), réalise la gestion du calcul de l'opacité. Pour un triplet : (composition chimique, température, densité), les routines de type `opa` déterminent l'opacité Rosseland $\kappa(T, \rho, X, Z)$, ainsi que ses dérivées premières par rapport à i) la température, ii) la densité et, iii) la teneur en hydrogène. Deux types d'opacités sont implantés dans `CESAM2k`, des opacités analytiques et des opacités tabulées. Les premières sont des approximations, leur utilisation est robuste et ne demande que très peu de ressources ; peu précises elles sont réservées aux tests. Les secondes, plus précises, mais d'utilisation délicate, sont destinées aux exploitations.

Description : Les routines de type `opa`, la plupart d'origine externe, n'ont pas une structure commune ; chacune d'entre elles fait l'objet d'une description personnalisée. L'opacité conductive, cf. § 8.67 (Page 200), est introduite à l'issue du calcul de l'opacité radiative.

Appel : `opa` est appelée de divers endroits en particulier de `evol`, `thermo`, `thermo_atm`, `eq_dif_chim`, `cesam`.

```
SUBROUTINE opa(xh,t,ro,kappa,dkapdt,dkapdr,dkapdx)
```

- Entrées :
 - `xh`, `t`, `ro` : composition chimique, température, densité.
- Sorties :
 - `kappa`, `dkapdt`, `dkapdr`, `dkapdx` : opacité, dérivées.

8.79 Routine opa_gong

La routine PRIVATE `opa_gong` du module `mod_opa`, cf. § E.6 (Page 290), est une routine de type opacité analytique. Le formalisme utilisé est décrit dans le "Solar Model Comparison Project" de Christensen-Dalsgaard (1988). L'opacité moyenne

de Rosseland est approchée par une formule de Kramers améliorée, sous la forme d'une somme de deux termes, l'un valable pour l'intérieur stellaire et l'autre pour l'extérieur :

$$\kappa_{\text{R}}^{-1} = \kappa_e^{-1} + \kappa_i^{-1}, \quad \kappa_e = c_e \rho^{m_e} T^{n_e}, \quad \kappa_i = c_i \rho^{m_i} T^{n_i},$$

avec : $c_e = 1.6236784 \times 10^{-33}$, $m_e = 0.407895$, $n_e = 9.28289$, $c_i = 7.1548412 \times 10^{13}$, $m_i = 0.138316$, $n_i = -1.97541$. Cette forme analytique correspond à $Z = 0.02$.

Appel : `opa_gong`, appelée par la routine générique `opa` cf. § 8.78 (Page 208), utilise les mêmes arguments.

8.80 Routine opa_houdek9

La routine PRIVATE `opa_houdek9` du module `mod_opa`, cf. § E.6 (Page 290), est une routine de type opacité tabulée. Elle utilise les interpolations des tables de Livermore 95 prolongées, pour les basses températures, par les opacités Kurucz ou Alexander ; elles peuvent, ou non, tenir compte de la conductivité électronique. Elles utilisent les bibliothèques et les données du package de Houdek & Rogl (1996). La mise en forme des données est décrite au § 2.2.3 (Page 15). Pour son utilisation, il est nécessaire d'indiquer dans la routine `opa_houdek` le chemin qui permet d'atteindre les données `OPINPATH_95` ; il faut aussi indiquer au `LINK` le chemin de la bibliothèque `libopint.a`. La mise en œuvre de `opa_houdek` a bénéficié d'une collaboration avec N.Audard. La version originale du package de routines d'interpolation de G.Houdek se trouve à : <ftp://solaris.tuwien.ac.at/incoming/>.

Description :

- Au premier appel :
 - Lecture et initialisation des tables.
 - Adaptation et encadrement des données.
 - Calcul de l'opacité, en cas d'échec appel à `opa_yveline_lisse`, cf. § 8.85 (Page 212).

Les routines d'interpolation d'origine externe ne sont pas décrites. Pour plus d'informations, se référer aux README du package "v9" du sous-directory `SUN_STAR_DATA`.

Appel : `opa_houdek9`, appelée par la routine générique `opa` cf. § 8.78 (Page 208), utilise les mêmes arguments.

8.81 Routine opa_int_zsx

La routine PRIVATE `opa_int_zsx` du module `mod_opa`, cf. § E.6 (Page 290), est une routine de type opacité tabulée. `opa_int_zsx` interpole les *anciennes* tables Livermore 1991, prolongées par les opacités Kurucz pour les basses températures. Elle utilise une interpolation spline 4D linéaire³ suivant les axes (T_6 , $\ln \rho/T_6^3$, X , Z), $T_6 \equiv T \times 10^{-6}$. Les données Livermore 1991 sont actuellement caduques. Des difficultés de convergence, observées principalement dans la restitution de l'atmosphère,

3. La base utilisée est duale, les calculs d'interpolation sont alors immédiats.

peuvent provenir de l'interpolation linéaire – non dérivable aux points de la table – dans ces tables aux larges intervalles tabulaires.

Description :

- Au premier appel :
 - Initialisation de constantes et extension des tables par interpolation.
 - Mise en forme des tables pour l'interpolation.
 - Estimations des limites des tables.
- Localisation du point de calcul.
- Interpolation et calcul des dérivées.

Appel : `opa_int_zsx`, appelée par la routine générique `opa` cf. § 8.78 (Page 208), utilise les mêmes arguments.

8.82 Routine `opa_opal2`

La routine PRIVATE `opa_opal2` du module `mod_opa`, cf. § E.6 (Page 290), est une routine de type opacité tabulée. Elle permet de dépasser le seuil $Z < 0.1$ des données de OPAL. Elle tient compte des modifications de composition chimique dues à la nucléosynthèse $H \rightarrow He \rightarrow C \rightarrow N \rightarrow O$. Elle ne permet le calcul de l'opacité que pour des températures supérieures à 1eV, le raccord avec des opacités basse température n'existant pas. Elles n'est donc pas utilisable pour les modèles comportant une atmosphère. Le calcul de l'opacité est basé sur les tables d'opacité de type 2 de Livermore, il utilise les routines du paquetage `z14xcotrin21`, cf. § 8.137 (Page 240), de A.I. Boothroyd. Les tables utilisées, placées dans le sous-directory `SUN_STAR_DATA` sont automatiquement décompressées et transformées en binaire à l'occasion de leur première utilisation.

`opa_opal2` est d'usage extrêmement délicat.

Description :

- Au premier appel :
 - Identification du chemin directory où se trouvent les fichiers de données.
 - Identification des indices de C, N, O, estimation de l'abondance en masse des éléments de la mixture de GN93 qui ne sont pas pris en compte et du rapport [O/Fe].
 - Détermination des rapports C/Z, O/Z, pour les excès de C et O de la mixture utilisée par rapport à GN93.
 - Lecture en binaire des tables d'opacité concernées si elles existent, sinon création.
 - Calcul de X, Y, Z initiaux et définition des limites pour les sorties de table.
 - Calcul de l'opacité.
 - Estimation des dérivées par dérivation numérique.

Les routines d'interpolation, d'origine externe ne sont pas décrites, cf. § 8.137 (Page 240).

Appel : La liste d'appel de cette routine est celle de sa routine générique `opa` cf. § 8.78 (Page 208), avec un argument supplémentaire, la variable logique `cno` qui permet une alternative :

- `cno=.FALSE.` : seules les variations de l'abondance en carbone sont prises en compte.
- `cno=.TRUE.` : les variations de l'abondance en carbone et en oxygène sont prises en compte.

En raison de l'absence d'un raccord avec des opacités basse température, `opa_opal2` est utilisée pour prolonger les routines d'opacité `opa_yveline`, cf. § 8.84 (Page 212), et `opa_yveline_lisse`, cf. § 8.85 (Page 212), au delà de $Z > 0.1$.

Il est toutefois possible d'utiliser `opa_opal2` directement, en codant dans le fichier de données, cf. § 3.4 (Page 26), `NOM_OPA=opa_opal2_cno`, pour tenir compte seulement des variations de l'abondance du carbone, ou `NOM_OPA=opa_opal2_co`, pour tenir compte des variations de l'abondance du carbone et de l'oxygène.

8.83 Routine opa_opalCO

La routine PRIVATE `opa_opalCO` du module `mod_opa`, cf. § E.6 (Page 290), est du type opacité tabulée. Elle permet de dépasser le seuil $Z < 0.1$ des données de OPAL. Elle tient compte des modifications de composition chimique dues à la nucléosynthèse $H \rightarrow He \rightarrow C \rightarrow O$. Elle ne permet le calcul de l'opacité que pour des températures supérieures à 1eV, le raccord avec des opacités basse température n'existant pas. Elle est inutilisable pour les modèles comportant une atmosphère.

Les tables utilisées par ces routines, placées dans le sous-directory `SUN_STAR_DATA` sont automatiquement décompressées et transformées en binaire à l'occasion de leur première utilisation.

La mise en œuvre de `opa_opalCO` a bénéficié d'une collaboration avec L.Piau.

Description :

- Au premier appel :
 - Identification des indices de H, He, C, O.
 - Définition de la métallicité de la table et calcul des rapports C/Z, O/Z.
 - Détermination des abondances de H, He, C, O, et des excès de C et O.
 - Calcul des opacités.

`opa_opalCO` utilise une interpolation moins sophistiquée des tables OPAL que `opa_opal2`. Les routines d'interpolation utilisées sont des routines subordonnées de `opa_opalCO`; d'origine externe, ces routines ne sont pas décrites. Il s'agit de : `opac`, `cointerp`, `t6rinterp`, `readco`, `quad`, `spline`, `splint`, `fity`, `fitx`, `getd`, `interp`, `smooth`, `opaltab`.

Appel : `opa_opalCO`, appelée par la routine générique `opa` cf. § 8.78 (Page 208), utilise les mêmes arguments.

8.84 Routine opa_yveline

La routine PRIVATE `opa_yveline` du module `mod_opa`, cf. § E.6 (Page 290), est une routine de type opacité tabulée. L'interpolation du package construit par Y. Lebreton pour le calcul de l'opacité OPAL (Iglesias & Rogers, 1991). `opa_yveline` effectue une interpolation cubique dans la table, linéaire aux bords. La table en ASCII se trouve dans le fichier `opa_yveline_etendu.dat` du sous-directory `SUN_STAR_DATA`.

Extraire cette table et la transformer en un fichier binaire à l'aide du programme `ascii2bin_opa` de ce package *cf.* § 2.2.1 (Page 13).

Lorsque l'abondance en éléments lourds dépasse $Z > 0.1$ `opa_yveline` fait appel à `opa_opal2`, *cf.* § 8.82 (Page 210), qui permet de tenir compte de la nucléosynthèse $H \rightarrow He \rightarrow C \rightarrow N \rightarrow O$.

Description :

- Au premier appel :
 - Vérification de la cohérence des données.
 - Vérification de l'accessibilité du point d'interpolation.
 - Calcul de l'opacité si $Z < 0.1$.
 - Appel à `opa_opal2` si $Z > 0.1$.

Les routines d'interpolation utilisées par `opa_yveline` sont introduites sous forme de routines subordonnées, d'origine externe, elles ne sont pas décrites. Il s'agit de : `kappa_opal`, `intl_in_opal`, `intl_opal`, `sub_quad`, `lect_opal`, `lpol_op`, `pos_table_op`.

Appel : `opa_yveline`, appelée par la routine générique `opa` *cf.* § 8.78 (Page 208), utilise les mêmes arguments.

8.85 Routine `opa_yveline_lisse`

La routine PRIVATE `opa_yveline_lisse` du module `mod_opa`, *cf.* § E.6 (Page 290), est une routine de type opacité tabulée. L'exploitation du package construit par Y. Lebreton pour le calcul de l'opacité OPAL (Iglesias & Rogers, 1991) et effectué par `opa_yveline_lisse` est un lissage par des Béziérs, dont l'ordre peut être facilement adapté pour chacune des dimensions (X, Z, T, ρ). Par défaut l'ordre 4 est utilisé, l'ordre 2 correspond à l'interpolation linéaire. Le lissage assure des dérivées d'ordre élevé, il est plus robuste que l'interpolation. Le prix à payer est une précision réduite. La table en ASCII se trouve dans le fichier `opa_yveline_etendu.dat` du sous-directory `SUN_STAR_DATA`. Extraire cette table et la transformer en un fichier binaire à l'aide du programme `ascii2bin_opa` de ce package *cf.* § 2.2.1 (Page 13).

Description :

- Au premier appel :
 - Lecture des tables.
 - Définitions diverses.
- Vérification de l'accessibilité du point d'interpolation.
- Appel à `opa_opal2` si $Z > 0.1$.
- Si $Z < 0.1$, recherche des indices, calcul de l'opacité.

Appel : `opa_opa_yveline_lisse`, appelée par la routine générique `opa` *cf.* § 8.78 (Page 208), utilise les mêmes arguments.

8.86 Routine `osc_adia`, `osc_invers`, `osc_noad`

Ces routines PRIVATE du module `mod_cesam`, *cf.* § E.13 (Page 295), de type `output`, *cf.* § 8.87 (Page 213), ont pour fonction la création des fichiers utilisés

respectivement pour le calcul des oscillations adiabatiques, des inversions et des oscillations non adiabatiques.

Description :

Les tableaux d'indices `tglob(nglob)` et `tvar(nvar)`, adaptés à chaque fichier de sortie, définissent l'ordre dans lequel sont disposés les éléments de ces fichiers ASCII. Les éléments des tableaux `var(P, T, L...)` et `glob(age, d2ro...)` construits dans la routine `cesam`, cf. § 8.10 (Page 173), et complétés dans la routine `add_ascii`, cf. § 8.4 (Page 169), sont réorganisés dans les tableaux intermédiaires `eglob(nglob)` et `evar(nvar, itot)`, suivant l'ordre défini dans `tglob(nglob)` et `tvar(nvar)`. Le fichier de sortie est ensuite créé à partir des fichiers intermédiaires. L'identification des quantités est décrite au § D.1 (Page 267). Un exemple de création de tels fichiers est décrit au § 4.8 (Page 58).

Appel : `osc_adia`, `osc_invers`, `osc_noad`, appelées par la routine générique `output` cf. § 8.87 (Page 213), utilisent les mêmes arguments.

8.87 Routine générique output

Cette routine PRIVATE du module `mod_cesam`, cf. § E.13 (Page 295), est la *routine générique* des routines de formation des fichiers ASCII de sortie, utilisés pour les calcul des oscillations ou des inversions, cf. § 8.86 (Page 213), ou encore pour créer un fichier de sortie ASCII personnalisé, cf. § 4.8 (Page 58).

Description :

Suivant le type de sortie ASCII indiqué par la variable `NOM_OUTPUT` du fichier de données `mon_modele.don`, cf. § 3.4 (Page 26), la routine de formation du fichier ASCII est appelée.

Appel : `output` est appelée par la routine `ecrit_ascii`, cf. § 8.40 (Page 188) subordonnée de la routine `cesam`; `output` n'a que des arguments d'entrée.

```
SUBROUTINE output(var, glob, itot, ivar)
```

- Entrées :
- `glob`, `itot`, `ivar` : quantités globales, nombre total de points, nombre de variables

8.88 Routine générique pertm

Cette routine PUBLIC du module `mod_static`, cf. § E.12 (Page 295), est la *routine générique* du calcul de la perte de masse.

Description :

Suivant le type de sortie ASCII indiqué par la variable `NOM_PERTM` du fichier de données `mon_modele.don`, cf. § 3.4 (Page 26), la routine de perte de masse est appelée.

Appel : `pertem` est appelée par `evol`, cf. § 8.54 (Page 195), et `resout`, cf. § 8.116 (Page 223).

SUBROUTINE `pertem(dt)`)

- Entrée
- `dt` : pas temporel

8.89 Routine `pertm_ext`, `pertm_msol`

Pour ces deux routines PRIVATE du module `mod_static`, *cf.* § E.12 (Page 295), la perte de masse est supposée linéaire en fonction du temps. Le taux \dot{M}_{\odot} par an est celui indiqué dans le fichier de données `mon_modele.don`, *cf.* § 3.4 (Page 26). Avec `pertm_msol`, le taux de perte de masse ainsi que le traitement du vent *cf.* § 4.5 (Page 56), sont annulés dès que la masse du modèle atteint la masse solaire.

Description :

- Extraction des masses.
- Formation du tableau des masses corrigées de la perte/gain de masse.

Appel : ces deux routines ont les mêmes arguments que leur routine générique `pertm`, *cf.* § 8.88 (Page 214).

8.90 Routine `pertm_tot`

Cette routine PRIVATE du module `mod_static`, *cf.* § E.12 (Page 295), tient compte de la masse perdue par la transformation de la masse en énergie, par les réactions thermonucléaires. La perte de masse externe est supposée linéaire en fonction du temps. Le taux \dot{M}_{\odot} par an est celui indiqué dans le fichier de données `mon_modele.don`, *cf.* § 3.4 (Page 26).

Description :

- Extraction des masses.
- Calcul de la perte de masse locale due aux réactions nucléaires.
- Formation du tableau des masses corrigées de la perte/gain de masse totale.

Appel : cette routine a les mêmes arguments que sa routine générique `pertm`, *cf.* § 8.88 (Page 214).

8.91 Routine `pertem_waldron`

Cette routine PRIVATE du module `mod_static`, *cf.* § E.12 (Page 295), utilise la perte de masse empirique de Waldron, citée dans l'article A.Ap 229 (1990) 469-474.

Description :

- Extraction des masses.
- Calcul de la perte de masse.
- Formation du tableau des masses corrigées de la perte de masse.

Appel : cette routine a les mêmes arguments que sa routine générique `pertm`, *cf.* § 8.88 (Page 214).

8.92 Routine générique pertw

Cette routine PRIVATE du module `mod_evol`, cf. §E.11 (Page 294), est la *routine générique* du calcul de la perte/gain de moment cinétique.

Description :

Suivant la description de la perte de moment cinétique indiquée par la variable `NOM_PERTW` du fichier de données `mon_modele.don`, cf. §3.4 (Page 26), la routine spécifique de perte de moment cinétique est utilisée.

Appel : `pertw` est appelée par `coeff_rota`, cf. §8.13 (Page 175).

SUBROUTINE `pertw(nu,omega,r,mw_dot)`

- Entrées :
 - `nu`, `omega`, `r` : masse, vitesse angulaire, rayon au point de calcul.
- Sortie
 - `mw_dot` : taux de perte/gain de moment cinétique.

8.93 Routine pertw_loc

Avec cette routine PRIVATE du module `mod_evol`, cf. §E.11 (Page 294), de type `pertw`, la perte / gain de moment cinétique est proportionnelle à l'énergie cinétique de rotation locale et supposée localisée dans la fraction de masse externe $M > pw_extend$, cf. §7.9.10 (Page 147).

Description :

La variation temporelle de moment cinétique par unité de masse est modélisée par :

$$\dot{\mathcal{M}}_{\Omega} = \gamma R^2 \Omega^2, \quad |\gamma| \sim 10^{-13} \sim 10^{-14}$$

Le paramètre γ correspond au taux `p_pertw` indiqué dans le fichier de données `mon_modele.don`, cf. §3.4 (Page 26), le taux de perte/gain de moment cinétique `mw_dot` est nul en deça de la limite fixée.

Appel : cette routine utilise les mêmes arguments que sa routine générique `pertw`, cf. §8.92 (Page 215).

8.94 Routine pertw_ptm

Avec cette routine PRIVATE du module `mod_evol`, cf. §E.11 (Page 294), de type `pertw`, le taux `p_pertw` de perte de moment cinétique indiqué dans le fichier de données `mon_modele.don`, cf. §3.4 (Page 26), est utilisé comme un paramètre fixant l'efficacité de la perte de moment cinétique résultant de la perte de masse, cf. §7.9.10 (Page 147). La perte de moment cinétique est supposée localisée dans la fraction de masse externe $M > pw_extend$.

Description :

L'apport/perte de moment cinétique par unité de masse est approché par :

$$\dot{\mathcal{M}}_{\Omega} = \frac{\dot{\mathcal{M}}}{m_{\text{rot}}} R^2 \Omega_p.$$

Le taux de perte/gain de moment cinétique `mw_dot` est fixé à 0 en deça de la limite fixée.

Appel : cette routine utilise les mêmes arguments que sa routine générique `pertw`, *cf.* § 8.92 (Page 215).

8.95 Routine `pertw_sch`

Avec cette routine PRIVATE du module `mod_evol`, *cf.* § E.11 (Page 294), de type `pertw`, le taux `p_pertw` de perte de moment cinétique indiqué dans le fichier de données `mon_modele.don`, *cf.* § 3.4 (Page 26), est utilisé comme un paramètre fixant l'efficacité d'une perte de moment cinétique proportionnelle à Ω^3 (Schumanish), *cf.* § 7.9.10 (Page 147).

Description :

La variation temporelle de moment cinétique par unité de masse est modélisée par :

$$\dot{\mathcal{M}}_{\Omega} = aR^2\Omega_s^3, \quad |a| \sim 1.10^{-9}\text{s}.$$

Le taux de perte de moment cinétique `mw_dot` est fixé à 0 en deça de la limite fixée.

Appel : cette routine utilise les mêmes arguments que sa routine générique `pertw`, *cf.* § 8.92 (Page 215).

8.96 Routine `planetoides`

CESAM2*k* offre la possibilité de tenir compte des variations de composition chimique dues à des chutes de planétoïdes dans la zone convective externe, *cf.* § 7.8.2 (Page 116). Cette possibilité est gérée par la routine PUBLIC `planétoïdes` du module `mod_nuc`, *cf.* § E.9 (Page 292).

Description :

- Lors d'un appel d'initialisation, depuis la routine `evol`, on effectue une recherche dans l'environnement du fichier où sont indiquées les fractions de masse des planétoïdes, *cf.* § 4.6 (Page 57). Si un tel fichier est trouvé, les fractions de masse sont normalisées, puis réparties entre les différents isotopes du réseau nucléaire, et enfin transformées en fraction par mole ; la variable logique `l_planet` est déclarée `.TRUE`.
- Lors des appels suivants, une correction aux taux de réactions nucléaires est effectuée lorsque l'âge du modèle est dans l'intervalle de temps des chutes. *cf.* § 7.8.2 (Page 116).

Appel : `planetoides` est appelée par :

- `evol`, *cf.* § 8.54 (Page 195), pour initialisation.
- `rkimps`, *cf.* § 8.120 (Page 227) ou `eq_dif_chim`, *cf.* § 8.36 (Page 186), à la suite d'un appel à la routine `nuc`, *cf.* § 8.77 (Page 206).

Les arguments sont optionnels.

SUBROUTINE `planetoides(xchim,dxchim,jac,m_planet,mw_planet)`

- Entrées :
 - `xchim` : Composition chimique par mole.
- Entrées/Sorties :
 - `dxchim`, `jac` : Taux de réactions nucléaires et jacobien.
 - `m_planet`, `mw_planet` : taux d'accroissement de masse et de moment cinétique.

8.97 Routine poisson_initial

Cette routine PRIVATE du module `mod_cesam`, permet l'initialisation du potentiel gravitationnel Ψ dans le cadre du formalisme de Mathis & Zahn (2004) de la diffusion du moment cinétique. L'intégration de l'équation différentielle utilise la méthode des éléments finis Galerkin.

Description :

- initialisations.
- formation des produits scalaires avec appel à `eq_diff_poisson.f`, cf. § 8.45 (Page 191).
- résolution du système linéaire.
- test de précision et gestion des itérations.
- initialisation des coefficients du potentiel gravitationnel.

Appel : `poisson_initial` est appelée de la routine `cesam.f`, cf. § 8.10 (Page 173), lors des initialisations.

8.98 Routine pp1

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *test*, cf. § 8.77 (Page 206). Selon les " *Computational procedures for GONG solar model project*" (Christensen-Dalsgaard, 1988), seul l'isotope ^1H est explicitement pris en compte. L'abondance initiale d'hydrogène, le taux de la réaction et l'énergétique sont calculés *in situ* par une approximation analytique du cycle PP.

Toutefois la structure générale des routines de type `nuc` est respectée.

Appel : `pp1` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.99 Routine pp3

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *test*, cf. § 8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, cf. § 7.11.4 (Page 161). Cette routine, qui ne concerne que le cycle PP, a pour vocation de servir pour les tests de mise au point.

Appel : `pp3` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.100 Routine ppcno10

Cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est de type `nuc`, catégorie *pré-3α*, *cf.* § 8.77 (Page 206).

Les isotopes ^1H , ^3He , ^4He , ^7Li , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Be sont supposés à l'équilibre, *cf.* § 7.11.4 (Page 161). Cette routine a été construite pour décrire la diffusion microscopique du lithium, en tenant compte de l'évolution au long de la pré-séquence principale.

Appel : `ppcno10` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.101 Routine ppcno10BeBFe

Cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est de type `nuc`, catégorie *pré-3α*, *cf.* § 8.77 (Page 206).

Les isotopes ^1H , ^3He , ^4He , ^6Li , ^7Li , ^9Be , ^{11}B , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , ^{56}Fe et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Be sont supposés à l'équilibre, *cf.* § 7.11.4 (Page 161). Cette routine a été construite pour décrire la diffusion microscopique des éléments légers et du fer, en tenant compte de l'évolution au long de la pré-séquence principale.

Appel : `ppcno10BeBFe` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.102 Routine ppcno10Fe

Cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est de type `nuc`, catégorie *pré-3α*, *cf.* § 8.77 (Page 206).

Les isotopes ^1H , ^3He , ^4He , ^7Li , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , ^{56}Fe et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Be sont supposés à l'équilibre, *cf.* § 7.11.4 (Page 161). Cette routine a été construite pour décrire la diffusion microscopique du lithium et du fer.

Appel : `ppcno10Fe` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.103 Routine ppcno10K

Cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est de type `nuc`, catégorie *pré-3α*, *cf.* § 8.77 (Page 206).

Les isotopes ^1H , ^3He , ^4He , ^7Li , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et deux éléments fictifs `ExK` et `Ex`, sont explicitement pris en compte ; ^2H , ^7Be sont supposés à l'équilibre, *cf.* § 7.11.4 (Page 161). Les deux éléments fictifs représentent respectivement les éléments de masse inférieure et supérieure au potassium. Cette routine a été construite pour décrire la diffusion microscopique différentielle entre les éléments de masse supérieure ou inférieure au potassium.

Appel : ppcno10K appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.104 Routine ppcno11

Cette routine PRIVATE du module `mod_nuc`, cf. §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , cf. § 8.77 (Page 206).

Les isotopes ^1H , ^2H , ^3He , ^4He , ^7Li , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte; ^7Be est supposé à l'équilibre, cf. § 7.11.4 (Page 161). Cette routine a été construite pour décrire la pré-séquence principale, en évitant de suivre l'abondance du ^7Be dont l'échelle de temps d'évolution est infime.

Appel : ppcno11 appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.105 Routine ppcno12

Cette routine PRIVATE du module `mod_nuc`, cf. §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , cf. § 8.77 (Page 206).

Les isotopes ^1H , ^2H , ^3He , ^4He , ^7Li , ^7Be , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte; Aucun isotope n'est à l'équilibre, cf. § 7.11.4 (Page 161). Cette routine a été construite pour décrire l'évolution sans élément à l'équilibre.

Appel : ppcno12 appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.106 Routine ppcno12Be

Cette routine PRIVATE du module `mod_nuc`, cf. §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , cf. § 8.77 (Page 206).

Les isotopes ^1H , ^2H , ^3He , ^4He , ^7Li , ^7Be , ^9Be , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte; Aucun isotope n'est à l'équilibre, cf. § 7.11.4 (Page 161). Cette routine a été construite pour décrire la diffusion microscopique du beryllium en tenant compte de l'évolution au long de la pré-séquence principale.

Appel : ppcno12Be appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.107 Routine ppcno12BeBFe

Cette routine PRIVATE du module `mod_nuc`, cf. §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , cf. § 8.77 (Page 206). Les isotopes ^1H , ^2H , ^3He , ^4He , ^6Li , ^7Li , ^7Be , ^9Be , ^{11}B , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , ^{56}Fe et l'élément fictif `Ex`, sont explicitement pris en compte. Aucun isotope n'est à l'équilibre, cf. § 7.11.4 (Page 161). Cette

routine a été construite pour décrire la diffusion microscopique des éléments légers, et du fer, en tenant compte de l'évolution au long de la pré-séquence principale.

Appel : `ppcno12BeBFe` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.108 Routine `ppcno12Li`

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , cf. § 8.77 (Page 206).

Les isotopes ^1H , ^2H , ^3He , ^4He , ^6Li , ^7Li , ^7Be , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte. Cette routine a été construite pour décrire la diffusion microscopique des éléments légers, plus particulièrement du lithium, en tenant compte de l'évolution au long de la pré-séquence principale. Aucun isotope n'est à l'équilibre, cf. § 7.11.4 (Page 161).

Appel : `ppcno12Li` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.109 Routine `ppcno3a12Ne`

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *3 α* , cf. § 8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , ^{18}O , ^{20}Ne , ^{22}Ne et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, cf. § 7.11.4 (Page 161).

Appel : `ppcno3a12Ne` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.110 Routine `ppcno3a9`

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *3 α* , cf. § 8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, cf. § 7.11.4 (Page 161).

Appel : `ppcno3a9` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.111 Routine `ppcno3aco`

Cette routine PRIVATE du module `mod_nuc`, cf. § E.9 (Page 292), est de type `nuc`, catégorie *co*, cf. § 8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , ^{20}Ne , ^{23}Na , ^{24}Mg , ^{27}Al , ^{28}Si , ^{31}P , ^{32}S , et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, cf. § 7.11.4 (Page 161). `ppcno3aco` permet d'atteindre la combustion de l'oxygène.

Appel : `ppcno3aco` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.112 Routine `ppcno9`

Cette routine PRIVATE du module `mod_nuc`, *cf.* §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* *cf.* §8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, *cf.* §7.11.4 (Page 161).

Appel : `ppcno9` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.113 Routine `ppcno9Fe`

Cette routine PRIVATE du module `mod_nuc`, *cf.* §E.9 (Page 292), est de type `nuc`, catégorie *pré-3 α* , *cf.* §8.77 (Page 206). Les isotopes ^1H , ^3He , ^4He , ^{12}C , ^{13}C , ^{14}N , ^{15}N , ^{16}O , ^{17}O , Fe56 et l'élément fictif `Ex`, sont explicitement pris en compte ; ^2H , ^7Li , ^7Be sont supposés à l'équilibre, *cf.* §7.11.4 (Page 161). Cette routine a été construite pour décrire la diffusion microscopique du fer.

Appel : `ppcno9Fe` appelée par la routine générique de réactions nucléaires `nuc` a la même liste d'appel.

8.114 Routine `print_ctes`

Cette routine PUBLIC du module `mod_donnees`, *cf.* §E.3 (Page 278), a pour fonction la transcription sur l'unité d'écriture des valeurs des principales constantes physiques utilisées.

Appel : `print_ctes` est appelée de `cesam`, *cf.* §8.10 (Page 173)

SUBROUTINE `print_ctes(i)`

- Entrée
- `i` : indice de l'unité FORTRAN sur laquelle l'écriture est effectuée.

8.115 Routine `read_ascii`

Cette routine PUBLIC du module `mod_exploit`, *cf.* §E.14 (Page 296), a pour fonction la lecture des fichiers ASCII d'oscillation, *cf.* §8.86 (Page 213).

Description

- Ouverture du fichier de type `mon_modele.osc`, lecture de l'en-tête et des constantes.
- Lecture des variables.

Appel : `read_osc` est appelée par divers programmes de dessin, en particulier `des_osc`, `des_abon`, `des_diff_osc` ou encore `lit_osc`.

FIGURE 8.4 – Organigramme de la routine de la gestion de la résolution des équations de la structure interne `resout`.

SUBROUTINE `read_ascii(nom_fich,itot,nglob,nvar,abid)`

- Entrée
 - `nom_fich` : nom du fichier de type `mon_modele.osc` (sans l'extension `.osc`).
- Sorties :
 - `itot`, `nglob`, `nvar`, `abid` : nombre de points, de "constantes global", de variables du tableau `var`, sans les éléments chimiques,

La description des quantités extraites de ces fichiers est donnée au § D.1 (Page 267).

8.116 Routine `resout`

La fonction de la routine PRIVATE `resout` du module `mod_static`, *cf.* § E.12 (Page 295), est la gestion de la résolution des équations de l'évolution stellaire pour :

1. Le modèle en cours d'évolution.
2. Le modèle de séquence principale d'âge zéro homogène.
3. Le modèle initial de pré-séquence principale homogène.

La figure 8.4 (Page 223), présente un organigramme de la routine `resout`.

Description : L'architecture des algorithmes de résolution des équations de la structure interne est assez similaire dans les trois phases de l'évolution évoquées ci-avant. La structure de chacune d'entre elles est celle d'une itération Newton-Raphson.

- Initialisation de divers paramètres généraux.
- Poursuite d'une évolution :
 - Ajustement de paramètres s'il s'agit du calcul du modèle ultime.
 - Appel à `update`, *cf.* § 8.134 (Page 238) pour mise en mémoire du modèle au pas temporel précédent.
 - Initialisation de compteurs et début de l'itération Newton-Raphson.
 - Calcul de la perte de masse.
 - Estimation du nombre de couches nécessaire pour assurer à la constante de répartition la valeur définie par la précision requise, *cf.* § 7.2.4 (Page 87).
 - Appel à `lim_zc`, *cf.* § 8.70 (Page 202), pour un réaménagement éventuel de la discrétisation spatiale, la détection des limites zones radiatives / zones convectives et l'ajustement des facteurs de répartition, *cf.* § 7.2.5 (Page 89).
 - Evolution de la composition chimique, éventuellement du moment cinétique, par appel à `evol`, *cf.* § 8.54 (Page 195).
 - Formation et résolution des équations de l'équilibre quasi-statique par appel à `coll_qs`, *cf.* § 8.16 (Page 177).
 - Suivant l'estimation de la précision :
 - Ajustement du pas temporel pour satisfaire les conditions d'arrêt particulières, réinitialisation et reprise des itérations Newton-Raphson.
 - Poursuite des itérations Newton-Raphson.
 - Réinitialisation avec diminution du pas temporel et reprise.

- Gestion éventuelle de la rotation solide, estimation de l'énergie graviphique, tabulation de diverses quantités, retour vers `cesam`, *cf.* § 8.10 (Page 173).
- Appel à `sortie`, *cf.* § 8.124 (Page 231) pour arrêt du calcul.
- Modèle initial de séquence principale d'âge zéro homogène, ou de pré-séquence principale homogène :
 - Estimation du nombre de couches nécessaires pour assurer à la constante de répartition la valeur définie par la précision requise, *cf.* § 7.2.4 (Page 87).
 - Appel à `lim_zc`, *cf.* § 8.70 (Page 202), pour un réaménagement éventuel de la discrétisation spatiale, la détection des limites zones radiatives / zones convectives et l'ajustement des facteurs de répartition, *cf.* § 7.2.5 (Page 89).
 - Initialisation de compteurs et début de l'itération Newton-Raphson.
 - Formation et résolution des équations de l'équilibre quasi-statique par appel à `coll_qs`, *cf.* § 8.16 (Page 177).
 - Suivant l'estimation de la précision :
 - Poursuite des itérations Newton-Raphson, précédée par une réactualisation de la localisation des limites zones radiatives / zones convectives par appel à `lim_zc`.
 - Gestion éventuelle de la rotation solide, estimation de l'énergie graviphique, tabulation de diverses quantités, retour vers `cesam`, *cf.* § 8.10 (Page 173).
 - Appel à `sortie`, *cf.* § 8.124 (Page 231), pour arrêt du calcul.

Appel : à diverses reprises `resout` est appelée par `cesam`, *cf.* § 8.10 (Page 173).

SUBROUTINE `resout(un23,dt,dts)`

- Entrées :
 - `un23,dt`, : Paramètre du type de modèle, pas temporel.
- Sorties :
 - `dts` : Estimation de la valeur du pas temporel suivant.

8.117 Routine `resout_chim`

Cette routine PRIVATE du module `mod_evol`, *cf.* § E.11 (Page 294), a pour fonction la résolution par éléments finis du système implicite des équations de la diffusion des espèces chimiques, *cf.* § 7.8 (Page 113). Les produits scalaires sont approchés par une intégration numérique de type Gauss, suivant un algorithme inspiré de l'algorithme 5.22 p.203 de Schumaker Schumaker (alg. 5.22 1981, p.203). De structure identique à `resout_rota`, *cf.* § 8.118 (Page 226), ces deux routines pourraient être rassemblées en une seule, elles sont distinctes pour faciliter la lisibilité.

Description : L'architecture globale de la routine est celle d'une itération Newton-Raphson. Les coefficients provisoires des B-splines, *i.e.* les inconnues, sont initialisés dans la routine d'appel `diffus`, *cf.* § 8.36 (Page 186).

- Initialisations au premier appel.
- Définition du rang du système, du nombre de lignes, des indices de première colonne, initialisation de compteurs.

- Itérations Newton-Raphson :
 - Pour chaque intervalle du vecteur nodal, détermination des abscisses de Gauss, formation des coefficients des équations par appel à `eq_diff_chim`, *cf.* § 8.44 (Page 190).
 - Pour chaque B-spline non identiquement nulle sur l'intervalle, évaluation du second membre du système linéaire et de sa contribution au jacobien.
 - Résolution du système linéaire bande, correction des coefficients provisoires des B-splines, Estimation de la précision.
- Suivant la précision obtenue, poursuite des itérations Newton-Raphson, retour vers la routine appelante, réinitialisation si la précision requise n'a pas été atteinte après un nombre fixé d'itérations.

Appel : `resout_chim` est appelée par `diffus`, *cf.* § 8.36 (Page 186).

SUBROUTINE `resout_chim(dt,ok)`

- Entrées :
 - `dt` : Pas temporel.
- Sorties :
 - `ok=.TRUE.` : La convergence a été obtenue.

8.118 Routine `resout_rota`

Cette routine générique PRIVATE du module `mod_evo1`, *cf.* § E.11 (Page 294), a pour fonction le choix entre les deux formalismes de diffusion du moment cinétique implanté dans `CESAM2k`.

Description : suivant la valeur de la variable `Krot`, il y a choix entre le formalisme de Talon et al. (1997) ou celui de Mathis & Zahn (2004).

Appel : `resout_rota` est appelée par `diffus`, *cf.* § 8.36 (Page 186).

SUBROUTINE `resout_rota(dt,ok)`

- Entrées :
 - `dt` : Pas temporel.
- Sorties :
 - `ok=.TRUE.` : La convergence a été obtenue.

8.119 Routine `resout_rota3/4`

Ces routine PRIVATE du module `mod_evo1`, *cf.* § E.11 (Page 294), ont pour fonctions la résolution par éléments finis du système implicite des équations de la diffusion du moment cinétique suivant les formulations de Talon et al. (1997); Mathis & Zahn (2004), *cf.* § 7.9 (Page 129). Les produits scalaires sont approchés par une intégration numérique de type Gauss, suivant un algorithme inspiré de Schumaker (Alg. 5.22 1981, p. 203). De structure identique à `resout_chim`, *cf.* § 8.117 (Page 225), ces routines pourraient être rassemblées en une seule, elles sont distinctes pour faciliter leur lisibilité.

Description : L'architecture globale de la routine est celle d'une itération Newton-Raphson. Les coefficients provisoires des B-splines, *i.e.* les inconnues, sont initialisés dans la routine d'appel `diffus`, *cf.* § 8.36 (Page 186).

- Initialisations au premier appel.
- Définition du rang du système, du nombre de lignes, des indices de première colonne, initialisation de compteurs.
- Itérations Newton-Raphson :
 - Pour chaque intervalle du vecteur nodal, détermination des abscisses de Gauss, formation des coefficients des équations par appel à `eq_diff_rota`, *cf.* § 8.46 (Page 191).
 - Pour chaque B-spline non identiquement nulle sur l'intervalle, évaluation du second membre du système linéaire et de sa contribution au jacobien.
 - Pour le centre, formation des coefficients des parties intégrées par appel à `eq_diff_rota3/4`, *cf.* § 8.46 (Page 191), et calcul des contributions au second membre et au jacobien du système linéaire.
 - Pour la surface, formation des coefficients des parties intégrées par appel à `eq_diff_rota3/4`, *cf.* § 8.46 (Page 191), et calcul des contributions au second membre et au jacobien du système linéaire.
 - Résolution du système linéaire bande, correction des coefficients provisoires des B-splines, estimation de la précision.
- Suivant la précision obtenue, poursuite des itérations Newton-Raphson, retour vers la routine appelante, réinitialisation si la précision requise n'a pas été atteinte après un nombre fixé d'itérations.

Appel : `resout_rota3/4` sont appelées par `resout_rota` et ont les mêmes arguments.

8.120 Routine `rkimps`

Cette routine PRIVATE du module `mod_evo1`, *cf.* § E.11 (Page 294), a pour fonction l'intégration des équations de l'évolution des espèces chimiques *e.g.* Eq. 7.114 (Page 162) en l'absence de diffusion. Les échelles de temps d'évolution des divers isotopes diffèrent par plusieurs dizaines de magnitudes. A titre d'exemple, dans le soleil, l'échelle de temps d'évolution de ^2H est de quelques minutes, alors que celle de ^1H est de l'ordre de la centaine de million d'années. La disparité de ces échelles de temps est à l'origine de difficultés numériques : si on s'impose de suivre la plus courte échelle de temps, le temps de calcul est inimaginable, si on s'impose de suivre la plus grande échelle de temps, les erreurs numériques engendrées sur les espèces chimiques avec courte échelle de temps font "exploser" le calcul.

Pour ce type de problème numérique, qualifié de "raide", ou encore de "mal posé", il existe des algorithmes spécifiques qui, *en gros*, permettent de suivre l'échelle de temps que l'on désire, *e.g.* celle de ^1H pour le soleil, sans que, pour autant, les erreurs numériques engendrées sur les espèces chimiques à courte échelle de temps ne perturbent la solution de façon catastrophique. Ces algorithmes sont implicites, ils nécessitent le calcul d'un Jacobien ; en cas de non linéarité, la solution est obtenue par approximations successives, le plus souvent par la méthode de Newton-Raphson, qu'il convient parfois d'aménager.

Dans le cas de la structure interne, le mélange convectif complique encore le problème qui, non seulement est raide, mais est donc aussi intégral. En résumé, le problème du calcul de l'évolution des espèces chimiques est un problème intégro-différentiel aux valeurs initiales, non linéaire, raide.

La routine `rkimps` utilise le formalisme de Runge-Kutta implicite avec le schéma Lobatto IIIC, *cf.* Hairer & Wanner (1991), ainsi qu'il a été décrit au §7.5.3 (Page 104), noté dans la suite IRK Lobatto IIIC. L'ordre d'intégration est limité au troisième ordre. L'utilisation d'un ordre supérieur nécessiterait l'interpolation de la température, de la densité et de la composition chimique en tous les points du modèle entre différents pas temporel, interpolations dans lesquelles serait perdu le gain en précision que l'on peut attendre de l'utilisation d'un schéma d'ordre élevé. Cette routine originale est l'une des plus "techniques" et élégantes de CESAM2k.

Choix du pas temporel :

La valeur du pas temporel à utiliser dépend de la précision requise. A l'issue de chaque intégration, la précision obtenue est estimée. Dans le programme appelant `evol`, *cf.* §8.54 (Page 195), le pas temporel est alors diminué (*respt.* augmenté) si la précision obtenue est inférieure (*respt.* supérieure) à ce qui est souhaité. La précision d'une intégration peut être estimée en l'effectuant avec des algorithmes différents, ou des pas temporels différents, ce qui nécessite de refaire plusieurs fois le calcul. Pour limiter l'effort de calcul, avec `rkimps` on se contente de contrôler le pas temporel en limitant les variations de certains isotopes au cours d'un pas temporel. Cette méthode, quoique très répandue, n'est pas correcte car une grande (*respt.* petite) variation d'une abondance ne signifie pas, ***nécessairement***, que l'intégration est imprécise (*respt.* précise).

Description :

- Initialisations des matrices de IRK Lobatto IIIC suivant l'ordre de précision, **ordre**, défini dans la routine `cesam`, *cf.* §8.10 (Page 173), des contraintes du schéma de Newton-Raphson et de l'ordre de grandeur de chaque abondance utilisée pour estimer la précision relative de l'intégration et de la convergence.
- Pour chaque temps intermédiaire, appel à la routine de réactions thermonucléaires pour le calcul des Ψ_i , éventuellement appel à la routine `vent`, *cf.* §4.5 (Page 56), pour tenir compte de la perte ou du gain de masse, calcul de la contribution à la ligne correspondante du jacobien, détermination de la contribution à l'intégrale de Gauss dans les zones convectives.
- Formation de la ligne correspondante de Eq. ?? (Page ??).
- Solution du système linéaire donnant les corrections du processus itératif.
- Estimation de la précision et des nouvelles abondances, et poursuite, si besoin, du processus itératif.

Arguments de `rkimps`

SUBROUTINE `rkimps(t_t,ro_t,comp,t,ro,compy,dt,esti,ok,nuc,kk,z_vent,dm)`

- Entrées :
 - `t_t, ro_t, t, ro` : tables des températures, densité aux temps t et $t + dt$;
 - `dt, kk, dm, compx, z_vent` : pas temporel, nombre de couches à mélanger, tables des intervalles de masse et de la composition chimique au temps t , variable logique indiquant que l'on doit corriger les taux de

Appel :

SUBROUTINE `rq_reac(comp,t,ro,r,drt,dro,drx,q,dqt,dqo,dqx,mue,dmuedx)`

- Entrées :
 - `comp, t, ro` : Composition chimique par volume, température, densité.
- Sorties :
 - `r, drt, dro, drx, q, dqt, dqo, dqx, mue, dmuedx` : Taux, énergie des réactions et dérivées par rapport à la température, densité, composition chimique, nombre d'électrons e^- par volume, et dérivées par rapport à la composition chimique.

8.123 Routine saha

Cette routine PUBLIC du module `mod_etat`, cf. §E.5 (Page 290), résout l'équation de Saha (Cox & Giuli, 1968, eq. 5.30) pour la détermination des taux d'ionisation des divers éléments chimiques. On limite les fonctions de partition aux poids statistiques des niveaux fondamentaux. La formulation est décrite au §7.8.4 (Page 118). Les équations, écrites sous la forme :

$$\rho N_0 \sum_i \bar{z}_i(n_e) x_i - n_e = 0$$

sont résolues par itération Newton-Raphson.

Description :

- Initialisations au premier appel :
 - On admet que les taux d'ionisation sont identiques pour tous les isotopes d'une même espèce chimique. Les différents isotopes d'une même espèce chimique sont regroupés dans un seul élément auquel l'abondance totale de l'espèce est affectée. `CESAM2k` signale si un des éléments utilisés est inconnu, le déroulement du calcul est alors arrêté.
 - Initialisation des potentiels d'ionisation et des poids statistiques pour les espèces identifiées par leur charge. La fonction de Fermi-Dirac $F_{1/2}$ est ensuite tabulée en fonction du paramètre de dégénérescence.
- Pour les appels suivants :
 - Initialisation des abondances de chaque espèce et du nombre d'électrons libres.
 - Calcul du facteur de dégénérescence et résolution des équations de Saha pour chaque ion, redétermination du nombre d'électrons libres. Itération Newton-Raphson sur le nombre d'électrons libres.
 - Détermination de la charge moyenne de chaque ion.
 - Après convergence : Restitution des taux d'ionisation et de la charge moyenne pour chaque isotope.

Appel : `saha` est appelée par `cesam, coeff_rota, diffm_br`.

SUBROUTINE `saha(xchim,t,ro,ioni,z_bar,nel,eta)`

- Entrées :
 - `xchim`, `t`, `ro` : composition chimique par volume, température, densité;
- Sorties :
 - `ioni`, `z_bar`, `nel`, `eta` : taux d'ionisation et charge moyenne de chaque ion, nombre d'électrons libres, facteur de dégénérescence.

8.124 Routine sortie

La routine PUBLIC `sortie` du module `mod_variables`, cf. § E.4 (Page 286), est sans argument. Elle est appelée lorsque CESAM2k détecte une anomalie fatale au bon déroulement du calcul et à laquelle il ne peut remédier, par exemple une sortie de table. `sortie` ferme les unités d'écriture et de dessin avant l'arrêt des calculs.

8.125 Routine générique static

Cette routine PRIVATE du module `mod_static`, cf. § E.12 (Page 295), constitue la *routine générique* du calcul des coefficients des équations de l'équilibre quasi-statique dans l'espace physique.

Description :

`static` oriente le calcul des coefficients des équations d'équilibre quasi-statique vers leurs formes lagrangienne ou eulérienne, cf. § 7.2.4 (Page 87).

Appel :

SUBROUTINE `static(fait,cx,li,y,be,ae,compt,dt,repren,ip)`

- Entrées :
 - `fait=1` : Calcul des résidus des équations au point de collocation.
 - `fait=2` : Calcul du résidu pour la condition au point limite.
 - `cx` : Indice du point de collocation.
 - `li` : Numéro de la limite.
 - `y` : variables et dérivées au point de collocation ou limite.
 - `compt` : Compteur du nombre d'itérations Newton-Raphson globales.
 - `dt` : Pas temporel.
 - `ip` : Indice du facteur de répartition.
- Sorties :
 - `be` : Résidus.
 - `ae` : Eléments du jacobien.
 - `repren=.TRUE.` : La variation relative d'énergie graviphique est trop forte, le calcul devra être réinitialisé.

8.126 Routines static_m, static_r

Ces deux routines PRIVATE du module `mod_static`, cf. § E.12 (Page 295), sont équivalentes, elles forment les coefficients des équations de l'équilibre quasi-statique, sous leur forme lagrangienne et eulérienne respectivement, cf. § E.12 (Page 295).

Pour chaque point de collocation, les équations de la structure interne sont écrites sous la forme $f(x) = 0$, ainsi que les conditions limites. On utilise les variables retenues pour l'intégration numérique. Ces équations ont été détaillées § 7.2 (Page 82). Bien que, formellement, la variable indépendante soit la masse (lagrangien), dans `static_m` et le rayon (eulérien) dans `static_r`, les équations sont écrites pour "l'indice réel de couche" q , cf. § 7.2 (Page 82).

Les singularités centrales des équations de la structure interne écrites sous la forme Eq. 7.2 (Page 83) disparaissent sous la forme Eq. ?? (Page ??) utilisée par `CE-SAM2k`. Par ailleurs, le centre n'étant pas un point de collocation, il n'y a pas lieu d'introduire explicitement les formes limites des équations. Au centre, les variables ont pour valeurs limites celles, *sans singularité*, des polynômes par morceaux qui approchent la solution, ce qui constitue un des avantages de la méthode.

Pour les limites externes, les fonctions qui dépendent de la physique utilisée : $P_b(L, R)$, $P_{\text{gaz } b}(L, R)$, $T_b(L, R)$ et $M_b(L, R)$ sont calculées, au point $q = n$ par une des routines de physique de type `lim_atm`, cf. § 8.68 (Page 201).

Description :

Après quelques initialisations, on détermine la fonction de répartition et ses dérivées, puis on forme les équations et leurs dérivées par rapport à toutes les variables dépendantes. Les équations pour les conditions limites sont écrites à la fin de la routine. On a conservé dans la source des instructions permettant des vérifications, en particulier celles des coefficients du jacobien.

- Initialisations de quantités invariantes.
- Pour chaque point de collocation :
 - Extraction des variables physiques et vérification que le rayon, la masse et la luminosité⁵, ont des valeurs positives.
 - Extraction de la composition chimique locale, initialisation de quantités liées à la rotation, à la pression turbulente.
 - Calcul par un appel à `thermo`, cf. § 8.131 (Page 236), des valeurs locales de grandeurs thermodynamiques, de l'opacité, de l'énergie nucléaire, des gradients et de leurs dérivées, transformations de ces grandeurs des unités cgs. en unités locales avec des aménagements algorithmiques.
 - Calcul de l'énergie gravifique :
 - Pour un nouveau pas temporel, à la première itération globale gérée par `resout`, cf. § 8.116 (Page 223), *i.e.* `new=.TRUE.`, la valeur est interpolée, en masse, à partir de ses valeurs au pas temporel précédent.
 - Pour les itérations suivantes, cette quantité est calculée avec l'alternative de l'approximation de Kippenhahn cf. § 7.4.1 (Page 99) ou du calcul complet cf. § 7.4.2 (Page 100). Au cas où, à la même abscisse lagrangienne, un écart relatif trop important de pression ou de température est observé avec le pas temporel précédent cf, § 6.2 (Page 70), la variable logique `reprend=.TRUE.` et les calculs sont réinitialisés par un retour à `resout`.
- Après quelques dispositions algorithmiques concernant le moment cinétique, détermination de la fonction de répartition et des dérivées, puis formulation des équations et de leurs dérivées partielles par rapport à chacune des variables.

5. Pour `static_m`

- Ensemble d'instructions permettant de tester numériquement l'exactitude des dérivées.
- Pour chaque point limite :
 - Ecriture des conditions limites au centre *i.e.* `li=1, 2, 3`.
 - Ecriture des conditions limites à l'extérieur *i.e.* `li=4, 5, 6`. Pour limiter le nombre de calculs de l'atmosphère, `lim_atm`, *cf.* § 8.68 (Page 201), n'est appelée qu'une seule fois, les grandeurs alors calculées étant utilisées pour formuler les autres conditions externes. Il suffit d'enlever les `c` de commentaires pour rendre effectifs des tests de vérification des dérivées.

Appel : Ces routines appelées par la routine générique `static`, *cf.* § 8.125 (Page 231), utilisent les mêmes arguments.

8.127 Routine tabul_nuc

La fonction de cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est la gestion de la tabulation des réactions nucléaires.

Description :

- Pour chaque réseau de réactions nucléaires :
 - Initialisation du nombre de réactions, du nombre d'isotopes utilisés par les réactions, des indices de ^4He et de la réaction 3α .
 - Initialisation des indices des isotopes utilisés dans les réactions.
 - Définition de l'ordre des réactions et du domaine de tabulation.
- Sélection du type de compilation à utiliser.
- Appel à `taux_nuc`, *cf.* § 8.129 (Page 234), pour le calcul des taux de *toutes* les réactions implémentées.
- Extraction des taux des réactions relevant du réseau retenu et tabulation.

Appel : cette routine d'initialisation `tabul_nuc` est appelée par `rq_reac`, *cf.* § 8.122 (Page 229), en début de calcul. Cette routine n'a que des arguments de sortie.

SUBROUTINE `tabul_nuc(ar, i3al, knot_temp, m_temp,`
`1 nom_reac, n_temp, q0, taux_reac, temp, ttemp)`

- `nom_reac, i3al` : Noms des réactions du réseau utilisé et indice de la réaction 3α .
- `ar, q0, taux_reac` : Tabulation des masses réduites, des énergies, des taux de réaction.
- `knot_temp, m_temp, n_temp, temp, ttemp` : Eléments du vecteur nodal de la tabulation des taux de réaction.

8.128 Routine taueff

Cette routine PRIVATE du module `mod_atm`, *cf.* § E.8 (Page 291), a pour fonction la détermination de la profondeur optique τ_* correspondant au rayon bolométrique du modèle R_* . τ_* vérifie l'équation implicite :

$$T_{\text{eff}} = \mathcal{T}(\tau_*, T_{\text{eff}}, g).$$

Description :

- Initialisation $\tau_\star = 2/3$.
- Détermination de τ_\star par l'algorithme de Newton-Raphson avec appel à la routine de la loi $T(\tau)$ concernée, *cf.* § 8.130 (Page 235).

Appel : `taueff` est appelée par `lim_atm`, *cf.* § 8.68 (Page 201).

SUBROUTINE `taueff(teff,grav,tau)`

- Entrées :
 - `teff`, `grav` : température effective, gravité.
- Sorties :
 - `tau` : τ_\star .

8.129 Routine `taux_nuc`

La fonction de cette routine PRIVATE du module `mod_nuc`, *cf.* § E.9 (Page 292), est l'initialisation des paramètres nucléaires, le calcul des énergies et des taux d'un grand nombre de réactions. Il est évidemment possible d'implémenter de nouvelles réactions, la procédure à suivre en est détaillée au § 5.1 (Page 67).

Description :

Après une initialisation, les taux de réactions sont dérivés des formules de diverses compilations. C'est la nécessité du calcul des dérivées qui a motivé la tabulation des taux de réactions nucléaires plutôt que l'emploi direct des formules.

- Au premier appel :
 - Initialisations diverses.
 - Initialisation des excès de masse.
 - Initialisation des masses atomiques.
 - Initialisation des charges des isotopes.
 - Initialisation des noms des isotopes.
 - Initialisation des noms des réactions, des masses réduites, des énergies, des charges des noyaux de la réaction.
- Initialisations diverses reliées à la température.
- Calcul des taux de *toutes* les réactions, suivant la compilation retenue.

Appel : `taux_nuc` est appelée par `tabul_nuc`, *cf.* § 8.127 (Page 233).

SUBROUTINE `taux_nuc(t,total,rt,zit,izzt,qt,nom_react,nucleot,
1 at,nom_element)`

- Entrées :
 - `t`, `total` : température, indice du choix de la compilation à utiliser, *cf.* § 3.4 (Page 26).
- Sorties :
 - `rt` : $\ln(\text{taux des réactions})$.
 - `zit` : Charges des noyaux.
 - `izzt` : Charges des noyaux de la réaction.
 - `qt` : Energies.
 - `nom_react` : Symboles des réactions.

- `nucleot` : Masses des isotopes .
- `at` : Masses réduites.
- `nom_elem` : Noms des isotopes.

8.130 Routine générique `tdetau`

Cette routine PUBLIC du module `mod_atm`, cf. § E.8 (Page 291), constitue la routine générique de la gestion du calcul de la température en fonction de la profondeur optique.

Description :

Le calcul est orienté vers la routine de loi $T(\tau)$ indiquée dans le fichier de données `mon_modele.don`, cf. § 3.4 (Page 26).

Appel : Les lois $T(\tau)$ étant utilisées pour reconstruire une atmosphère, la routine `tdetau` est appelée de `lim_atm`, cf. § 8.68 (Page 201).

```
SUBROUTINE tdetau(tau,teff,grav,t,dtsdtau,dtsdteff,dtsdg,
1 ro_ext,dro_grav,dro_teff,f_tau,df_tau,d2f_tau)
```

- Entrées :
 - `tau`, `teff`, `grav` : profondeur optique, température effective, gravité.
- Sorties :
 - `t`, `dtsdtau`, `dtsdteff`, `dtsdg` : température et dérivées ;
 - `ro_ext`, `dro_grav`, `dro_teff` : densité externe et dérivées ;
 - `f_tau`, `df_tau`, `d2f_tau` : fonction f et dérivées cf. § 7.3.3 (Page 94).

8.131 Routine thermo

Cette routine PRIVATE du module `mod_static`, cf. § E.12 (Page 295), effectue le calcul des variables thermodynamiques au sens large, à l'aide d'appels aux diverses routines génériques.

Description

- Initialisations effectuées lors du premier appel.
- Appel à l'équation d'état pour le calcul des grandeurs thermodynamiques.
- Détermination de l'énergie thermonucléaire.
- Détermination de l'opacité.
- Calcul du gradient radiatif.
- Formation du critère de convection (Schwarzschild ou Ledoux).
- Affectation du gradient adiabatique ou radiatif dans les extensions des zone convective, cf. § 3.9 (Page 37).
- Pour les zones convectives, calcul du gradient.

Appel : `thermo` est appelée de divers endroits en particulier par `lim_zc`, cf. § 8.70 (Page 202), `static_m`, `static_r`, cf. § 8.126 (Page 232).

```
SUBROUTINE thermo(pt,p,t,m,l,r,dlpp,xchim,dxchim,
1 ro,drop,drot,drox,u,dup,dut,dux,grad,dgradpt,dgradp,dgradt,
```

2 dgradx, dgradm, dgradl, dgradr, dgradlpp,
 3 gam, dgampt, dgamp, dgamt, dgamx, dgamm, dgaml, dgamr, dgamlpp,
 4 epsilon, dejsp, depst, dejsx, kap, dkapp, dkapt, dkapx,
 5 delta, deltap, deltat, deltax, cp, dcpp, dcpt, dcp, dcp, dcp,
 6 gradad, dgradadp, dgradadt, dgradadx,
 7 hp, dhppt, dhpp, dhpt, dhpx, dhpr, dhpm,
 8 gradrad, alfa, beta, gamma1, radiatif)

- Entrées :
 - pt, p, t, m, l, r : Pression totale, pression gazeuse, température, masse, luminosité, rayon.
 - xchim, dxchim, dlpp : Composition chimique par mole, dérivée / $(M/M_{\odot}), \frac{d \ln P_{\text{gaz}}}{d \ln P_{\text{tot}}}$.
- Sorties :
 - ro, drop, drot, drox : densité et dérivées ;
 - u, dup, dut, dux : énergie interne et dérivées ;
 - grad, dgradpt, dgradp, dgradt, dgradx, dgradm, dgradl, dgradr, dgradlpp : gradient de température $\nabla \equiv \frac{\partial \ln T}{\partial \ln P_{\text{tot}}}$ et dérivées ;
 - gam, dgampt, dgamp, dgamt, dgamx, dgamm, dgaml, dgamr, dgamlpp : efficacité de la convection et dérivées ;
 - epsilon, dejsp, depst, dejsx : énergie nucléaire et gravifique et dérivées ;
 - kap, dkapp, dkapt, dkapx : opacité et dérivées ;
 - delta, deltap, deltat, deltax : δ et dérivées ;
 - cp, dcpp, dcpt, dcp, dcp, dcp : c_p et dérivées ;
 - gradad, dgradadp, dgradadt, dgradadx : gradient adiabatique et dérivées ;
 - hp, dhppt, dhpp, dhpt, dhpx, dhpr, dhpm : échelle de hauteur de pression et dérivées ;
 - gradrad, alfa, beta, gamma1, radiatif : gradient radiatif, α, β, Γ_1 , radiatif=.TRUE. : on est dans une zone radiative.

8.132 Routine thermo_atm

Pour la restitution de l'atmosphère, cette routine PRIVATE du module mod_atm, cf. § E.8 (Page 291), est l'homologue de la précédente. Elle n'en diffère que par :

- l'absence d'appel aux taux des réactions thermonucléaires,
- le calcul du gradient de température qui tient compte de la loi $T(\tau)$ utilisée pour la restitution de l'atmosphère cf. § 7.3.2 (Page 92),
- la composition chimique, la luminosité et la vitesse angulaire y sont spatialement constantes.

Description :

- Initialisations effectuées lors du premier appel.
- Appel à l'équation d'état pour le calcul des grandeurs thermodynamiques.
- Détermination de l'opacité.
- Calcul du gradient radiatif.

- Formation du critère de convection (Schwarzschild ou Ledoux).
- Pour les zones convectives, calcul du gradient.

Appel : thermo_atm est appelée par eq_atm, cf. § 8.43 (Page 189).

```
SUBROUTINE thermo_atm(pt,p,t,xchim,m,l,r,dlpp,
1 tau,df_tau,d2f_tau,rstar,ro,drop,drot,kap,dkapp,dkapt,gradad,
2 dgradadp,dgradadt,grad,dgradpt,dgradp,dgradt,dgradr,dgradrs,
3 dgradm,dgradtau,dgradlpp,gam,dgampt,dgamp,dgamr,dgamrs,
4 dgamm,dgamtau,dgamlpp,hp,dhppt,dhpp,dhpt,dhpr,dhpm,delta,deltap,
5 deltat,cp,gradrad,alfa,beta,gamma1,radiatif,deriv)
```

- Entrées :
 - pt, p, t, m, l, r : pression totale, pression gazeuse, température, masse, luminosité, rayon,
 - xchim, dlpp : composition chimique par mole, $\frac{d \ln P_{\text{gaz}}}{d \ln P_{\text{tot}}}$;
 - tau, df_tau, d2f_tau, rstar : profondeur optique, partie dépendante de τ de la loi $T(\tau)$ et dérivées premières et secondes, rayon total ;
 - deriv=.TRUE. : le calcul des dérivées est requis.
- Sorties :
 - ro, drop, drot : densité et dérivées ;
 - grad, dgradpt, dgradp, dgradt, dgradr, dgradrs, dgradm, dgradtau, dgradr, dgradlpp : gradient de température $\nabla \equiv \frac{\partial \ln T}{\partial \ln P_{\text{tot}}}$ et dérivées ;
 - gam, dgampt, dgamp, dgamr, dgamrs, dgamm, dgamtau, dgamlpp : efficacité de la convection et dérivées ;
 - kap, dkapp, dkapt : opacité et dérivées ;
 - gradad, dgradadp, dgradadt : gradient adiabatique et dérivées ;
 - hp, dhppt, dhpp, dhpt, dhpr, dhpm : échelle de hauteur de pression et dérivées ;
 - delta, deltap, deltat : δ et dérivées ;
 - cp, gradrad, alfa, beta, gamma1, radiatif : c_p , gradient radiatif, α , β , Γ_1 , radiatif=.TRUE. : on est dans une zone radiative.

8.133 Routine trho

La fonction de cette routine PRIVATE du module mod_atm, cf. § E.8 (Page 291), est la lecture des données et l'interpolation en (T, ρ) des lois $T(\tau)$ de type roger**, cf. § 8.121 (Page 229).

Description : D'origine externe cette routine n'est pas décrite.

Appel : trho est appelée des routines de lois $T(\tau)$ du type roger**.

8.134 Routine update

Suivant la valeur .TRUE. / .FALSE. de la variable logique next, cette routine PRIVATE du module mod_static, cf. § E.12 (Page 295), permet, soit de translater

au temps t le modèle calculé pour le temps $t + dt$, soit de réinitialiser un modèle pour le temps $t + dt$ avec celui obtenu pour le temps t . L'extension `_t` permet de différencier les variables au temps t de celles au temps $t + dt$.

Description :

- Si `next=.TRUE.` les grandeurs caractérisant le modèle au temps $t + dt$ sont transcrites sur celles du modèle au temps t , le pas temporel est ajusté de façon à éventuellement arrêter le calcul sur l'âge maximum désiré.
- Si `next=.FALSE.` les grandeurs caractérisant le modèle au temps t sont transcrites sur celles du modèle au temps $t + dt$, le pas temporel est ajusté de façon à éventuellement arrêter le calcul sur l'âge maximum désiré.

Appel : `update` est appelée par `resout`, *cf.* §8.116 (Page 223).

SUBROUTINE `update(next,dt,dts)`

- Entrées :
 - `next=.TRUE.` : On écrase les variables sur celles du pas temporel précédent.
 - `next=.FALSE.` : On restitue les variables du pas temporel précédent.
- Entrées/Sorties :
 - `dt, dts` : Pas temporel, pas temporel estimé.

8.135 Routine vent

Lorsqu'il y a perte ou gain de masse, *CESAM2k* offre la possibilité que la composition chimique de la matière gagnée ou perdue diffère de celle des couches les plus externes *cf.* §7.8.1 (Page 115). Cette possibilité est gérée par la routine `PUBLIC vent` du module `mod_nuc`, *cf.* §E.9 (Page 292).

Description :

- Lors d'un appel d'initialisation, depuis la routine `evol`, on effectue une recherche dans l'environnement du fichier où sont indiquées les fractions de masse du vent, *cf.* §4.5 (Page 56). Si un tel fichier est trouvé, les fractions de masse sont normalisées, puis réparties entre les différents isotopes du réseau nucléaire, et enfin transformées en fraction par mole ; la variable logique `l_vent` est déclarée `.TRUE.` Si le fichier est inconnu, la variable logique `l_vent` est déclarée `.FALSE.`, la composition chimique du vent sera celle de la couche la plus externe du modèle.
- Lors des appels suivants, une correction aux taux de réactions nucléaires est effectuée, *cf.* §7.8.1 (Page 115).

Appel : `vent` est appelée par :

- `evol`, *cf.* §8.54 (Page 195), pour initialisation et dans le cas où le vent a une composition chimique différente de celle de la couche la plus externe.
- `rkimps`, *cf.* §8.120 (Page 227) ou `eq_dif_chim`, *cf.* §8.36 (Page 186), à la suite d'un appel à la routine `nuc`, *cf.* §8.77 (Page 206).

Les arguments sont optionnels.

SUBROUTINE `vent(xchim,dxchim,jac)`

- Entrées :

- `xchim` : Composition chimique par mole.
- Entrées/Sorties :
- `dxchim`, `jac` : Taux de réactions nucléaires et jacobien.

8.136 Routine `write_nl`

Cette routine PUBLIC du module `mod_exploit`, *cf.* §E.14 (Page 296), a pour fonction la formation d'un fichier de données `mon_modele.don`, *cf.* §3.4 (Page 26).

Description : Les NAMELISTs sont écrites après l'ouverture du fichier.

Appel : `write_nl`, qui n'a pas d'argument, est utilisée par des programmes d'exploitation, *cf.* §3.18 (Page 50).

8.137 Package `z14xcotrin21`

Le package d'interpolation d'opacité `z14xcotrin21` de A.I.Boothroyd, est utilisé par `CESAM2k` pour le calcul de l'opacité, lorsque l'abondance en éléments lourds dépasse $Z > 0.1$. Les routines d'interpolations utilisent des tables du sous-directory `SUN_STAR_DATA`, *cf.* §1.3 (Page 3). Elles nécessitent une mise en forme qui est transparente pour l'utilisateur *cf.* §2.2.2 (Page 14).

La mise en œuvre du package `z14xcotrin21` a bénéficié d'une collaboration avec A.I.Boothroyd. En raison de la complexité de sa programmation, `z14xcotrin21` n'est pas inclus dans un module. Le package est compilé conjointement au module `mod_opa`, *cf.* §E.6 (Page 290).

Description :

En raison de leur complexité et de leur origine externe, les routines du package ne sont pas détaillées. Pour plus d'informations se reporter aux explications données en tête de la source de `z14xcotrin21`.

Appel :

Avec le package `z14xcotrin21` `CESAM2k` utilise la routine `opal_x_cno_fu` ou à défaut la routine `opal`. Ces routines sont appelées par `opa_opal2`, *cf.* §8.82 (Page 210). En raison de la complexité, les listes d'appel ne sont pas détaillées.

Chapitre 9

Les routines numériques

Ces routines sont des routines PUBLIC ou PRIVATE du module `mod_numerique`, cf. §E.2 (Page 277) ou encore `mod_exploit`, cf. §E.14 (Page 296). Elles relèvent de l'analyse numérique et de l'algorithmique. Certaines routines sont originales, d'autres ont été construites en s'inspirant des descriptions données dans des ouvrages d'analyse numérique élémentaire. On n'en donne qu'une description sommaire. Ces routines sont regroupées dans le sous-directory `SOURCE`.

9.1 Routines numériques et assimilées

9.1.1 Routine `arb_rom`

Cette fonction transforme l'entier $i \in [0, 30]$ en notation arabe en notation romaine, Exemple : $21 \implies XXI$. La notation romaine comporte au plus 10 caractères, au delà, la sortie est CCC. Cette fonction est utilisée pour noter les niveaux d'ionisation.

9.1.2 Routine `boite`

Cette routine permet de dessiner une boîte d'erreur dont on donne les coordonnées du centre et la longueur des cotés.

9.1.3 Routine `box`

Cette routine permet de dessiner une boîte de dimensions $\pm\Delta x \times \Delta y$ autour du point de coordonnées (x, y) .

9.1.4 Routine `delete_doubles`

Cette routine permet de supprimer les éléments successifs identiques dans un tableau ALLOCATABLE ordonné par ordre croissant ; si besoin la dimension du tableau est réajustée.

9.1.5 Routine difdiv

Calcul des différences divisées pour interpolation par la formule de Newton *cf.* Stoer & Bulirsch (1979).

9.1.6 Routine fermi_dirac

Approximations analytiques des intégrales de Fermi-Dirac, reprises du package MHD, *cf.* § 8.52 (Page 194).

9.1.7 Routine gauss_band

Routine de résolution d'un système linéaire bande, dont la largeur de la bande est constante et comportant plusieurs seconds membres. L'algorithme est celui de l'élimination de Gauss avec équilibrage et pivot partiel. L'espace mémoire utilisé correspond à la partie non identiquement nulle du système.

9.1.8 Routine horner

La subroutine `horner` permet de calculer, pour $x = \alpha$, les dérivées de tous ordres d'un polynôme ainsi que le quotient de ce polynôme par $x - \alpha$ (algorithme de Hörner).

9.1.9 Routine intgauss

Initialisation des poids et des abscisses pour l'intégration de Gauss de divers ordres.

9.1.10 Routine matinv

Programme d'inversion de matrice. Utilise l'élimination de Gauss avec pivot total de `gauss_band`.

9.1.11 Routine max_local

Recherche des maxima de plusieurs tables dans un intervalle d'abscisses. Cette routine est adaptée au calcul des échelles pour des tracés.

9.1.12 Routine min_max

Recherche du maximum et du minimum d'une table. Cette routine est adaptée au calcul des échelles pour des tracés.

9.1.13 Routine min_max_cond

Recherche du maximum et du minimum d'une table sous condition. Cette routine est adaptée au calcul des échelles pour des tracés.

9.1.14 Routine neville

Algorithme de Neville pour interpolation polynômiale, *cf.* Stoer & Bulirsch (1979).

9.1.15 Routine newton

Interpolation polynômiale avec dérivées, par la formule de Newton, *cf.* Conte & de Boor (1987).

9.1.16 Routine pause

Pause avec commentaire et poursuite/arrêt.

9.1.17 Routine pgplot_factice

Ensemble des intitulés des routines du logiciel PGPLOT utilisées dans CESAM2*k*. Ces routines permettent d'exploiter le code sans que le logiciel de dessin soit implémenté, *cf.* § A.2.3 (Page 251). Cet ensemble est disposé à l'extérieur du MODULE `mod_cesam`.

9.1.18 Routine plot_rota

Routine subordonnée de `ecrit_rota.f`, *cf.* § 8.41 (Page 189) Routine de dessin "on line" des variables de la diffusion du moment cinétique.

9.1.19 Routine polyder

C'est une autre version de l'algorithme de Hörner dont les arguments diffèrent de ceux de `horner`. Pour des raisons historiques CESAM2*k* fait appel aux deux versions.

9.1.20 Routine shell

Routine tri d'un tableau optimisé pour F95.

9.1.21 Routine zoning

Permet de déterminer la répartition des abscisses de façon à ce que les incréments des ordonnées soient constants.

9.2 Routines spécifiques aux B-splines

Ces routines sont originales dans leur grande majorité. Les algorithmes utilisés sont, pour la plupart, inspirés de ceux décrits par De Boor (1978); Schumaker (1981).

9.2.1 Routine `bsp1ddn`

Généralise `bsp1dn` en calculant toutes les dérivées non identiquement nulles. Ne doit être utilisée que si des dérivées d'ordre supérieur à l'unité sont requises car elle demande plus de ressources que `bsp1dn`.

9.2.2 Routine `bsp1dn`

Interpolation au même point de n fonctions développées sur une base de B-splines; calcul du développement si nécessaire.

9.2.3 Routine `bsp_dis`

Routine de calcul des coefficients des B-splines pour n fonctions avec discontinuités pour interpolations ou lissage par Bézier.

9.2.4 Routine `bsp_gal`

Routine de calcul des coefficients des B-splines pour n fonctions avec discontinuités pour interpolations; utilise le formalisme intégral de Galerkin.

9.2.5 Routine `bval0`

Routine qui calcule la valeur de toutes les B-splines non identiquement nulles en un point fixé, cf. § 7.1.1 (Page 78).

9.2.6 Routine `bval1`

Routine qui calcule la valeur de toutes les B-splines non identiquement nulles et leurs dérivées premières en un point fixé. Demande plus de ressources que `bval0`.

9.2.7 Routine `bvald`

Routine qui calcule la valeur de toutes les B-splines non identiquement nulles ainsi que toutes leurs dérivées en un point fixé. Demande plus de ressources que `bval1`.

9.2.8 Routine `coll`

Routine qui génère les points de collocation de la base de De Boor (1978) pour l'intégration des équations différentielles.

9.2.9 Fonction `colpnt`

Fonction calculant l'abscisse du i -ième point de collocation dans l'intervalle $]x(l), x(l + 1)[$ pour la résolution d'une équation différentielle d'ordre r . `colpnt` est appelée par `coll`.

9.2.10 Routine `left_right`

Calcul des valeurs des n fonctions f et de leurs dérivées premières de part en un point quelconque d'une représentation par B-Spline. De part et d'autre d'un point du vecteur nodal ces valeurs peuvent différer, par exemple s'il y a une discontinuité.

9.2.11 Routine `linf`

Localisation d'un point dans une table d'abscisses en ordre strictement croissant, adapté aux B-splines.

9.2.12 Routine `newspl`

Effectue un changement de base de B-splines pour n polynômes par morceaux.

9.2.13 Routine `newspl_gal`

Effectue un changement de base de B-splines pour n polynômes par morceaux ; utilise le formalisme intégral de Galerkin.

9.2.14 Routine `noedif`

Routine d'initialisation du réseau de points de table de la base de De Boor (1978), pour la résolution des équations différentielles par collocation.

9.2.15 Routine `noein`

Détermine la séquence de nœuds de raccord pour une interpolation "optimale" suivant De Boor (1978).

9.2.16 Routine `noeu_dis`

Détermine la séquence de nœuds de raccord pour une interpolation "optimale" en tenant compte de discontinuités ; identique a `noein` s'il n'y a pas de discontinuité.

9.2.17 Routine `noeud`

Détermine la séquence de nœuds de raccord pour un vecteur des multiplicités donné.

9.2.18 Routine `schu58_n`

Calcul des valeurs en un point de n polynômes par morceaux à partir de leurs développements sur une base de B-splines et des valeurs des B-splines non identiquement nulles en ce point, *cf.* Schumaker (1981).

9.2.19 Routine `sum_n`

Calcule les intégrales de `n` splines entre des bornes données. Les coefficients doivent avoir été précédemment calculés par `bsp1dn` par exemple. Au premier appel, ces coefficients seront adaptés au calcul de l'intégrale, donc modifiés. Ils ne seront pas recalculés lors des appels ultérieurs. S'inspire de Schumaker (1981, alg. 5.19).

Troisième partie
APPENDICES

Annexe A

Installation sous LINUX

A.1 Création de la bibliothèque avec ifc

La Norme F95, telle qu'elle a été définie en 1996, est implémentée de façon stricte dans le compilateur¹ ifc. Depuis, on s'est aperçu d'une limitation "idiote" de cette Norme, interdisant de mettre un tableau ALLOCATABLE alors que :

```
REAL(kind=dp), INTENT(out), POINTER, DIMENSION(:)
```

était autorisé. Dans certains contextes, POINTER et ALLOCATABLE sont synonymes. Quelques temps plus tard, un Technical Report qui "corrigeait" et autorisait cette limitation est paru. Les compilateurs de Compaq/Digital (stations Unix, PC Windows) Lahey-Fujitsu (Linux, Windows) ont été corrigés ; INTEL semble n'avoir effectué la correction que dans la dernière version de ifc.

Remède : Remplacer ALLOCATABLE par POINTER, chaque fois que le compilateur manifeste. Par la suite, avec le compilateur ifc à jour, tout rentrera dans l'ordre. Actuellement la Norme Fortran 2003 différencie ALLOCATABLE et POINTER.

Les corrections à apporter sont les suivantes :

1. Routine tabul_nuc.f transformer :

```
REAL (kind=dp), INTENT(out), ALLOCATABLE, DIMENSION(:, :) :: taux_reac
REAL (kind=dp), INTENT(out), ALLOCATABLE, DIMENSION(:) :: ar, q0, temp, ttemp
INTEGER, INTENT(out) :: i3al, knot_temp, m_temp, n_temp
CHARACTER (len=20), INTENT(out), ALLOCATABLE, DIMENSION(:) :: nom_reac
```

en :

```
REAL (kind=dp), POINTER, DIMENSION(:, :) :: taux_reac
REAL (kind=dp), POINTER, DIMENSION(:) :: ar, q0, temp, ttemp
INTEGER, INTENT(out) :: i3al, knot_temp, m_temp, n_temp
CHARACTER (len=20), POINTER, DIMENSION(:) :: nom_reac
```

2. Routine tabul_nuc, transformer :

```
REAL (kind=dp), ALLOCATABLE, SAVE, DIMENSION(:, :) :: decrx, dzetax, taux_reac
REAL (kind=dp), ALLOCATABLE, SAVE, DIMENSION(:) :: ar, datx, dbidx,
1 dc1x, decro, decrt, dfx, df12x, dkx, dnex, dr, dzstarx, ecran,
2 lambda, mz1z2, q0, temp, ttemp, tx, z1z2
```

1. Diagnostique, commentaires et remèdes selon B.Pichon.

```

REAL (kind=dp), SAVE, DIMENSION(2) :: zeta, dzetat, dzetaro
REAL (kind=dp), SAVE :: cte1, cte2, cte3, cte4, cte5, cte6, cte7,
1 cte8, cte9, cte10
REAL (kind=dp) :: zstar, ne, c1, f12, at, k, bid, bid1, df,
1 e0, eps, beta, dnero, dc1ro, df12t, df12ro, datt, datro, dkt,
2 dkro, dbidt, dbidro, dft, dfro
INTEGER, SAVE :: i3al, knot_temp, l=1, m_temp, n_temp
INTEGER :: i, j
CHARACTER (len=20), ALLOCATABLE, DIMENSION(:) :: nom_reac
en :
REAL (kind=dp), POINTER, SAVE, DIMENSION(:, :) :: decrx, dzetax, taux_reac
REAL (kind=dp), POINTER, SAVE, DIMENSION(:) :: ar, datx, dbidx,
1 dc1x, decro, decrt, dfx, df12x, dkx, dnex, dr, dzstarx, ecran,
2 lambda, mz1z2, q0, temp, ttemp, tx, z1z2
REAL (kind=dp), SAVE, DIMENSION(2) :: zeta, dzetat, dzetaro
REAL (kind=dp), SAVE :: cte1, cte2, cte3, cte4, cte5, cte6, cte7,
1 cte8, cte9, cte10
REAL (kind=dp) :: zstar, ne, c1, f12, at, k, bid, bid1, df,
1 e0, eps, beta, dnero, dc1ro, df12t, df12ro, datt, datro, dkt,
2 dkro, dbidt, dbidro, dft, dfro
INTEGER, SAVE :: i3al, knot_temp, l=1, m_temp, n_temp
INTEGER :: i, j
CHARACTER (len=20), POINTER, DIMENSION(:) :: nom_reac

```

3. Routine, mod_donnees transformer :

```

CHARACTER (len=4), SAVE, PUBLIC, ALLOCATABLE, DIMENSION(:) :: nom_elem
en :
CHARACTER (len=4), SAVE, PUBLIC, POINTER, DIMENSION(:) :: nom_elem

```

A.2 Installation de PGPLOT

A.2.1 Installation intégrale avec ifc

Thierry Corbard, Observatoire de la Côte d'Azur, a construit le fichier de configuration `ifc_gcc.conf` de PGPLOT avec LINUX pour le compilateur F90 `ifc` de INTEL. Pour créer la source de PGPLOT en utilisant ce compilateur :

1. Se placer dans le sous-directory `/usr/local/src/pgplot/sys_linux` de la source de PGPLOT.
 - (a) Y inclure le fichier de configuration : `ifc_gcc.conf`, que l'on trouvera dans le sous-directory `SCRIPTS`.
 - (b) Y créer le sous-directory `ifc_src` : `mkdir ifc_src`.
 - (c) Y inclure le fichier `ifc_src/grgenv.f`, que l'on trouvera dans le sous-directory `SCRIPTS`.
2. Exécuter :

```

/usr/local/src/pgplot/makemake /usr/local/src/pgplot linux ifc_gcc

```


A.2.2 Installations sans larme

Sous LINUX avec les distributions dérivées de RedHat, Mandrake et SuSE en particulier, il est possible d'installer PGPLOT sans trop de difficultés en procédant de la façon suivante :

1. Avec un moteur de recherche, Google par exemple, rechercher `pgplot rpm`.
2. Parmi les sites détectés, choisir celui qui propose de télécharger une version récente correspondant au processeur et au système d'exploitation utilisés. Par exemple, `pgplot-5.2.2-1.i686.rpm` est satisfaisant pour un Pentium4 sous Mandrake².
3. Télécharger ce logiciel. A l'issue du téléchargement, se placer dans le directory où s'est effectué le téléchargement. Cliquer sur le nom du logiciel téléchargé, acquérir les droits du super-utilisateur, les logiciels s'installent.
4. Le système place les deux bibliothèques partageables `libpgplot.so.5.2.0` et `libpgplot.so.5` dans le directory `/usr/lib` et y crée le sous-directory `pgplot`. Il contient les 3 fichiers : `grexec.f`, `rgb.txt`, `grfont.dat`.
5. Créer dans le sous-directory `pgplot` deux liens vers les bibliothèques partageables : `ln -s /usr/lib/libpgplot.so.5 libpgplot.so`,
`ln -s /usr/lib/libpgplot.so.5.0 libpgplot.so.0`
6. Dans le fichier `.cshrc` ajouter :
 - (a) Le chemin de PGPLOT `LD_LIBRARY_PATH **** :/usr/lib/pgplot`.
 - (b) La variable d'environnement `setenv PGPLOT_DIR /usr/lib/pgplot`.

Avec les distributions DEBIAN, GNOME ou KUNBUTU, le logiciel s'installe automatiquement avec `apt-get -install pgplot5`³. Toutefois, certains logiciels, de traitement d'image par exemple, peuvent ne pas avoir été installés. Pour ce faire, la difficulté est d'en connaître les identificateurs, ceux-ci étant acquis, leur installation est immédiate avec `apt-get`. La localisation de la bibliothèque `libpgplot` s'obtient assez facilement, avec un `find` par exemple.

A.2.3 CESAM2k sans PGPLOT

Il est possible d'utiliser `CESAM2k` sans PGPLOT. 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. Evidemment, coder `NOM_DES='no_des'` et `NOM_DES_ROT='no_des'` dans les fichiers de données; il ne faut pas non plus chercher à linker avec la bibliothèque de PGPLOT; il convient aussi de ne pas indiquer de chemin à cette bibliothèque dans le fichier `cshrc`, ou ses équivalents.

A.3 Exemple de MAKEFILE

Le MAKEFILE suivant, créé par Th. Corbard O.C.A., est placé dans le sous-directory `EXPLOIT`. Les paramètres correspondent au compilateur `ifc` d'Intel et

2. On trouvera ce logiciel dans le sous-directory `SCRIPTS` de la distribution.

3. Vérifier que `PGPLOT5` est bien dans la liste des `SOURCEs` de `apt-get` (voir man `apt-get`).

les chemin à une implantation standard. Son utilisation est décrite au § 1.6.1 (Page 5).

```
RM=rm -f
```

```
AR=ar ruv
```

```
FC90 = ifort
```

```
mod_ext=mod #extension for module files
```

```
FFLAGS90 = -extend_source -O -xW -traceback -save -nbs
```

```
OPA_HOUDEK_PATH=../SUN_STAR_DATA/v9/lib
```

```
LIBLINK= -L. -lcesam2k -L$(OPA_HOUDEK_PATH) -lopint -lpgplot -lX11
```

```
#Adapt the following if you need to specify the location of X11 and pgplot lib
```

```
#PGPLOT_DIR=/usr/local/pgplot
```

```
#X11_DIR=/usr/lib64
```

```
#LIBLINK= -L. -lcesam2k -L$(PGPLOT_DIR) -lpgplot -L$(X11_DIR) -lX11
```

```
#Mis à jour pour SOURCE et EXPLOIT du 05.11.09, P.Morel
```

```
SRCS = \  
mod_kind.f\  
mod_numerique.f\  
mod_donnees.f\  
mod_variables.f\  
mod_etat.f\  
mod_opa.f\  
mod_conv.f\  
mod_atm.f\  
mod_nuc.f\  
mod_bp_for_alecian.f\  
mod_evol.f\  
mod_static.f\  
mod_cesam.f\  
mod_exploit.f
```

```
OBJS = \  
mod_kind.o\  
mod_numerique.o\  
mod_donnees.o\  
mod_variables.o\  
mod_etat.o\  
mod_opa.o\  
mod_conv.o
```

```

mod_atm.o\
mod_nuc.o\
mod_bp_for_alecian.o\
mod_evol.o\
mod_static.o\
mod_cesam.o\
mod_exploit.o

MODS = \
mod_kind.$(mod_ext)\
mod_numerique.$(mod_ext)\
mod_donnees.$(mod_ext)\
mod_variables.$(mod_ext)\
mod_etat.$(mod_ext)\
mod_opa.$(mod_ext)\
mod_conv.$(mod_ext)\
mod_atm.$(mod_ext)\
mod_nuc.$(mod_ext)\
mod_bp_for_alecian.$(mod_ext)\
mod_evol.$(mod_ext)\
mod_static.$(mod_ext)\
mod_cesam.$(mod_ext)\
mod_exploit.$(mod_ext)

cesam2k.out: cesam2k.f libcesam2k.a
$(FC90) $(FFLAGS90) -o cesam2k.out cesam2k.f $(LIBLINK)

libcesam2k.a: $(MODS)
$(AR) libcesam2k.a $(OBJS)
$(RM) $(OBJS)

mod_kind.o mod_kind.$(mod_ext): mod_kind.f
$(FC90) $(FFLAGS90) -c $*.f

mod_numerique.o mod_numerique.$(mod_ext): \
    arb_rom.f\
    boite.f \
    box.f \
    bsp1dn.f\
    bsp1ddn.f\
    bsp_dis.f\
    bsp_gal.f\
    bval0.f\
    bval1.f\
    bvald.f\
    coll.f\
    colpnt.f\

```

```
delete_doubles.f\  
difdiv.f\  
entre_d.f\  
entre_i.f\  
entre_s.f\  
fermi_dirac.f\  
fleche.f\  
gauss_band.f\  
horner.f\  
inside_d.f\  
inside_i.f\  
inside_s.f\  
intgauss.f\  
least_sq.f\  
left_right.f\  
linf.f\  
matinv.f\  
max_local.f\  
min_max.f\  
neville.f\  
newspl.f\  
newspl_gal.f\  
newton.f\  
noedif.f\  
noein.f\  
noeud.f\  
noeu_dis.f\  
pause.f\  
polyder.f\  
pulsation.f\  
schu58_n.f\  
shell.f\  
sum_n.f\  
zoning.f\  
mod_kind.$(mod_ext)\  
mod_numerique.f  
$(FC90) $(FFLAGS90) -c $*.f
```

```
mod_donnees.o mod_donnees.$(mod_ext): \  
  ini_ctes.f\  
  lit_nl.f\  
  print_ctes.f\  
  mod_kind.$(mod_ext)\  
  mod_numerique.$(mod_ext)\  
  mod_donnees.f  
  $(FC90) $(FFLAGS90) -c $*.f
```

```
mod_variables.o mod_variables.$(mod_ext): \  
    chim_gram.f\  
    inter.f\  
    sortie.f\  
    mod_kind.$(mod_ext)\  
    mod_numerique.$(mod_ext)\  
    mod_donnees.$(mod_ext)\  
    mod_variables.f  
    $(FC90) $(FFLAGS90) -c $*.f
```

```
mod_etat.o mod_etat.$(mod_ext): \  
    df_rotx.f\  
    etat.f\  
    etat_ceff.f\  
    etat_eff.f\  
    etat_gong1.f\  
    etat_gong2.f\  
    etat_irwin.f\  
    etat_mhd.f\  
    etat_opal.f\  
    etat_opalX.f\  
    etat_opalZ.f\  
mu_mol.f\  
    saha.f\  
    mod_kind.$(mod_ext)\  
    mod_numerique.$(mod_ext)\  
    mod_donnees.$(mod_ext)\  
    mod_etat.f  
    $(FC90) $(FFLAGS90) -c $*.f
```

```
mod_opa.o mod_opa.$(mod_ext): \  
    cond_mestel.f\  
    opa.f\  
    opa_compton.f\  
    opa_cond_iben.f\  
    opa_cond_mestel.f\  
    opa_gong.f\  
    opa_houdek9.f\  
    opa_int_zsx.f\  
    opa_opalC0.f\  
    opa_opal2.f\  
    opa_yveline.f\  
    opa_yveline_lisse.f\  
    z14xcotrin21.f\  
    mod_kind.$(mod_ext)\  
    mod_numerique.$(mod_ext)\  
    mod_donnees.$(mod_ext)
```

```
mod_variables.$(mod_ext)\
mod_opa.f
$(FC90) $(FFLAGS90) -c $*.f

mod_conv.o mod_conv.$(mod_ext): \
conv.f\
conv_a0.f\
conv_cgm_reza.f\
conv_cm.f\
conv_cml.f\
conv_cm_reza.f\
conv_jmj.f\
mod_kind.$(mod_ext)\
mod_donnees.$(mod_ext)\
mod_variables.$(mod_ext)\
mod_conv.f
$(FC90) $(FFLAGS90) -c $*.f

mod_atm.o mod_atm.$(mod_ext): \
atm.f\
coll_atm.f\
edding.f\
eq_atm.f\
hopf.f\
hsra.f\
k5750.f\
k5777.f\
lim_atm.f\
lim_gong1.f\
lim_tau1.f\
marcs.f\
roger.f\
taueff.f\
tdetau.f\
thermo_atm.f\
trho.f\
trho_4000.f\
mod_kind.$(mod_ext)\
mod_numerique.$(mod_ext)\
mod_donnees.$(mod_ext)\
mod_variables.$(mod_ext)\
mod_etat.$(mod_ext)\
mod_opa.$(mod_ext)\
mod_conv.$(mod_ext)\
mod_atm.f
$(FC90) $(FFLAGS90) -c $*.f
```

```

mod_nuc.o mod_nuc.$(mod_ext): \
    abon_ini.f\
    iben.f\
    nuc.f\
    planetoides.f\
    pp1.f\
    pp3.f\
    ppcno10BeBFe.f\
    ppcno10Fe.f\
    ppcno10K.f\
    ppcno10.f\
    ppcno11.f\
    ppcno12Be.f\
    ppcno12BeBFe.f\
    ppcno12Li.f\
    ppcno12.f\
    ppcno3a12Ne.f\
    ppcno3a9.f\
    ppcno3aco.f\
    ppcno3acos.f\
    ppcno9.f\
    ppcno9Fe.f\
    rq_reac.f\
    tabul_nuc.f\
    taux_nuc.f\
    vent.f\
    mod_kind.$(mod_ext)\
    mod_numerique.$(mod_ext)\
    mod_donnees.$(mod_ext)\
    mod_variables.$(mod_ext)\
    mod_nuc.f
    $(FC90) $(FFLAGS90) -c $*.f

mod_bp_for_alecian.o mod_bp_for_alecian.$(mod_ext): \
    mod_donnees.$(mod_ext)\
    mod_variables.$(mod_ext)\
    mod_bp_for_alecian.f
    $(FC90) $(FFLAGS90) -c $*.f

mod_evol.o mod_evol.$(mod_ext): \
    alecian1.f\
    coeff_rota.f\
    coeff_rota3.f\
    coeff_rota4.f\
    coeff_vth.f\
    collision.f\
    coulomb.f\

```

```
diffm.f\  
diffm_br.f\  
diffm_mp.f\  
diffft.f\  
diffft_gab.f\  
diffft_nu.f\  
diffft_smc.f\  
diffft_sun.f\  
diffus.f\  
diffw.f\  
diffw_cte.f\  
diffw_mpz.f\  
diffw_p03.f\  
diffw_toul.f\  
ecrit_rota.f\  
eq_diff_chim.f\  
eq_diff_rota3.f\  
eq_diff_rota4.f\  
eq_lisse_difg.f\  
eq_ini_rota4.f\  
evol.f\  
f_rad.f\  
initialise_rota.f\  
initialise_rota4.f\  
integrales.f\  
lmix.f\  
pertw.f\  
pertw_loc.f\  
pertw_ptm.f\  
pertw_sch.f\  
resout_chim.f\  
resout_lisse_difg.f\  
resout_rota.f\  
rkimps.f\  
tab_vth.f\  
mod_kind.$(mod_ext)\  
mod_numerique.$(mod_ext)\  
mod_donnees.$(mod_ext)\  
mod_variables.$(mod_ext)\  
mod_etat.$(mod_ext)\  
mod_nuc.$(mod_ext)\  
mod_bp_for_alecian.$(mod_ext)\  
mod_evol.f  
$(FC90) $(FFLAGS90) -c $*.f
```

```
mod_static.o mod_static.$(mod_ext): \  
coll_qs.f\  

```



```

dgrad.f\
lim_zc.f\
pertm.f\
pertm_ext.f\
pertm_msol.f\
pertm_tot.f\
pertm_waldron.f\
resout.f\
static.f\
static_m.f\
static_r.f\
thermo.f\
update.f\
mod_kind.$(mod_ext)\
mod_numerique.$(mod_ext)\
mod_donnees.$(mod_ext)\
mod_variables.$(mod_ext)\
mod_etat.$(mod_ext)\
mod_opa.$(mod_ext)\
mod_conv.$(mod_ext)\
mod_atm.$(mod_ext)\
mod_nuc.$(mod_ext)\
mod_evol.$(mod_ext)\
mod_static.f
$(FC90) $(FFLAGS90) -c $*.f

mod_cesam.o mod_cesam.$(mod_ext): \
add_ascii.f\
ascii.f\
cesam.f\
des.f\
des_m.f\
des_r.f\
dnunl.f\
list.f\
output.f\
osc_adia.f\
osc_invers.f\
osc_nadia.f\
mod_kind.$(mod_ext)\
mod_numerique.$(mod_ext)\
mod_donnees.$(mod_ext)\
mod_variables.$(mod_ext)\
mod_etat.$(mod_ext)\
mod_opa.$(mod_ext)\
mod_atm.$(mod_ext)\
mod_nuc.$(mod_ext)\

```

```
    mod_static.$(mod_ext)\
    mod_cesam.f
$(FC90) $(FFLAGS90) -c $*.f

mod_exploit.o mod_exploit.$(mod_ext): \
    add_ascii.f\
    ascii.f\
    diffw.f\
    diffw_cte.f\
    diffw_mpz.f\
    diffw_p03.f\
    diffw_toul.f\
    ini_ctes.f\
    inter_atm.f\
    lit_binaire.f\
    lit_hr.f\
    lit_nl.f\
    min_max.f\
    min_max_cond.f\
    osc_adia.f\
    osc_invers.f\
    osc_nadia.f\
    output.f\
    read_ascii.f\
    write_nl.f\
    mod_kind.$(mod_ext)\
    mod_numerique.$(mod_ext)\
    mod_donnees.$(mod_ext)\
    mod_variables.$(mod_ext)\
    mod_etat.$(mod_ext)\
    mod_opa.$(mod_ext)\
    mod_nuc.$(mod_ext)\
    mod_exploit.f
$(FC90) $(FFLAGS90) -c $*.f

clean:
$(RM) cesam2k.out libcesam2k.a $(OBJS) $(MODS)

touch:
touch $(SRCS)
```

Annexe B

Données diverses

On donne la liste des tableaux `glob` et `var` créés par `cesam`, *cf.* § 8.10 (Page 173), pour les sorties ASCII, ainsi que celle du tableau `list_cesam` servant à la compilation et à la création du module exécutable, *cf.* § 1.6 (Page 5).

B.1 Liste du tableau `glob`

```
glob(1)=mstar*msol
glob(2)=rtot*rsol
glob(3)=ltot*lsol
glob(4)=z0
glob(5)=x0
glob(6)=alpha
glob(7)=9./4.
glob(8)=1./162.
glob(9)=X dans ZC
glob(10)=Y dans ZC
glob(11)=d2p
glob(12)=d2ro
glob(13)=age
glob(14)=wrot vitesse de rotation globale
glob(15)=w_rot initial

nglob=15
```

B.2 Liste du tableau `var`

```
var(1,i): Rayon
var(2,i): Ln M/Mtot
var(3,i): Température
var(4,i): Pression
var(5,i): Densité
var(6,i): Gradient
var(7,i): Gradient
```

```

var(8,i): Luminosité
var(9,i): Opacité
var(10,i): Energie nuc+grav
var(11,i): Grand Gamma1
var(12,i): Gradient adiabatique
var(13,i): Delta
var(14,i): Cp
var(15,i): Mue^(-1)
var(16,i): Mu
var(17,i): Vaissala
var(18,i): Omega
var(19,i): dln kappa/dln T
var(20,i): dln kappa/dln ro
var(21,i): d epsilon(nuc) / d ln T
var(22,i): d epsilon(nuc) / d ln ro
var(23,i): !Ptot/Pgaz
var(24,i): !Gradient radiatif
var(25,i): d Gamma1 / d lnP (TY)
var(26,i): d Gamma1 / d lnT (PY)
var(27,i): d Gamma1 / dY (PT)
var(28,i): dP / dro (TX)
var(29,i): dP / dT (roX)
var(30,i): dP / dX (Tro)
var(31,i): du / dro (TX)
var(32,i): du / dT (roX)
var(33,i): du / dX(Tro)
var(34,i): énergie interne
var(35,i): d^2P / dro^2 (TX)
var(36,i): d^2P / dro dT (X)
var(37,i): d^2P / dT^2(roX)
var(38,i): d^2U / dro^2 (TX)
var(39,i): d^2U / dro dT (X)
var(40,i): d^2U / dT^2 (X)
var(41,i): dK / dX
var(42,i): d^2K / dT^2
var(43,i): d epsi / dX
var(44,i): dX / dR
var(45,i): J-B
var(46,i): Edding. facteur
var(ivar+j,i): xchim1g(j=1,nbelem) Abondances / gramme

```

ivar=46

B.3 Liste du tableau list_cesam

mod_kind

mod_numerique
mod_donnees
mod_variables
mod_etat
mod_opa
mod_conv
mod_atm
mod_nuc
mod_bp_for_alecian
mod_evol
mod_static
mod_cesam
mod_exploit

Annexe C

Liste des fichiers binaires

Les Fichiers binaires n'ayant pour fonction que des exploitations internes, leur contenu n'est pas détaillé. A l'exception du fichier binaire d'atmosphère `mon_modele.atm`, ces fichiers peuvent être lus par la routine `lit_binaire`, cf. §8.72 (Page 204).

- Fichier binaire du modèle de pré-séquence principale homogène : `mon_modele.B.pms`.
- Fichier binaire du modèle de séquence principale d'âge zéro homogène : `mon_modele.B.hom`.
- Fichier binaire du modèle de séquence principale d'âge zéro : `mon_modele.B.zams`, créé à la fin de ma PMS dès que l'énergie d'origine thermonucléaire devient égale à l'énergie d'origine gravifique.
- Fichier binaire du modèle de post-séquence principale, créé lorsque l'abondance d'hydrogène au centre devient inférieure à 0.001 : `mon_modele.B.post` ou `mon_modele.B.tams`.
- Fichier binaire du modèle du début de la combustion de l'hélium, créé dès que la température au centre dépasse 10^8K : `mon_modele.B.cohé`.
- Fichier binaire du modèle du début de la combustion du carbone, créé dès que la température au centre dépasse $6 \cdot 10^8\text{K}$: `mon_modele.B.coca`.
- Fichier binaire du modèle du début de la combustion de l'oxygène, créé dès que la température au centre dépasse 10^9K : `mon_modele.B.coox`.
- Fichier binaire du modèle final : `mon_modele.B.dat`.
- Fichier binaire du modèle intermédiaire : `mon_modele.B.rep`.
- Fichier binaire du modèle d'atmosphère : `mon_modele.B.atm`.

Annexe D

Liste des fichiers ASCII

Se reporter §3.4 (Page 26) pour la description du fichier ASCII de données `mon_modele.don`.

D.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, MACRE
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
      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
```

D.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
```

D.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)=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
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
```

D.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

```

D.1.4 Utilisation du nombre maximum de couches

Avec certains types de précision, *cf.* §3.4 (Page 26), 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.* § 3.5 (Page 29).

D.2 Fichier pour diagramme HR

Le fichier pour tracé/exploitation du diagramme HR est systématiquement implémenté au cours de chaque exécution de `CESAM2k`. 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 Ω),
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_{ov}/R_{\odot} , $\frac{\partial N^2}{\partial \ln R}$ (r_{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.* § 3.5 (Page 29).

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_{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 : ν_0 , $\partial\nu_{02}$,

$\partial\nu_{13}$, $A = -\int 1/rdc/dr$ et du moment d'inertie normalisé :

$$\mathcal{M} = \frac{1}{M_* R_*^2} \int_0^{M_*} R^2 dM,$$

puis les valeurs au centre P_c , T_c , ρ_c de la pression, température et densité. – 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 2.683E-01 1.484E+17 1.368E+07 7.939E+01
<-- nu0, dnu02, dnu13, A, mtI, Pc, Tc, Ro_c
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 2.695E-01 1.484E+17 1.368E+07 7.939E+01
<-- nu0, dnu02, dnu13, A, mtI, Pc, Tc, Ro_c
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  7.939E-02  1.484E+17  1.368E+07  7.939E+01
      <-- nu0, dnu02, dnu13, A, mtI, Pc, Tc, Ro_c
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`.

D.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, $\log T_{eff}$, $\log L/L_{\odot}$, $\log R/R_{\odot}$, 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`.

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

Le fichier ASCII des variables de la diffusion du moment cinétique est créé dans l'environnement du calcul suivant les circonstances décrites au §3.11 (Page 39). Il peut être exploité par le programme de dessin `des2k_rot` du sous-directory `EXPLOIT`, cf. §1.4 (Page 3).

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 ^4He , numéro du modèle, indice de ^7Li .

– Ligne suivantes : FORMAT(5es19.12) variables et composition chimique.

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

```
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-06-5.76838655587E-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
```

D.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. § 3.5 (Page 29). 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
.....
```

D.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. § 3.5 (Page 29). 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-14,0.130735,2,1,0,0,0,0,0.143355,0.74339
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-14,0.129992,2,1,0,0,0,0,0.13
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.,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.
.....
.....

```

D.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.* § 3.5 (Page 29). La première ligne permet l'identification des quantités extraites. Exemple :

```

model_num,age_Ma,R_sun,Teff_K,lum_sun,A,dnu02,dnu13,Nu0,X_c,Y,Z/X
0,0.,0.878318,5726.22,0.744397,76.3024,22.8907,38.1512,167.668,0.715686,0.270000,0.200000E-01
1,5.00000000000,0.879827,5732.36,0.750164,76.4207,22.9262,38.2104,167.250,0.715411,0.270000,0.200000E-01
2,11.0000000000,0.880609,5728.61,0.749535,76.1302,22.8391,38.0651,167.024,0.715024,0.270000,0.200000E-01
3,18.2000000000,0.882366,5727.15,0.751761,75.9132,22.7739,37.9566,166.533,0.714548,0.270000,0.200000E-01
.....
.....

```

Annexe E

Liste détaillée des modules

Afin de faciliter la lecture des algorithmes de CESAM2*k*, on donne la liste détaillée des variables et routines des modules, en indiquant l'endroit de leur initialisation ; celles-ci sont souvent effectuées dans des routines dépendantes, *cf.* § 8.1 (Page 167), par raison de concision, seuls les noms des routines génériques sont mentionnés. Les routines étant par ailleurs largement décrites, elles ne sont ci-après que citées. Chaque routine PRIVATE et/ou PUBLIC est introduite sous la forme d'INCLUDE dans son module d'appartenance. Ci-après, la description des fonctions des variables logiques correspond à leur valeur .TRUE. L'ordre de présentation est celui de la compilation indiqué dans le fichier `list_cesam`, *cf.* § B.3 (Page 263), du sous-directory SOURCE.

E.1 Module `mod_kind`

Le module `mod_kind` regroupe les types des variables.

- Paramètres PUBLIC :
 - `dp` : Définit les variables en double précision.
 - `sp` : Définit les variables en simple précision.

E.2 Module `mod_numerique`

Le module `mod_numerique` regroupe les routines purement numériques.

- Variable PUBLIC logical :
 - `no_croiss` : La suite des abscisses n'est pas strictement croissante, ou encore une autre difficulté est survenue lors de l'exécution.
- Routines PRIVATE : `bval0`, `colpnt`, `difdiv`, `horner`, `schu58_n`.
- Routines PUBLIC : `arb_rom`, `boite`, `box`, `bsp1ddn`, `bsp1dn`, `bsp_dis`, `bvald`, `bval1`, `coll`, `delete_doubles`, `fermi_dirac`, `gauss_band`, `genere_bases`, `intgauss`, `linf`, `matinv`, `max_local`, `min_max`, `neville`, `newspl`, `newton`, `noedif`, `noein`, `noeud`, `noeu_dis`, `pause`, `polyder`, `shell`, `sum_n`, `zoning`

E.3 Module `mod_donnees`

Le module `mod_donnees` regroupe l'ensemble des quantités fixes au cours de l'évolution.

– Paramètres PUBLIC double précision :

`d_conv` : Coefficient de diffusion dans les zones mélangées.

`dtmin` : Pas temporel minimum en Myrs.

`dx_tams` : Précision sur l'abondance en X à la fin de la séquence principale d'âge zéro.

`phi` : $d \ln \rho / d \ln \mu$.

`x_tams` : Abondance en X définissant la fin de la séquence principale d'âge zéro.

– Paramètres PUBLIC integer :

`n_min` : Nombre minimum de couches.

`pnzc` : Nombre maximum de zones convectives pouvant être prises en compte.

`r_qs` : Ordre des équations différentielles des équations d'équilibre quasi-statique.

– Paramètres PUBLIC character :

`version` : Numéro de version lu par un `INCLUDE` du fichier `journal`, cf. § 1.2 (Page 2).

– Variables PUBLIC double précision :

`ab_ini` : Abondances initiales, initialisé dans `nuc.f`.

`ab_min` : Abondances minimales, initialisé dans `nuc.f`.

`nucleo` : Masses atomiques des isotopes, initialisé dans `tabul_nuc.f`.

`rot_min` : Valeurs négligeables des variables de la diffusion du moment cinétique, initialisé dans `lit_n1.f`.

`xvent` : Composition chimique du vent, initialisé dans `vent.f`.

`zi` : Charges des isotopes, initialisé dans `tabul_nuc.f`.

`abon_m` : Abondances initiales par masse des éléments chimiques, initialisé dans `abon_ini.f`.

`aal27` : Masse atomique du ^{27}Al , initialisé dans `taux_nuc.f`.

`abe7` : Masse atomique du ^7Be , initialisé dans `taux_nuc.f`.

`abe9` : Masse atomique du ^9Be , initialisé dans `taux_nuc.f`.

`ab11` : Masse atomique du ^{11}B , initialisé dans `taux_nuc.f`.

`ac12` : Masse atomique du ^{12}C , initialisé dans `taux_nuc.f`.

`ac13` : Masse atomique du ^{13}C , initialisé dans `taux_nuc.f`.

`afe56` : Masse atomique du ^{56}Fe , initialisé dans `taux_nuc.f`.

af18 : Masse atomique du ^{18}F , initialisé dans `taux_nuc.f.`
af19 : Masse atomique du ^{19}F , initialisé dans `taux_nuc.f.`
agemax : Age maximum à atteindre, lu dans `lit_nl.f.`
ah : Masse atomique du ^1H , initialisé dans `taux_nuc.f.`
ah2 : Masse atomique du ^2H , initialisé dans `taux_nuc.f.`
ahe3 : Masse atomique du ^3He , initialisé dans `taux_nuc.f.`
ahe4 : Masse atomique du ^4He , initialisé dans `taux_nuc.f.`
ali6 : Masse atomique du ^6Li , initialisé dans `taux_nuc.f.`
ali7 : Masse atomique du ^7Li , initialisé dans `taux_nuc.f.`
alpha : Longueur de mélange, lu dans `lit_nl.f.`
amg23 : Masse atomique du ^{23}Mg , initialisé dans `taux_nuc.f.`
amg24 : Masse atomique du ^{24}Mg , initialisé dans `taux_nuc.f.`
amg25 : Masse atomique du ^{25}Mg , initialisé dans `taux_nuc.f.`
amg26 : Masse atomique du ^{26}Mg , initialisé dans `taux_nuc.f.`
amu : Masse atomique unité m_u , initialisé dans `taux_nuc.f.`
an : Masse atomique du neutron, initialisé dans `taux_nuc.f.`
ana23 : Masse atomique du ^{23}Na , initialisé dans `taux_nuc.f.`
ane20 : Masse atomique du ^{20}Ne , initialisé dans `taux_nuc.f.`
ane21 : Masse atomique du ^{21}Ne , initialisé dans `taux_nuc.f.`
ane22 : Masse atomique du ^{22}Ne , initialisé dans `taux_nuc.f.`
an13 : Masse atomique du ^{13}N , initialisé dans `taux_nuc.f.`
an14 : Masse atomique du ^{14}N , initialisé dans `taux_nuc.f.`
an15 : Masse atomique du ^{15}N , initialisé dans `taux_nuc.f.`
ap : Masse atomique du proton, initialisé dans `taux_nuc.f.`
ap31 : Masse atomique du ^{31}P , initialisé dans `taux_nuc.f.`
ao16 : Masse atomique du ^{16}O , initialisé dans `taux_nuc.f.`
ao17 : Masse atomique du ^{17}O , initialisé dans `taux_nuc.f.`
ao18 : Masse atomique du ^{18}O , initialisé dans `taux_nuc.f.`
asi28 : Masse atomique du ^{28}Si , initialisé dans `taux_nuc.f.`
as32 : Masse atomique du ^{32}S , initialisé dans `taux_nuc.f.`
aradia : Constante de la radiation a , initialisé dans `ini_ctes.f.`
clight : Célérité de la lumière c , initialisé dans `ini_ctes.f.`
cpturb : Paramètre de pression turbulente, lu dans `lit_nl.f.`
ctel : Facteur de répartition en luminosité, initialisé dans `cesam.f.`
ctem : Facteur de répartition en masse, initialisé dans `cesam.f.`
ctep : Facteur de répartition en pression, initialisé dans `cesam.f.`
cter : Facteur de répartition en rayon, initialisé dans `cesam.f.`

- ctet** : Facteur de répartition en température, initialisé dans `cesam.f`.
- dpsi** : Taux de variation maximale pour la constante de répartition entraînant une modification du nombre de couches `n_qs` dans le cas standard d'une grille ajustable, initialisé dans `cesam.f`.
- dn_fixe** : Taux de variation maximale pour la constante de répartition entraînant une modification du nombre de couches `nc_fixe` dans le cas d'une grille fixe, initialisé dans `cesam.f`.
- dtlist** : Intervalle de temps entre deux listings complets, lu dans `lit_nl.f`.
- dtmax** : Pas temporel maximal, initialisé dans `cesam.f`.
- dt0** : Pas temporel initial, initialisé dans `cesam.f`.
- d_grav** : Variation maximale de l'énergie graviphique sur un pas temporel, initialisé dans `cesam.f`.
- d_turb** : Coefficient de diffusion turbulente, lu dans `lit_nl.f`.
- echarg** : Charge de l'électron, initialisé dans `ini_ctes.f`.
- eve** : Energie de l'électronvolt, initialisé dans `ini_ctes.f`.
- fesh_sol** : Valeur du [Fe/H] solaire, initialisé dans `ini_ctes.f`.
- g** : Constante de la gravitation G , initialisé dans `ini_ctes.f`.
- gmsol** : Produit $G \times M_{\odot}$, initialisé dans `ini_ctes.f`.
- granr** : Constante des gaz parfaits \mathcal{R} , initialisé dans `ini_ctes.f`.
- he_core** : Masse du noyau d'hélium à partir de laquelle l'évolution sera arrêtée, lu dans `lit_nl.f`.
- hhe_core** : Masse réduite du noyau d'hélium à partir de laquelle l'évolution sera arrêtée, initialisé dans `lit_nl.f`.
- hpl** : Constante de Plank h , initialisé dans `ini_ctes.f`.
- kbol** : Constante de Boltzman k , initialisé dans `ini_ctes.f`.
- lbol0** : Zéro des magnitudes bolométriques, initialisé dans `ini_ctes.f`.
- li_ini** : Abondance de ${}^7\text{Li}$ dans le noyau d'un modèle de séquence principale d'âge zéro homogène, initialisé dans `abon_ini.f`.
- lnt_stop** : L'évolution est arrêtée lorsque cette température effective, en \ln , est franchie par valeurs croissantes ou décroissantes, initialisé dans `lit_nl.f`.
- ln_Tli** : Température au dessus de laquelle il n'y a pas de lithium initial pour le modèle de séquence principale d'âge zéro, initialisé dans `abon_ini.f`.
- ln10** : $\ln(10)$, initialisé dans `ini_ctes.f`.
- loc_zc** : Paramètre de localisation des limites zones radiatives / zones convectives, initialisé dans `cesam.f`.
- log_teff** : L'évolution est arrêtée lorsque cette température effective est franchie par valeurs croissantes ou décroissantes, lu dans `lit_nl.f`.
- lsol** : Luminosité solaire L_{\odot} , initialisé dans `ini_ctes.f`.
- mdot** : Taux de perte de masse, lu dans `lit_nl.f`.

me : Masse de l'électron m_e , initialisé dans `ini_ctes.f`.
msol : Masse solaire M_\odot , initialisé dans `ini_ctes.f`.
mterre : Masse de la terre, initialisé dans `ini_ctes.f`.
mtot : Masse initiale du modèle, lu dans `lit_nl.f`.
ovshti : Taux d'overshoot inférieur, lu dans `lit_nl.f`.
ovshts : Taux d'overshoot supérieur, lu dans `lit_nl.f`.
pi : π , initialisé dans `ini_ctes.f`.
precit : Paramètre de précision de l'intégration temporelle de la composition chimique, initialisé dans `cesam.f`.
precix : Paramètre de précision de l'intégration spatiale de l'équilibre quasi-statique, initialisé dans `cesam.f`.
p_pertw : Paramètre de perte de moment cinétique, lu dans `lit_nl.f`.
re_nu : Taux de turbulence radiative, lu dans `lit_nl.f`.
ro_test : Valeur minimale de la densité en dessous de laquelle la variation d'énergie graviphique n'est plus considérée comme significative, initialisé dans `cesam.f`.
rsol : Rayon solaire R_\odot , initialisé dans `ini_ctes.f`.
secon6 : Nombre de secondes par million d'années, initialisé dans `ini_ctes.f`.
sigma : Constante de Stéfan σ , initialisé dans `ini_ctes.f`.
tau_max : Profondeur optique de raccord atmosphère/enveloppe, lu dans `lit_nl.f`.
t_inf : Température minimale de tabulation des réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
t_sup : Température maximale de tabulation des réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
t_stop : Température centrale arrêtant le calcul de l'évolution, lu dans `lit_nl.f`.
ua : Unité astronomique, initialisé dans `ini_ctes.f`.
w_form : Facteur entre 0 et 1 permettant de modifier le profil de la vitesse angulaire initiale, initialisé dans `initialise_u.f`.
w_rot : Vitesse angulaire initiale, lu dans `lit_nl.f`.
x0 : Abondance initiale par masse d'hydrogène X , lu ou déterminé dans `lit_nl.f`.
x_stop : Abondance centrale d'hydrogène arrêtant le calcul de l'évolution, lu dans `lit_nl.f`.
y0 : Abondance initiale par masse d'hélium Y , lu dans `lit_nl.f`.
zsx_sol : Z/X solaire, initialisé dans `ini_ctes.f`.
zsx0 : Rapport initial Z/X , lu ou initialisé dans `lit_nl.f`.
z0 : Abondance initiale par masse des métaux Z , initialisé dans `lit_nl.f`.

– Variables PUBLIC simple précision :

- dh** : Espace en hauteur entre cadres des dessins, initialisé dans `des.f`.
dl : Espace en largeur entre cadres des dessins, initialisé dans `des.f`.
h : Hauteur des cadres des dessins, initialisé dans `des.f`.
ld : Largeur des cadres des dessins, initialisé dans `des.f`.
fesh_des : Cible en [Fe/H] en surface, initialisé dans `des.f`.
L_des : Cible en L/L_{\odot} , initialisé dans `des.f`.
teff_des : Cible en température, initialisé dans `des.f`.
logteff_max : Limite à gauche des abscisses du cadre du diagramme HR, initialisé dans `des.f`.
logteff_min : Limite à droite des abscisses du cadre du diagramme HR, initialisé dans `des.f`.
logl_max : Limite supérieure des ordonnées du cadre du diagramme HR, initialisé dans `des.f`.
logl_min : Limite inférieure des ordonnées du cadre du diagramme HR, initialisé dans `des.f`.
xleft : Limite à droite des dessins, initialisé dans `des.f`.
ybot : Limite inférieure des dessins, initialisé dans `des.f`.
y_age : Ordonnées pour l'écriture de l'âge au dessus des cadres des dessins, initialisé dans `des.f`.

– Variables PUBLIC integer :

- Kdes_rot** : Flag d'identification du type de dessin pour la rotation, initialisé dans `lit.nl`.
Krot : Flag d'identification du type de rotation, initialisé dans `lit.nl`.
ife56 : Indice de l'isotope ^{56}Fe , initialisé dans `nuc.f`.
ihe4 : Indice de l'isotope ^4He , initialisé dans `nuc.f`.
iLi7 : Indice de l'isotope ^7Li , initialisé dans `nuc.f`.
ini0 : Nombre maximum d'itérations globales avec réestimation de la composition chimique et de la délimitation des limites zones radiatives / zones convectives, initialisé dans `cesam.f`.
Ipg : Indice de la variable pression gazeuse, initialisé dans `cesam.f`.
i_ex : Indice de l'élément fictif, initialisé dans `nuc.f`.
m_ch : Ordre des B-splines d'interpolation de la composition chimique, initialisé dans `cesam.f`.
m_ptm : Ordre des B-splines d'interpolation de la perte de masse, initialisé dans `cesam.f`.
m_qs : Ordre des B-splines pour la résolution de l'équilibre quasi-statique, initialisé dans `cesam.f`.

- m_rot** : Ordre des B-splines d'interpolation du moment cinétique, initialisé dans `cesam.f`.
- m_tds** : Ordre des B-splines d'interpolation de l'énergie gravipitique, initialisé dans `cesam.f`.
- nb_max_modeles** : Nombre maximum de modèles à calculer, lu dans `lit_nl.f`.
- nchim** : Nombre d'isotopes, initialisé dans `nuc.f`.
- ne** : Nombre de variables pour l'équilibre quasi-statique, initialisé dans `cesam.f`.
- nrl** : Nombre de coefficients pour la rotation, initialisé dans `lit_nl.f`.
- nrot** : Nombre de variables pour la diffusion du moment cinétique, initialisé dans `lit_nl.f`.
- n_atm** : Nombre de couches dans l'atmosphère, initialisé dans `cesam.f`.
- n_max** : Nombre maximum de couches pour l'équilibre quasi-statique, initialisé dans `cesam.f`.
- ordre** : Ordre d'intégration des équations d'évolution de la composition chimique sans diffusion, initialisé dans `cesam.f`.
- ord_qs** : Ordre des B-splines pour l'intégration des équations d'équilibre quasi-statique, initialisé dans `cesam.f`.

– Variables PUBLIC logical :

- ajuste** : On ajustera à 1% près les valeurs de sortie, par exemple l'abondance d'hydrogène au centre, initialisé dans `cesam.f` suivant le type de précision requis.
- all_output** : On garde dans l'environnement tous les fichiers binaires de reprise dont le nom comporte le numéro du modèle; initialisé dans `lit_nl.f`.
- all_rep** : On garde dans l'environnement tous les fichiers ASCII dont le nom comporte le numéro du modèle; initialisé dans `lit_nl.f`.
- diffusion** : On tient compte de la diffusion microscopique des isotopes, lu dans `lit_nl.f`.
- en_masse** : Utilisation des variables lagrangiennes, lu dans `lit_nl.f`.
- garde_xish** : On conserve les rapports métal/hydrogène de la mixture initiale, lu dans `lit_nl.f`.
- grad_ovi** : On utilise le gradient adiabatique dans les extensions inférieures des zone convective; initialisé dans `lit_nl.f`.
- grad_ovs** : On utilise le gradient adiabatique dans les extensions supérieures des zone convective; initialisé dans `lit_nl.f`.
- grille_fixe** : La grille d'interpolation de la composition chimique est fixe, lu dans `lit_nl.f`.
- He_ajuste** : On ajustera la limite du noyau d'hélium avant de sortir, initialisé dans `lit_nl.f`.

jpz : Utilisation du formalisme de pénétration convective de JP. Zahn, lu dans `lit_nl.f`.

kipp : Utilisation de l'approximation de Kippenhahan, lu dans `lit_nl.f`.

ledoux : Utilisation du critère de convection de Ledoux, lu dans `lit_nl.f`.

lim_ro : Utilisation de la limite en densité pour la restitution de l'atmosphère, lu dans `lit_nl.f`.

lim_jpz : Utilisation de la limite de JpZh pour la diffusion du moment angulaire, lu dans `lit_nl.f`.

lisse : Il y aura lissage par `contour` des profils des abondances, initialisé dans `cesam.f` suivant le type de précision requis.

mitler : Utilisation du formalisme de Mitler pour le calcul de l'effet d'écran, lu dans `lit_nl.f`.

modif_chim : On personnalisera la composition chimique initiale, lu dans `lit_nl.f`.

mu_saha : Pour le calcul des coefficients de la diffusion du moment cinétique, le poids moléculaire moyen sera calculé en tenant compte des divers taux d'ionisation calculés par `saha.f`, initialisé dans `cesam.f` suivant le type de précision requis.

mvt_dis : On tiendra compte des déplacements des discontinuités pour la détermination de la composition chimique, initialisé dans `cesam.f`.

pturb : On tiendra compte de la pression turbulente, initialisé dans `lit_nl`.

rep_atm : On utilisera un fichier binaire d'atmosphère, s'il existe, initialisé dans `cesam.f`.

rot_solid : On supposera la rotation solide, lu dans `lit_nl.f`.

t_ajuste : On ajustera la température centrale avant de sortir, initialisé dans `lit_nl.f`.

x_ajuste : On ajustera l'abondance centrale d'hydrogène avant de sortir, initialisé dans `lit_nl.f`.

– Variables PUBLIC character :

precision : Désignation du type de calcul, lu dans `lit_nl.f`.

nom_elem : Noms des isotopes utilisés, initialisé dans `tabul_nuc.f`.

nom_rot : Noms des variables utilisées pour la diffusion du moment cinétique, initialisé dans `lit_nl.f`.

arret : Désignation du type d'arrêt, lu dans `lit_nl.f`.

nom_xheavy : Nom de l'élément fictif, initialisé dans `nuc.f`.

unit : Nom de l'unité utilisée pour la donnée de la vitesse angulaire initiale, initialisé dans `lit_nl.f`.

langue : Nom de la langue utilisée pour les commentaires, initialisé dans `cesam.f`.

nom_atm : Nom du type d'atmosphère, lu dans `lit_nl.f`.

nom_abon : Nom des abondances initiales, lu dans `lit_nl.f`.

nom_conv : Nom du type de convection, lu dans `lit_nl.f`.

nom_ctes : Nom du groupe des constantes physiques, lu dans `lit_nl.f`.

nom_des : Nom du type de dessin, lu dans `lit_nl.f`.

nom_diffm : Nom du type de coefficients de diffusion microscopique, lu dans `lit_nl.f`.

nom_diffw : Nom du type de coefficients de diffusion de moment cinétique, lu dans `lit_nl.f`.

nom_difft : Nom du type de coefficients de diffusion turbulente à utiliser, lu dans `lit_nl.f`.

nom_etat : Nom du type d'équation d'état, lu dans `lit_nl.f`.

nom_frad : Nom du type d'accélération radiatives, lu dans `lit_nl.f`.

nom_nuc : Nom du type de réseau de réactions thermonucléaires, lu dans `lit_nl.f`.

nom_nuc_cpl : Nom du type de compilation de réactions thermonucléaires, lu dans `lit_nl.f`.

nom_output : Nom du type de sortie ASCII, lu dans `lit_nl.f`.

nom_pertm : Nom du type de perte de masse, lu dans `lit_nl.f`.

nom_pertw : Nom du type de perte de moment cinétique, lu dans `lit_nl.f`.

nom_tdetau : Nom du type de loi $T(\tau)$, lu dans `lit_nl.f`.

nom_thw : Nom de la théorie de la diffusion du moment du moment cinétique, lu dans `lit_nl.f`.

nom_fich2 : Nom du fichier de sortie `mon_modele.lis`, lu dans `lit_nl.f`.

thw : Noms des théories de la diffusion du moment du moment cinétique implantées dans `CESAM2k`, initialisé dans `lit_nl.f`.

f_eos : Nom du fichier d'équation d'état, lu dans `lit_nl.f`.

f_opa : Nom du fichier d'opacité, lu dans `lit_nl.f`.

nom_opa : Nom du type d'opacité, lu dans `lit_nl.f`.

source : Nom de la source des constantes physiques, initialisé dans `ini_ctes.f`.

methode : Description de la méthode de calcul, initialisé dans `cesam.f`.

device : Nom du device de dessin, initialisé dans `des.f`.

nom_chemin : Nom du chemin du directory des données physiques, lu dans `lit_nl.f`.

– **Routines PUBLIC :**

`lit_nl`, `ini_ctes`, `print_ctes`.

– **Routines PRIVATE :**

`ctes_85`, `ctes_94`, `ctes_94m`.

E.4 Module `mod_variables`

Ce module regroupe les routines d'usage général et les quantités variant au cours d'une évolution. Par exemple, le vecteur des variables quasi-statiques, ou encore le vecteur de composition chimique.

– Variables **PUBLIC** double précision :

bp : Table des coefficients des B-splines des variables quasi-statiques, calculée dans `coll_qs.f`.

bp_t : Table des coefficients des B-splines des variables quasi-statiques au pas temporel précédent, réinitialisé dans `update.f`.

chim : Table des coefficients des B-splines de la composition chimique, calculé dans `evol.f` ou `resout_diff_chim.f`.

chim_t : Table des coefficients des B-splines de la composition chimique au pas temporel précédent, réinitialisé dans `update.f`.

old_ptm : Tables des coefficients des B-splines des masses, calculé dans `pertm.f`.

rota : Table des coefficients des B-splines de la rotation, calculé dans `resout_rota.f`.

rota_t : Table des coefficients des B-splines de la rotation au pas temporel précédent, réinitialisé dans `update.f`.

tds : Table des coefficients des B-splines de l'énergie graviphique, calculé dans `cesam.f`.

tds_t : Table des coefficients des B-splines de l'énergie graviphique au pas temporel précédent, réinitialisé dans `update.f`.

mc : Table des masses pour l'interpolation de la composition chimique, calculé dans `evol.f` ou `diffus.f`.

mct : Vecteur nodal des masses pour l'interpolation de la composition chimique, calculé dans `evol.f` ou `diffus.f`.

mc_fixe : Table des masses de la grille fixe, calculé dans `evol.f`.

mc_t : Table des masses pour l'interpolation de la composition chimique au pas temporel précédent, réinitialisé dans `update.f`.

mct_t : Vecteur nodal des masses pour l'interpolation de la composition chimique au pas temporel précédent, réinitialisé dans `update.f`.

mrot : Table des masses pour l'interpolation du moment cinétique, calculé dans `diffus.f`.

mrott : Vecteur nodal des masses pour l'interpolation du moment cinétique, calculé dans `diffus.f`.

mrott_t : Vecteur nodal des masses pour l'interpolation du moment cinétique au pas temporel précédent, réinitialisé dans `update.f`.

mrot_t : Table des masses pour l'interpolation du moment cinétique au pas temporel précédent, réinitialisé dans `update.f`.

- m23** : Table de masses simplifiant l'interpolation inverse, calculé dans `resout.f`.
- m23_t** : Table de masses simplifiant l'interpolation inverse au pas temporel précédent, réinitialisé dans `update.f`.
- q** : Table des abscisses entières pour l'interpolation des variables quasi-statiques, calculé dans `lim_zc.f`.
- qt** : Vecteur nodal des abscisses entières, calculé dans `lim_zc.f`.
- q_t** : Table des abscisses entières au pas temporel précédent, réinitialisé dans `update.f`.
- qt_t** : Vecteur nodal des abscisses entières au pas temporel précédent, réinitialisé dans `update.f`.
- r2** : Table de rayons simplifiant l'interpolation inverse, calculé dans `.f`.
- r2_t** : Table de rayons simplifiant l'interpolation inverse au pas temporel précédent, réinitialisé dans `update.f`.
- xl** : Abscisses des limites pour l'équilibre quasi-statique, calculé dans `coll_qs.f`.
- xt_ptm** : Vecteur nodal pour la tabulation de la perte de masse, calculé dans `pertm.f`.
- xt_tds** : Vecteur nodal pour la tabulation de l'énergie graviphique, calculé dans `cesam.f`.
- xt_tds_t** : Vecteur nodal pour la tabulation de l'énergie graviphique au pas temporel précédent, réinitialisé dans `update.f`.
- x_ptm** : Table des masses pour la tabulation de la perte de masse, calculé dans `.f`.
- x_tds** : Table des masses pour la tabulation de l'énergie graviphique, calculé dans `.f`.
- x_tds_t** : Table des masses pour la tabulation de l'énergie graviphique au pas temporel précédent, réinitialisé dans `update.f`.
- x_planet** : Composition chimique des planétoïdes, initialisé dans `planetoides.f`.
- r_zc_conv** : Rayons des limites des zones convectives, calculé dans `lim_zc.f`.
- m_zc** : Masses aux limites des zones convectives, calculé dans `lim_zc.f`.
- m_zc_t** : Masses aux limites des zones convectives au pas temporel précédent, réinitialisé dans `update.f`.
- r_zc** : Rayons des zones convectives, calculé dans `lim_zc.f`.
- r_zc_t** : Rayons des zones convectives au pas temporel précédent, réinitialisé dans `update.f`.
- r_ov** : Rayons des zones convectives avec overshoot, calculé dans `lim_zc.f`.
- r_ov_t** : Rayons des zones convectives avec overshoot au pas temporel précédent, réinitialisé dans `update.f`.

age : Age du modèle, calculé dans `cesam.f`.
c_iben : Constante de Iben , calculé dans `cesam.f`.
mstar : Masse du modèle, calculé dans `atm.f`.
mstar_t : Masse du modèle au pas temporel précédent, réinitialisé dans `update.f`.
mw_tot : Moment cinétique total, calculé dans `resout.f`.
mw_tot_t : Moment cinétique total au pas temporel précédent, réinitialisé dans `update.f`.
psi0 : Constante de répartition, calculé dans `resout.f`
rstar : Rayon de l'étoile, calculé dans `atm.f`.
wrot : Vitesse angulaire, calculé dans `resout.f`.
wrot_t : Vitesse angulaire au pas temporel précédent, réinitialisé dans `update.f`.

– Variables PUBLIC integer :

jlim : Indices des limites zones radiatives / zones convectives, calculé dans `lim_zc.f`.
jlim_t : Indices des limites zones radiatives / zones convectives au pas temporel précédent, réinitialisé dans `update.f`.
dim_ch : Dimension de l'espace des splines pour la composition chimique, réinitialisé dans `evol.f`.
dim_qs : Dimension de l'espace des splines pour les variables de structure, réinitialisé dans `lim_zc.f`.
dim_rot : Dimension de l'espace des splines pour la diffusion du moment cinétique, réinitialisé dans `evol.f`.
id_conv : Indices des débuts des zones convectives, réinitialisé dans `evol.f`.
if_conv : Indices des fins des zones convectives, réinitialisé dans `evol.f`.
knot : Nombre de points du vecteur nodal pour l'interpolation des variables quasi-statiques, calculé dans `coll.f`.
knotc : Nombre de points du vecteur nodal pour l'interpolation de la composition chimique, calculé dans `evol.f` ou `resout_diff_chim`.
knotc_t : Nombre de points du vecteur nodal pour l'interpolation de la composition chimique au pas temporel précédent, réinitialisé dans `update.f`.
knot_ptm : Nombre de points du vecteur nodal pour l'interpolation de la perte de masse, calculé dans `pertm.f`.
knotr : Nombre de points du vecteur nodal pour l'interpolation du vecteur de mélange, calculé dans `diffus.f`.
knotr_t : Nombre de points du vecteur nodal pour l'interpolation du vecteur de mélange, au pas temporel précédent, réinitialisé dans `update.f`.
knot_t : Nombre de points du vecteur nodal pour l'interpolation des variables quasi-statiques au pas temporel précédent, réinitialisé dans `update.f`.

knot_tds : Nombre de points du vecteur nodal pour l'interpolation de l'énergie graviphique, calculé dans `cesam.f`.

knot_tds_t : Nombre de points du vecteur nodal pour l'interpolation de l'énergie graviphique au pas temporel précédent, réinitialisé dans `update.f`.

lim : Indices des limites zones radiatives / zones convectives, calculé dans `lim_zc.f`.

lim_t : Indices des limites zones radiatives / zones convectives au pas temporel précédent, réinitialisé dans `update.f`.

model_num : Indice du dernier modèle calculé depuis le début de l'évolution compte tenu des reprises, réinitialisé dans `cesam`.

nb_modeles : Nombre de modèles calculés au cours du présent run, réinitialisé dans `cesam`.

nc_fixe : Nombre de points de la grille fixe d'interpolation de la composition chimique, calculé dans `evol.f`.

n_ch : Nombre de points de la grille d'interpolation de la composition chimique, calculé dans `evol.f`.

n_ch_t : Nombre de points de la grille d'interpolation de la composition chimique au pas temporel précédent, réinitialisé dans `update.f`.

n_ptm : Nombre de points de la grille d'interpolation de la perte de masse, calculé dans `pertm.f`.

n_qs : Nombre de points de la grille d'interpolation de l'équilibre quasi-statique, calculé dans `lim_zc.f`.

n_qs_t : Nombre de points de la grille d'interpolation de l'équilibre quasi-statique au pas temporel précédent, réinitialisé dans `update.f`.

n_rot : Nombre de points de la grille d'interpolation de la rotation, calculé dans `diffus.f`.

n_rot_t : Nombre de points de la grille d'interpolation de la rotation au pas temporel précédent, réinitialisé dans `update.f`.

n_tds : Nombre de points de la grille d'interpolation de l'énergie graviphique, calculé dans `.f`.

n_tds_t : Nombre de points de la grille d'interpolation de l'énergie graviphique au pas temporel précédent, réinitialisé dans `update.f`.

– Variables PUBLIC logical :

lconv : Début d'une zone convective, calculé dans `lim_zc.f`.

lconv_t : Début d'une zone convective au pas temporel précédent, réinitialisé dans `update.f`.

lhe_stop : Arrêt lorsque le cœur d'hélium a atteint la masse requise, calculé dans `cesam.f`.

lt_stop : Arrêt lorsque la température centrale a atteint la valeur requise, calculé dans `.f`.

lx_stop : Arrêt lorsque l'abondance centrale d'hydrogène a atteint la valeur requise, calculé dans `cesam.f`.

tot_conv : Le modèle est totalement convectif, calculé dans `lim_zc.f`.

tot_rad : Le modèle est totalement radiatif, calculé dans `.f`.

– Routines PUBLIC :

`chim_gram`, `inter`, `sortie`.

E.5 Module `mod_etat`

Le module `mod_etat` regroupe les routines concernées par l'équation d'état.

– Routines PRIVATE :

`etat_ceff`, `etat_eff`, `etat_gong1`, `etat_gong2`, `etat_mhd`, `etat_opal`,
`etat_opalX`, `etat_opalZ`.

– Routines PUBLIC :

`df_rotx`, `etat`, `saha`.

E.6 Module `mod_opa`

Le module `mod_opa` regroupe les routines concernant le calcul de l'opacité. La routine d'opacité `z14xcotrin21.f` n'est qu'associée au module en raison de sa programmation ancienne.

– Routines PRIVATE :

`kappa_cond`, `opa_compton`, `opa_gong`, `opa_houdek9`, `opa_int_zsx`,
`opa_opalCO`, `opa_opal2`, `opa_yveline`, `opa_yveline_lisse`.

– Routine PUBLIC :

`opa`.

E.7 Module `mod_conv`

Le module `mod_conv` regroupe les routines concernant la convection.

– Routines PRIVATE :

`conv_a0`, `conv_cgm_reza`, `conv_cm`, `conv_cml`, `conv_cm_reza`,
`conv_jmj`.

– Routine PUBLIC :

`conv`.

E.8 Module mod_atm

Le module mod_atm regroupe les routines de restitution de l'atmosphère.

– Variables PUBLIC double précision :

bp_atm : Tableau des coefficients des B-splines d'interpolation de l'atmosphère, calculé dans coll_atm.f.

bp_atm_t : Tableau des coefficients des B-splines d'interpolation de l'atmosphère au pas temporel précédent, réinitialisé dans update.f.

dlpp_atm : Tableau des quotients $d \ln P_{\text{gaz}} / d \ln P_{\text{tot}}$, calculé dans atm.f.

m_atm : Tableau des masses, abscisses lagrangiennes, calculé dans coll_atm.f.

p_atm : Tableau des pressions gazeuses, calculé dans atm.f.

pt_atm : Tableau des pressions totales, calculé dans atm.f.

r_atm : Tableau des rayons, calculé dans atm.f.

tau : Tableau des profondeurs optiques, calculé dans atm.f.

t_atm : Tableau des températures, calculé dans atm.f.

x_atm : Tableau des abscisses pour la restitution de l'atmosphère, calculé dans atm.f.

x_atm_t : Tableau des abscisses pour la restitution de l'atmosphère au pas temporel précédent, réinitialisé dans update.f.

xt_atm : Vecteur nodal des abscisses pour la restitution de l'atmosphère, calculé dans lim_atm.f.

tau_min : Epaisseur optique de la couche externe de l'atmosphère, initialisé dans tdetau.f.

– Variables PRIVATE double précision :

delfim : Pente de la fonction de répartition à gauche de la couche d'indice n23_atm, calculé dans lim_atm.f.

delfip : Pente de la fonction de répartition à droite de la couche d'indice n23_atm, calculé dans lim_atm.f.

ltaue : $\ln(\tau_{\text{min}})$, initialisé dans lim_atm.f.

ltauf : $\ln(\tau_{\text{max}})$, initialisé dans lim_atm.f.

tau_min : Epaisseur optique de la couche externe de l'atmosphère, initialisé dans tdetau.f.

– Variables PRIVATE integer :

Ipgt : Indice de la variable pression gazeuse, initialisé dans lim_atm.f.

ne_atm : Nombre d'équations pour la restitution de l'atmosphère, initialisé dans lim_atm.f.

n23_atm : Indice de la couche définissant le rayon bolométrique où $T(\tau^*) = T_{\text{eff}}$, initialisé dans lim_atm.f.

- Variables PRIVATE logical :
 - rad** : La loi $T(\tau)$ est purement radiative, initialisé dans `tdetau.f`.
- Variable PRIVATE character :
 - nom_atm** : Nom de la routine de loi $T(\tau)$ à utiliser, lu dans `lit_n1.f`.
- Routines PRIVATE :
 - coll_atm, edding, eq_atm, hopf, k5750, k5777, lim_atm, lim_gong1, lim_tau1, roger00, roger02, roger05, roger10a, taueff, trho**
- Routines PUBLIC :
 - atm, tdetau, thermo_atm.**

E.9 Module mod_nuc

Le module `mod_nuc` regroupe les routines concernant les réactions thermonucléaires.

- Paramètres PUBLIC integer :
 - m_temp** : Ordre des splines pour l'interpolation des taux des réactions thermonucléaires.
 - niso_tot** : Nombre maximum d'isotopes.
 - nreac_tot** : Nombre maximum de réactions thermonucléaires.
- Paramètres PRIVATE integer :
 - nelem_ini** : Nombre maximum d'éléments.
- Variables PUBLIC double précision :
 - taux_reac** : Taux des réactions thermonucléaires, initialisé dans `taux_nuc.f`.
 - ar** : Masses réduites des cibles des réactions thermonucléaires, initialisé dans `taux_nuc.f`.
 - q0** : Energie des réactions thermonucléaires, initialisé dans `taux_nuc.f`.
 - temp, ttemp** : Table des températures et vecteur nodal associé pour l'interpolation des réactions thermonucléaires, initialisé dans `taux_nuc.f`.
- Variables PRIVATE double précision :
 - ab** : Tableau des abondances, initialisé dans `abon_ini.f`.
 - abon_rela** : Tableau des abondances relatives métal/Z, initialisé dans `abon_ini.f`.
 - m** : Masses atomiques, initialisé dans `abon_ini.f`.

- c** : Charges des éléments chimiques, initialisé dans `abon_ini.f`.
- be7sbe9** : Rapport isotopique $^7\text{Be}/^9\text{Be}$, initialisé dans `abon_ini.f`.
- be7sz** : Rapport isotopique $^7\text{Be}/Z$, initialisé dans `abon_ini.f`.
- c13sc12** : Rapport isotopique $^{13}\text{C}/^{12}\text{C}$, initialisé dans `abon_ini.f`.
- h2sh1** : Rapport isotopique $^2\text{H}/^1\text{H}$, initialisé dans `abon_ini.f`.
- he3she4** : Rapport isotopique $^3\text{He}/^4\text{He}$, initialisé dans `abon_ini.f`.
- he3she4z** : Rapport isotopique $^3\text{He}/^4\text{He}$, avec ^2H primordial transformé en ^3He , initialisé dans `abon_ini.f`.
- li6sli7** : Rapport isotopique $^6\text{Li}/^7\text{Li}$, initialisé dans `abon_ini.f`.
- mg25smg24** : Rapport isotopique $^{25}\text{Mg}/^{24}\text{Mg}$, initialisé dans `abon_ini.f`.
- mg26smg25** : Rapport isotopique $^{26}\text{Mg}/^{25}\text{Mg}$, initialisé dans `abon_ini.f`.
- ne22sne20** : Rapport isotopique $^{22}\text{Ne}/^{20}\text{Ne}$, initialisé dans `abon_ini.f`.
- n15sn14** : Rapport isotopique $^{15}\text{N}/^{14}\text{N}$, initialisé dans `abon_ini.f`.
- o17so16** : Rapport isotopique $^{17}\text{O}/^{16}\text{O}$, initialisé dans `abon_ini.f`.
- o18so16** : Rapport isotopique $^{18}\text{O}/^{16}\text{O}$, initialisé dans `abon_ini.f`.
- age_deb** : Instant du début des chutes de planétoïdes, lu dans `planetoides.f`.
- age_fin** : Instant de la fin des chutes de planétoïdes, lu dans `planetoides.f`.
- dt_planet** : Pas temporel pour décrire les chutes de planétoïdes, lu dans `planetoides.f`.
- t_sup** : Température maximale de tabulation des réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
- mzc_ext** : Masse de la zone convective externe, réinitialisé dans `evol.f`.
- nuzc_ext** : $M^{\frac{2}{3}}$ de la zone convective externe, réinitialisé dans `evol.f`.
- Variables PRIVATE integer :
- izz** : Table des charges des noyaux interagissant dans les réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
- i3al** : Indice de la réaction 3α , initialisé dans `tabul_nuc.f`.
- Variables PUBLIC integer :
- knot_temp** : Dimension du vecteur nodal pour l'interpolation des réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
- nreac** : Nombre de réactions thermonucléaires du réseau utilisé, initialisé dans `tabul_nuc.f`.
- n_temp** : Nombre de températures pour l'interpolation des réactions thermonucléaires, initialisé dans `tabul_nuc.f`.
- Variable PRIVATE character :
- elem** : Noms des éléments chimiques, initialisé dans `abon_ini.f`.

– Routines PRIVATE :

`iben`, `pp1`, `pp3`, `ppcno10BeBFe`, `ppcno10Fe`, `pcno10K`, `ppcno10`,
`ppcno11`, `ppcno12Be`, `ppcno12BeBFe`, `ppcno12Li`, `ppcno12`,
`ppcno3a12Ne`, `ppcno3a9`, `ppcno3aco`, `ppcno9`, `ppcno9Fe`, `rq_reac`,
`tabul_nuc`

– Routines PUBLIC :

`abon_ini`, `nuc`, `planetoides`, `taux_nuc`, `vent`.

E.10 Module `mod_bp_for_alecian`

Le module `mod_bp_for_alecian` regroupe les routines concernant le calcul des accélérations radiatives suivant le second formalisme de G.Alécian. Développé par B.Pichon, il constitue essentiellement une interface entre les routines fournies par G.Alécian et `CESAM2k`. D'origine externe, les fonctions des composantes de ce module ne sont pas détaillées.

E.11 Module `mod_evol`

Le module `mod_evol` regroupe les routines concernant l'évolution temporelle avec et sans diffusion de la composition chimique et du moment cinétique.

– Paramètres PRIVATE double précision :

`eps` : Petit écart permettant l'estimation numérique de dérivées.

`nu_min` : Valeur minimale pour éviter $\nu = 0$.

`t_gab` : Valeur minimale en dessous de laquelle il pourra y avoir mélange dans la routine `diff_t_gab.f`.

`un_eps` : Ecart pour l'estimation numérique de dérivées.

– Variables PRIVATE double précision :

`dr_zc` : Variations des rayons des limites zones radiatives / zones convectives-sur un pas temporel, réinitialisé dans `evol.f`.

`x_mix` : Abscisses des limites zones radiatives / zones convectives, réinitialisé dans `evol.f`.

– Variables PRIVATE integer :

`idis` : Table des indices des discontinuités, réinitialisé dans `evol.f`.

`convd` : Table des indices du début des discontinuités, réinitialisé dans `evol.f`.

`convf` : Table des indices de fin des discontinuités, réinitialisé dans `evol.f`.

`ndis` : Nombre de discontinuités, réinitialisé dans `evol.f`.

`nzc` : Nombre de zones convectives, réinitialisé dans `evol.f`.

n_mix : Nombre de points d'interpolation du vecteur de mélange, réinitialisé dans `evol.f`.

– Variables PUBLIC logical :

mix : Table logique des états de convection, réinitialisé dans `evol.f`.

lw_perte : Il y a perte de moment angulaire, initialisé dans `pertw.f`.

– Routines PRIVATE :

`alecian1`, `coeff_rota3`, `coeff_rota4`, `collision`, `coulomb`, `diffm`, `diffm_br`, `diffm_mp`, `difft`, `difft_gab`, `difft_nu`, `difft_sun`, `diffus`, `diffw`, `diffw_mpz`, `diffw_p03`, `eq_diff_chim`, `eq_diff_poisson`, `eq_diff_rota3`, `eq_diff_rota4`, `f_rad`, `initialise_poisson`, `initialise_u`, `initialise_w`, `lmix`, `pertw`, `pertw_loc`, `pertw_ptm`, `pertw_sch`, `resout_chim`, `resout_rota`, `resout_rota3`, `resout_rota4`.

– Routines PUBLIC :

`coeff_rota`, `ecrit_rota`, `evol`, `initialise_rota`, `rkimps`.

E.12 Module mod_static

Le module `mod_static` regroupe les routines concernées par la résolution des équations de l'équilibre quasi-statique.

– Paramètre PRIVATE character :

nom_qs : Noms des variables quasi-statiques.

– Variables PRIVATE double précision :

fac : Facteurs de répartition, calculé dans `lim_zc.f`.

xcoll : Abscisses des points de collocation, réinitialisé dans `coll_qs.f`.

– Routines PRIVATE :

`coll_qs`, `dgrad`, `lim_zc`, `pertm_ext`, `pertm_msol`, `pertm_tot`, `pertm_waldron`, `static`, `static_m`, `static_r`, `update`.

– Routines PUBLIC :

`pertm`, `resout`, `thermo`.

E.13 Module mod_cesam

Le module `mod_cesam` regroupe les routines concernant la gestion générale du calcul.

- Routines PRIVATE :
add_ascii, ascii, des, des_m, des_r, dnunl, list, output, osc_adia,
osc_invers, osc_nadia
- Routine PUBLIC :
cesam.

E.14 Module mod_exploit

Le module mod_exploit regroupe des routines concernées par l'exploitation des résultats.

- Routine PRIVATE :
ascii, diffw_mpz, diffw_p03, osc_adia, osc_invers, osc_nadia.
- Routines PUBLIC :
add_ascii, ctes_85, ctes_94, ctes_94m, diffw, ini_ctes, inter_atm,
lit_binaire, lit_hr, lit_nl, min_max, min_max_cond, output, read_ascii,
write_nl.

Annexe F

Contenu des sous-directory

F.1 Sous-directory SOURCE

- abon_ini.f** : Initialisation des abondances, routine PRIVATE du module `mod_nuc`.
- add_ascii.f** : Complète l'ensemble des variables pour sorties ASCII, routine PRIVATE du module `mod_cesam`.
- alecian1.f** : Calcul des accélérations radiatives, routine PRIVATE du module `mod_evol`.
- arb_rom.f** : Transforme la notation arabe en romaine, fonction PUBLIC du module `mod_numerique`.
- ascii.f** : Création d'un fichier de sortie ASCII personnalisé, routine PRIVATE du module `mod_cesam`.
- atm.f** : Routine générique de restitution de l'atmosphère, routine PUBLIC du module `mod_atm`.
- base_chim.f** : Routine subordonnée de `diffus.f`, formation de la base pour la diffusion des éléments chimiques.
- base_rota.f** : Routine subordonnée de `evol.f` formation de la base continue non dérivable pour la rotation.
- boite.f** : Dessine une boite de dimensions $\pm dx \times dy$ autour du point de coordonnées (x, y) , routine PUBLIC du module `mod_numerique`.
- box.f** : Dessine une boite asymétrique autour du point de coordonnées (x, y) , routine PUBLIC du module `mod_numerique`.
- bsp1ddn.f** : Interpolation 1D avec une spline polynômiale + dérivées, routine PUBLIC du module `mod_numerique`.
- bsp1dn.f** : Interpolation 1D avec une spline polynômiale, routine PUBLIC du module `mod_numerique`.
- bsp_dis.f** : Interpolation 1D avec discontinuités avec une spline polynômiale, routine PUBLIC du module `mod_numerique`.
- bval0.f** : Calcul les B-splines normalisées en un point de leur domaine, routine PRIVATE du module `mod_numerique`.
- bval1.f** : Calcul les B-splines normalisées et de leurs dérivées 1-ères en un point de leur domaine, routine PUBLIC du module `mod_numerique`.

- bvald.f** : Calcul les B-splines normalisées et de leurs dérivées en un point de leur domaine, routine PUBLIC du module `mod_numerique`.
- cesam.f** : Gestion de l'évolution, constitue en fait le programme principal, routine PUBLIC du module `mod_cesam`.
- cesam2k-dbg.f** : Programme principal pour debug.
- cesam2k.f** : Programme principal pour exploitation.
- cesamT-dbg.f** : Programme principal pour debug de tests.
- cesamT.f** : Programme principal pour exploitation de tests.
- chim_gram.f** : Transforme les abondances par mole en abondances par gramme, routine PUBLIC du module `mod_variables`.
- coeff_rota.f** : Routine générique PRIVATE du module `mod_evol` calcul des coefficients pour la rotation.
- coeff_rota3.f** : Routine PRIVATE du module `mod_evol` calcul des coefficients pour la rotation avec le formalisme de Talon& Zahn 1997, Krot=3.
- coeff_rota4.f** : Routine PRIVATE du module `mod_evol` calcul des coefficients pour la rotation avec le formalisme de Mathis& Zahn 2004, Krot=4.
- coeff_vth.f** : Routine PRIVATE du module `mod_etat`, détermination des coefficients pour l'approximation `vth` de variables thermodynamiques.
- coll.f** : Détermination des abscisses des points de collocation, routine PUBLIC du module `mod_numerique`.
- col_atm.f** : Résolution des équations de l'atmosphère, routine PRIVATE du module `mod_atm`.
- col_qs.f** : Résolution des équations de l'équilibre quasi-statique, routine PRIVATE du module `mod_static`.
- collision.f** : Calcul des intégrales de collision, routine PRIVATE du module `mod_evol`.
- colpnt.f** : Initialisation des abscisses des points de collocation, routine PRIVATE du module `mod_numerique`.
- conv.f** : Routine générique du calcul de la convection, routine PUBLIC du module `mod_conv`.
- conv_a0.f** : Calcul de la convection MLT avec longueur de mélange nulle aux limites ZR/ZC, routine PRIVATE du module `mod_conv`.
- conv_cgm_reza.f** : Calcul de la convection selon Canuto & Mazitelli avec la modification CGM, routine PRIVATE du module `mod_conv`.
- conv_cm.f** : Calcul de la convection selon Canuto & Mazitelli, routine PRIVATE du module `mod_conv`.
- conv_cm_reza.f** : Calcul de la convection selon Canuto & Mazitelli avec la modification de R.Samadi, routine PRIVATE du module `mod_conv`.
- conv_cml.f** : Calcul de la convection selon Canuto & Mazitelli avec la longueur de mélange égale à la plus courte distance des bords de la zone convective, routine PRIVATE du module `mod_conv`.

- conv_jmj.f** : Calcul de la convection MLT, routine PRIVATE du module `mod_conv`.
- coulomb.f** : Calcul du logarithme de Coulomb, routine PRIVATE du module `mod_evol`.
- ctes_85.f** : Initialisation de constantes physiques, routine PRIVATE du module `mod_donnees`.
- ctes_94.f** : Initialisation de constantes physiques, routine PRIVATE du module `mod_donnees`.
- ctes_94m.f** : Initialisation de constantes physiques, routine PRIVATE du module `mod_donnees`.
- delete_doubles** : Elimination des éléments identiques dans un tableau ordonné. Routine PUBLIC du module `mod_numerique`.
- des.f** : Routine générique du dessin au cours de l'évolution, routine PRIVATE du module `mod_cesam`.
- des_m.f** : Dessin en masse au cours de l'évolution, routine PRIVATE du module `mod_cesam`.
- des_r.f** : Dessin en rayon au cours de l'évolution, routine PRIVATE du module `mod_cesam`.
- df_rotx.f** : transformation des dérivées partielles $(P, T, X) \rightarrow (\rho, T, X)$, routine PUBLIC du module `mod_etat`.
- dgrad.f** : Calcul de la différence des gradients, routine PUBLIC du module `mod_variables`.
- difdiv.f** : Calcul des différences divisées, routine PRIVATE du module `mod_numerique`.
- diffm.f** : Routine générique du calcul des coefficients de diffusion microscopique, routine PRIVATE du module `mod_evol`.
- diffm_br.f** : Calcul des coefficients de diffusion microscopique selon Burgers, routine PRIVATE du module `mod_evol`.
- diffm_mp.f** : Calcul des coefficients de diffusion microscopique selon Michaud & Proffit, routine PRIVATE du module `mod_evol`.
- difft.f** : Routine générique du calcul des coefficients de diffusion turbulente, routine PRIVATE du module `mod_evol`.
- difft_gab.f** : Routine PRIVATE du module `mod_evol`, formation du coefficient de diffusion turbulente, suivant une idée de M.Gabriel, on évite la sédimentation de l'hélium en mélangeant si $T \downarrow T_{lim}=1.d6$.
- difft_nu.f** : Calcul des coefficients de diffusion turbulente et radiative, routine PRIVATE du module `mod_evol`.
- difft_smc.f** : Calcul des coefficients de diffusion turbulente en tenant compte de la semi-convection. Routine PRIVATE du module `mod_evol`.
- difft_sun.f** : Routine PRIVATE du module `mod_evol`, formation du coefficient de diffusion turbulente, suivant une idée de M.Gabriel.
- diffus.f** : Gestion des équations de diffusion des espèces chimiques et du moment cinétique, routine PRIVATE du module `mod_evol`.

- diffw.f** : Routine générique du calcul des coefficients de diffusion du moment cinétique, routine PRIVATE du module `mod_evol`.
- diffw_mpz.f** : Calcul des coefficients de diffusion du moment cinétique, selon Mathis, Palacios & Zahn Mathis et al. (2004), routine PRIVATE du module `mod_evol`.
- diffw_p03.f** : Routine PRIVATE du module `mod_evol`, calcul des coefficients de diffusion turbulente pour la rotation D_h , D_v , D_{eff} , suivant Palacios & al 2003.
- dnunl.f** : Calcul approximatif de la grande séparation, routine PRIVATE du module `mod_cesam`.
- ecrit_ascii** : Ecriture des variables de la rotation et des coefficients pour dessins, routine PUBLIC du module `mod_evol`.
- ecrit_rota** : Formation et écriture des fichiers ASCII, routine subordonnée de `cesam.f`.
- edding.f** : Loi $T(\tau)$ d'Eddington, routine PRIVATE du module `mod_atm`.
- eq_atm.f** : Calcul des coefficients des équations de restitution de l'atmosphère, routine PRIVATE du module `mod_atm`.
- eq_diff_chim.f** : Calcul des coefficients des équations de diffusion de la composition chimique, routine PRIVATE du module `mod_evol`.
- eq_diff_rota3.f** : Calcul des coefficients des équations de diffusion du moment cinétique, routine PRIVATE du module `mod_evol`.
- eq_diff_rota4.f** : Calcul des coefficients des équations de diffusion du moment cinétique, routine PRIVATE du module `mod_evol`.
- eq_ini_rota4.f** : Calcul des coefficients des équations différentielles pour l'initialisation de la diffusion du moment cinétique suivant le formalisme de Mathis & Zahn (2004), routine PRIVATE du module `mod_evol`.
- etat.f** : Routine générique du calcul de l'équation d'état, routine PUBLIC du module `mod_etat`.
- etat_ceff.f** : Equation d'état CEFF, routine PRIVATE du module `mod_etat`.
- etat_eff.f** : Equation d'état EFF, routine PRIVATE du module `mod_etat`.
- etat_gong1.f** : Equation d'état GONG1, routine PRIVATE du module `mod_etat`.
- etat_gong2.f** : Equation d'état GONG2, routine PRIVATE du module `mod_etat`.
- etat_mhd.f** : Equation d'état MHD, routine PRIVATE du module `mod_etat`.
- etat_opal.f** : Equation d'état OPAL, routine PRIVATE du module `mod_etat`.
- etat_opalX.f** : Equation d'état OPAL utilisable pour $Z > 0.1$, routine PRIVATE du module `mod_etat`, cf. § 8.53 (Page 194).
- etat_opalZ.f** : Equation d'état OPAL utilisable pour $Z > 0.1$, routine PRIVATE du module `mod_etat`, cf. § 8.53 (Page 194).
- evol.f** : Gestion de l'évolution temporelle de la composition chimique, routine PUBLIC du module `mod_evol`.
- f_rad.f** : Routine générique du calcul des accélérations radiatives, routine PRIVATE du module `mod_evol`.

- fcmax** : Détermine la nécessité d'utiliser le nombre maximum de couche lors du calcul des derniers modèles. Routine subordonnée de `resout` du module `mod_static`.
- fermi_dirac.f** : Intégrales de Fermi-Dirac, routine PUBLIC du module `mod_numerique`.
- from_alecian.f** : package de calcul des accélérations radiatives Alécian2, routine PRIVATE du module `mod_bp_for_alecian`.
- gauss_band.f** : Résolution d'un système linéaire bande, routine PUBLIC du module `mod_numerique`.
- hopf.f** : Loi $T(\tau)$ de Hopf, routine PRIVATE du module `mod_atm`.
- horner.f** : Algorithme de Horner, routine PRIVATE du module `mod_numerique`.
- hsra.f** : Loi $T(\tau)$ HSRA, routine PRIVATE du module `mod_atm`.
- iben.f** : Calcul de l'énergie de contraction selon Iben, routine PRIVATE du module `mod_nuc`.
- ini_ctes.f** : Routine générique d'initialisation des constantes physiques, routine PUBLIC du module `mod_donnees`.
- initialise_rota.f** : Initialisation des variables de la diffusion du moment cinétique, routine PRIVATE du module `mod_evol`.
- initialise_rota4.f** : Initialisation des variables de la diffusion du moment cinétique, suivant le formalisme de Mathis & Zahn (2004), routine PRIVATE du module `mod_evol`.
- integrales.f** : Calcul de la valeur moyenne de la vitesse angulaire et de la variation temporelle du moment cinétique dans les zones convectives suivant le formalisme de Talon et al. (1997), routine PRIVATE du module `mod_evol`.
- inter.f** : Interpolation inverse des variables, routine PUBLIC du module `mod_variables`.
- inter_atm.f** : Interpolation inverse pour l'atmosphère, routine PUBLIC du module `mod_exploit`.
- intgauss.f** : Initialisation des poids et des abscisses pour l'intégration de Gauss, routine PUBLIC du module `mod_numerique`.
- journal** : Description des aménagements dans `CESAM2k`.
- jacob_rota.f** : Calcul des premiers et seconds membres du jacobien pour la résolution du système différentiel non linéaire de la diffusion du moment cinétique, routine PRIVATE du module `mod_evol`.
- k5750.f** : Loi $T(\tau)$ de Kurucz, $T_{\text{eff}} = 5750\text{K}$, routine PRIVATE du module `mod_atm`.
- k5777.f** : Loi $T(\tau)$ de Kurucz, $T_{\text{eff}} = 5777\text{K}$, routine PRIVATE du module `mod_atm`.
- kappa_cond.f** : opacités conductives, routine PRIVATE du module `mod_opa`.
- left_right.f** : Calcul des n fonctions et de leurs dérivées premières de part et d'autre d'un point d'une interpolation B-Spline, routine PUBLIC du module `mod_numerique`.
- lim_atm.f** : Gestion de la restitution de l'atmosphère, routine PRIVATE du module `mod_atm`.

- lim_gong1.f** : Atmosphère monocouche, cas GONG1, routine PRIVATE du module `mod_atm`.
- lim_tau1.f** : Atmosphère monocouche, routine PRIVATE du module `mod_atm`.
- lim_zc.f** : Gestion des limites ZR/ZC, routine PRIVATE du module `mod_static`.
- linf.f** : Recherche de l'encadrement d'une valeur, routine PUBLIC du module `mod_numerique`.
- list.f** : Listing des modèles, routine PRIVATE du module `mod_cesam`.
- list_cesam** : Liste des modules à compiler.
- lit_binaire.f** : Lecture d'un fichier binaire de type `mon_modele_B.*`.
- lit_hr.f** : Lecture des fichiers de type `mon_modele.HR`, routine PUBLIC du module `mod_exploit`.
- lit_nl.f** : Lecture des NAMELISTs, routine PUBLIC du module `mod_donnees`.
- lmix.f** : Fonction PRIVATE du module `mod_evol` détermine si on se trouve dans une zone de mélange pour la composition chimique
- marcs.f** : Lois $T(\tau)$ de MARCS, implantées par B.Pichon, routine PRIVATE du module `mod_atm`.
- matinv.f** : Inversion de matrice, routine PUBLIC du module `mod_numerique`.
- max_local.f** : Maxima locaux de plusieurs tables, routine PUBLIC du module `mod_numerique`.
- min_max.f** : Maximum/minimum d'une table, routine PUBLIC du module `mod_numerique`.
- min_max_cond.f** : Routine PUBLIC du module `mod_exploit` détermination de x_{min} et x_{max} sous condition pour des plots.
- mod_atm.f** : module regroupant les variables et les routines de résolution des équations différentielles pour la restitution de l'atmosphère.
- mod_bp_for_alecian.f** : module regroupant les routines de calcul de l'accélération radiative selon G.Alécian.
- mod_cesam.f** : module contenant ce qui constitue, de fait, le programme principal *i.e.* la routine `cesam` et les routines de gestion.
- mod_conv.f** : module regroupant les routines de convection.
- mod_donnees.f** : module contenant les données du modèle, les constantes physiques et leurs routines d'initialisation.
- mod_etat.f** : module regroupant les équations d'état.
- mod_evol.f** : module regroupant les routines relatives à l'évolution de la composition chimique.
- mod_exploit.f** : module regroupant des routines et programmes pour l'exploitation des modèles.
- mod_kind.f** : module regroupant les définitions de types de variables.
- mod_nuc.f** : module regroupant les routines des réactions nucléaires et d'initialisation de la mixture.

- mod_numerique.f** : module regroupant les outils numériques et informatiques.
- mod_opa.f** : module regroupant les routines d'opacité.
- mod_static.f** : module regroupant les variables et les routines de résolution des équations différentielles pour l'équilibre quasi-statique.
- mod_variables.f** : module regroupant les variables et quelques fonction de calcul de quantités dérivées des variables.
- modif_mix.f** : Routine permettant la personnalisation d'une mixture, routine PRIVATE du module `mod_nuc`.
- neville.f** : Algorithme de Neville, routine PUBLIC du module `mod_numerique`.
- newspl.f** : Changement de base de spline, routine PUBLIC du module `mod_numerique`.
- newspl_gal.f** : Changement de base de spline utilisant la formulation intégrale de Galerkin, routine PUBLIC du module `mod_numerique`.
- newton.f** : formule d'interpolation de Newton, routine PUBLIC du module `mod_numerique`.
- noedif.f** : Création de la base pour la résolution des équations différentielles selon de Boor, routine PUBLIC du module `mod_numerique`.
- noein.f** : Création de la base d'interpolation, routine du module `mod_numerique`.
- noeu_dis.f** : Création du vecteur nodal pour interpolation avec discontinuités, routine PRIVATE du module `mod_numerique`.
- noeud.f** : Création d'un vecteur nodal à partir d'un vecteur de multiplicité, routine PUBLIC du module `mod_numerique`.
- nuc.f** : Routine générique de calcul des réactions nucléaires, routine PUBLIC du module `mod_nuc`.
- opa.f** : Routine générique du calcul d'opacité, routine PUBLIC du module `mod_opa`.
- opa_compton.f** : Routine PUBLIC du module `mod_opa`, Opacités free-free Compton utilisées pour $T \lesssim 7d7K \sim 7Kev$.
- opa_gong.f** : opacité de GONG, routine PRIVATE du module `mod_opa`.
- opa_houdek9.f** : opacité OPAL + Alexander, interpolation de Houdek, routine PRIVATE du module `mod_opa`.
- opa_int_zsx.f** : opacité OPAL, routine PRIVATE du module `mod_opa`.
- opa_opal2.f** : opacité OPAL permettant de dépasser $Z > 0.1$, routine PRIVATE du module `mod_opa`, cf. § 8.82 (Page 210).
- opa_opalCO.f** : opacité OPAL permettant de dépasser $Z > 0.1$, routine PRIVATE du module `mod_opa`, cf. § 8.82 (Page 210).
- opa_yveline.f** : opacité OPAL + Alexander, interpolation de Yveline, routine PRIVATE du module `mod_opa`.
- opa_yveline_lisse.f** : opacité OPAL + Alexander, raccord d'Yveline, routine PRIVATE du module `mod_opa`.
- osc_adia.f** : formation du fichier d'oscillations adiabatiques, routine PRIVATE du module `mod_cesam`.

- osc_invers.f** : formation du fichier ASCII pour inversions, routine PRIVATE du module `mod_cesam`.
- osc_nadia.f** : formation du fichier d'oscillations non-adiabatiques, routine PRIVATE du module `mod_cesam`.
- output.f** : Routine générique de création des fichiers de sortie en ASCII, routine PRIVATE du module `mod_cesam`.
- pause.f** : pause avec commentaire, routine PUBLIC du module `mod_numerique`.
- pertm.f** : Routine générique du calcul de la perte de masse, routine PRIVATE du module `mod_static`.
- pertm_ext.f** : perte de masse linéaire, routine PRIVATE du module `mod_static`.
- pertm_msol.f** : perte de masse linéaire limitée, routine PRIVATE du module `mod_static`.
- pertm_tot.f** : perte de masse linéaire + $E = mc^2$, routine PRIVATE du module `mod_static`.
- pertm_waldron.f** : perte de masse selon Waldron, routine PRIVATE du module `mod_static`.
- pertw.f** : Routine générique de perte de moment cinétique, routine PRIVATE du module `mod_evol`.
- pertw_loc.f** : perte locale de moment cinétique, routine PRIVATE du module `mod_evol`.
- pertw_ptm.f** : perte de moment cinétique associée à une perte de masse, routine PRIVATE du module `mod_evol`.
- pertw_sch.f** : perte de moment cinétique selon Schumanich, routine PRIVATE du module `mod_evol`.
- pgplot_factice.f** : Packtage de routines émulant le logiciel de dessin PGPLOT, extérieur au module `mod_cesam`.
- planetoides.f** : Routine PUBLIC du module `mod_nuc` modifie les dérivées temporelles de la composition chimique calcule le moment cinétique par unité de masse apporté par les planétoïdes.
- plot_rota.f** : Routine subordonnée de `ecrit_rota.f` dessins des variables de la rotation.
- polyder.f** : Algorithme de Horner, routine PUBLIC du module `mod_numerique`.
- pp1.f** : Cycle PP simplifié, routine PRIVATE du module `mod_chim`.
- pp3.f** : Cycle PP, routine PRIVATE du module `mod_chim`.
- ppcno10.f** : Cycles PP + CNO 10 éléments, routine PRIVATE du module `mod_chim`.
- ppcno10BeBFe.f** : Cycles PP + CNO 10 éléments + Fe, routine PRIVATE du module `mod_nuc`.
- ppcno10Fe.f** : Cycles PP + CNO 10 éléments + Fe, routine PRIVATE du module `mod_nuc`.
- ppcno10K.f** : Cycles PP + CNO 10 éléments + K, routine PRIVATE du module `mod_nuc`.

- ppcno11.f** : Cycles PP + CNO 11 éléments, routine PRIVATE du module `mod_nuc`.
- ppcno12.f** : Cycles PP + CNO 12 éléments, routine PRIVATE du module `mod_nuc`.
- ppcno12Be.f** : Cycles PP + CNO 12 éléments + Be, routine PRIVATE du module `mod_nuc`.
- ppcno12BeBFe.f** : Cycles PP + CNO 12 éléments + Be, routine PRIVATE du module `mod_nuc`.
- ppcno12Li.f** : Cycles PP + CNO 12 éléments + Li, routine PRIVATE du module `mod_nuc`.
- ppcno3a12Ne.f** : Cycles PP + CNO + 3α 12 éléments, routine PRIVATE du module `mod_nuc`.
- ppcno3a9.f** : Cycles PP + CNO + 3α 9 éléments, routine PRIVATE du module `mod_nuc`.
- ppcno3aco.f** : Cycles PP + CNO + 3α + C + O, routine PRIVATE du module `mod_nuc`.
- ppcno9.f** : Cycles PP + CNO 9 éléments, routine PRIVATE du module `mod_nuc`.
- ppcno9Fe.f** : Cycles PP + CNO 9 éléments, routine PRIVATE du module `mod_nuc`.
- print_ctes.f** : Ecriture des constantes, routine PUBLIC du module `mod_donnees`.
- read_ascii.f** : Lecture du fichier ASCII, routine du module `mod_exploit`.
- resout.f** : Gestion de la résolution de l'équilibre quasi-statique + évolution temporelle, routine du module `mod_static`.
- resout_chim.f** : Routine PUBLIC du module `mod_evol` résolution par éléments finis Galerkin du système d'équa. diff. ord. non linéaires de la rotation par itération Newton-Raphson.
- resout_rota.f** : Routine générique PRIVATE du module `mod_evol`, résolution du système des équations de la diffusion du moment cinétique.
- rkimps.f** : Algorithme Runge-Kutta implicite, routine PRIVATE du module `mod_evol`.
- roger.f** : Lois $T(\tau)$ Kurucz, routine PRIVATE du module `mod_atm`.
- rq_reac.f** : Calcul des énergies des réactions nucléaires, routine PRIVATE du module `mod_nuc`.
- saha.f** : Equation de Saha, routine PUBLIC du module `mod_etat`.
- schu58_n.f** : Algorithme de Schumaker, routine PRIVATE du module `mod_numerique`.
- shell.f** : Routine de tri, routine PUBLIC du module `mod_numerique`.
- static.f** : Routine générique du calcul des coefficients de l'équilibre quasi-statique, routine PRIVATE du module `mod_static`.
- static_m.f** : Calcul des coefficients de l'équilibre quasi-statique, cas lagrangien, routine PRIVATE du module `mod_quasi_static`.
- static_r.f** : Calcul des coefficients de l'équilibre quasi-statique, cas eulérien, routine PRIVATE du module `mod_quasi_static`.
- sum_n.f** : Intégration d'une spline, routine PUBLIC du module `mod_numerique`.

- tabul_nuc.f** : tabulation des réactions thermonucléaires, routine PRIVATE du module `mod_nuc`.
- tab_vth.f** : tabulation de variables thermodynamiques, routine PUBLIC du module `mod_evol`.
- taueff.f** : pour une loi $T(\tau)$, détermination de l'épaisseur optique où $T = T_{\text{eff}}$, routine PRIVATE du module `mod_atm`.
- taux_nuc.f** : taux des réactions thermonucléaires, routine PRIVATE du module `mod_nuc`.
- tdetau.f** : Routine générique de loi $T(\tau)$, routine PUBLIC du module `mod_atm`.
- thermo.f** : Calcul des variables thermodynamiques pour l'équilibre quasi-statique, routine PUBLIC du module `mod_static`.
- thermo_atm.f** : Calcul des variables thermodynamiques pour l'atmosphère, routine PUBLIC du module `mod_atm`.
- trho.f** : Interpolation en T et ρ pour lois $T(\tau)$ de Kurucz, routine PRIVATE du module `mod_atm`.
- update.f** : passage des variables du temps $t+dt$ au temps t , fonction PRIVATE du module `mod_static`.
- vent.f** : Gestion de la composition chimique du vent, routine PRIVATE du module `mod_nuc`.
- write_nl.f** : Ecriture des NAMELISTs, routine PRIVATE du module `mod_cesam`.
- z14xcotrin21.f** : Package d'interpolation d'opacité, extérieur au module `mod_opa`.
- zoning.f** : Répartition équidistante d'une fonction croissante, routine PUBLIC du module `mod_numerique`.

F.2 Sous-directory EXPLOIT

- 2d-2.pms** : Fichier ASCII d'initialisation de pré-séquence principale, on obtient une température centrale de l'ordre de 100 000K.
- 5d-4.pms** : Fichier ASCII d'initialisation de pré-séquence principale, on obtient une température centrale de l'ordre de 500 000K.
- 8d-5.pms** : Fichier ASCII d'initialisation de pré-séquence principale, on obtient une température centrale de l'ordre de 1 000 000K.
- blabla** : Fichier ASCII d'affectation de la variable logique `blabla` permettant de détourner la plupart des commentaires permettant le suivi des calculs.
- calib2k_pms.f** : Programme de calibration, initialisation sur la PMS.
- calib2k_zams.f** : Programme de calibration, initialisation sur la ZAMS.
- des2k_ZC.f** : Programme de dessin des zones convectives.
- des2k_abon.f** : Programme de dessin des abondances.
- des2k_abontc.f** : Programme de dessin de X, Y et Z au centre, en fonction du temps.

- des2k_abonts.f** : Programme de dessin de X, Y et Z en surface, en fonction du temps.
- des2k_bin.f** : Programme de dessin des variables quasi-statiques en fonction de la masse ou du rayon à partir d'un fichier binaire.
- des2k_diff_osc** : Programme de dessin des différences de 2 fichiers d'oscillations de type `mon_modele-**.osc`, interpolation par `neville.f`.
- des2k_diff_spl** : Programme de dessin des différences de 2 fichiers d'oscillations de type `mon_modele-**.osc`, interpolation par B-Spline.
- des2k_grad** : Programme de dessin des gradients.
- des2k_hr.f** : Programme de dessin de diagrammes HR.
- des2k_opa.f** : Programme de dessin des différences relatives de deux sources d'opacité, le long d'un modèle.
- des2k_osc.f** : Programme de dessin des principales variables d'un fichier d'oscillations de type `mon_modele-ad.osc`.
- des2k_rot.f** : Programme des dessins séparés des variables de la diffusion du moment cinétique à partir du fichier binaire optionnel créé dans la routine `coeff_rota.f`.
- des2k_rot_ext.f** : Programme de dessin, en fonction du temps, de la vitesse de rotation externe en Km/s ou en radian/s à partir d'un fichier HR.
- des2k_vaiss.f** : Programme de dessin de la fréquence de Brunt-Väissälä, à partir d'un fichier binaire de type `mon_modele.B.*`.
- device** : Exemple de fichier de personnalisation des paramètres de dessin "on line", cf. § 4 (Page 53).
- f037_2k.f** : Formation d'un fichier pour le calcul des oscillations à partir d'un fichier binaire de type `mon_modèle.B.dat`.
- fichier_vent.f** : Programme de construction, pour le vent, d'un fichier de composition chimique différente de celle des couches externes.
- langue** : Exemple de fichier de langue.
- m010.zams** : Fichier ASCII d'initialisation de séquence principale d'âge zéro homogène, pour $1M_{\odot}$.
- m020.zams** : Fichier ASCII d'initialisation de séquence principale d'âge zéro homogène, pour $2M_{\odot}$.
- m050.zams** : Fichier ASCII d'initialisation de séquence principale d'âge zéro homogène, pour $5M_{\odot}$.
- mixture** : Exemple de fichier de mixture initiale, différente de celles implémentées dans `CESAM2k`, cf. § 4.2 (Page 54).
- modif_mix** : Exemple de fichier de modification de mixture pour *tous les modèles* calculés dans l'environnement, cf. § 4.4 (Page 56).
- mon_modele.don** : Exemple de fichier de données.
- mon_modele.modif_mix** : Exemple de fichier de modification de mixture pour *le* modèle `mon_modele`.

- mon_modele.rap_iso** : Exemple de fichier de modification des rapports isotopiques pour *le* modèle `mon_modele`
- mon_modele.vent** : Exemple de fichier de vent pour *le* modèle `mon_modele`.
- planet** : Exemple de fichier permettant de définir les paramètres des chutes de planétoïdes, *cf.* § 7.8.2 (Page 116).
- rap_iso** : Exemple de fichier de modification des rapports isotopiques.
- reglages** : Exemple de fichier de réglage.
- sortie_ascii** : Exemple de fichier de personnalisation des sorties ASCII.
- vent** : Exemple de fichier de composition chimique du vent.
- zoom** : Exemple de fichier de personnalisation des dessins "on line".

F.3 Sous-directory TESTS

Certains de ces programmes de test ne sont peut être pas à jour de la toute dernière version de CESAM2k.

- plot_reac_nuc.f** : Programme de dessin des taux des réactions nucléaires.
- test_acc_rad.f** : Programme de test des accélérations radiatives.
- test_atm.f** : Programme de test de l'intégration de l'atmosphère.
- test_bsp_dis.f** : Programme de test de l'interpolation par B-splines avec discontinuités.
- test_cesam.f** : Programme de test de CESAM2k.
- test_coef_diff.f** : Programme de test du calcul des coefficients de diffusion microscopique.
- test_coef_rota.f** : Programme de test du calcul des coefficients de la diffusion du moment cinétique.
- test_collision.f** : Programme de test des coefficients de collision.
- test_colloc.f** : Programme de test de la construction d'un vecteur nodal pour la collocation.
- test_convection.f** : Programme de test d'une routine de convection.
- test_data_ceff.f** : Programme de test des DATA de l'équation d'état CEFF.
- test_data_eff.f** : Programme de test des DATA de l'équation d'état EFF.
- test_der_rotx.f** : Programme de test de transformation de dérivation.
- test_etat/2.f** : Programmes de test d'équation d'état.
- test_jacobien_reac_nuc.f** : Programme de test du jacobien d'un réseau de réactions nucléaires.
- test_lim_ZC.f** : Programme de test de la localisation des zones convectives.
- test_mu_mol.f** : Programme de test de calcul du poids moléculaire moyen.
- test_nl.f** : Programme de test de lecture des NAMELISTs.
- test_opa.f** : Programme de test des opacités.

- test_opa_opal2.f** : Programme de test des opacités OPAL2.
- test_opacite.f** : Programme de test des opacités avec différentes mixtures.
- test_read_osc.f** : Programme de test de la lecture des fichiers ASCII d'oscillations.
- test_rkimps.f** : Programme de test de la routine d'intégration des réactions du réseau nucléaire sans diffusion microscopique.
- test_saha.f** : Programme de test du calcul des taux d'ionisation.
- test_tabul_reac.f** : Programme de test de la tabulation des taux des réactions nucléaires.
- test_tdetau.f** : Programme de test d'un loi $T(\tau)$.
- test_thermo.f** : Programme de test du calcul des principales variables physiques de la structure interne.
- test_thermo_atm.f** : Programme de test du calcul des principales variables physiques de l'atmosphère.

F.4 Sous-directory SCRIPTS

- makefile** : MAKEFILE pour la création de l'exécutble `cesam2k.out` avec le compilateur `lf95`.
- pgplot-5.2.0-2.i686.rpm** : Source RPM de PGPLOT pour LINUX.
- Les deux sous-directory `SCRIPTS_CSH` et `SCRIPTS_BASH` contiennent des shell scripts respectivement en `csh` et `bash`, dont les fonctions sont les suivantes :
- calib2k_pms** : Calibration d'un modèle solaire incluant la PMS.
- calib2k_zams** : Calibration d'un modèle solaire initialisé sur la ZAMS homogène.
- compile2k** : Compilation d'une routine avec le compilateur `lf95`.
- compile2k-dbg** : Compilation d'une routine avec le compilateur `lf95` et options de debug.
- compile2k-dbg_info** : Compilation d'une routine avec le compilateur `lf95`, options de debug et constitution d'une bibliothèque suivant la liste de programmes du fichier `list`.
- compile2k-dbg_list** : Compilation d'une routine avec le compilateur `lf95`, options de debug et constitution d'une bibliothèque suivant la liste de programmes du fichier `list`.
- compile2k_list** : Compilation des routines de la liste du fichier `list` de l'environnement, avec le compilateur `lf95`.
- evol2k_pms** : Evolution d'un modèle incluant la PMS.
- evol2k_zams** : Evolution d'un modèle initialisé sur la ZAMS homogène.
- exe2k** : Exécution d'un programme avec le compilateur `lf95`.
- exe2k-dbg** : Exécution d'un programme avec le compilateur `lf95` et options de debug.
- genere_cesam2k** : Création de la bibliothèque et de l'exécutble `cesam2k.out` avec le compilateur `lf95`.

genere_cesam2k-dbg : Création de la bibliothèque et de l'exécutble `cesam2k-dbg.out` de debug avec le compilateur `lf95`

lib_del : Suppression de routines d'une bibliothèque.

lib_del_repl : suppression et remplacement de routines d'une bibliothèque.

lib_repl : Remplacement de routines d'une bibliothèque.

makefile : MAKEFILE pour la création de l'exécutble `cesam2k.out` avec le compilateur `lf95`.

rempl2k_mod : Remplacement d'un module et formation de l'exécutable pour exploitation.

rempl2k_mod-dbg : Remplacement d'un module et formation de l'exécutable pour debug.

F.5 Sous-directory SUN_STAR_DATA

*****.pms** : Fichiers ASCII d'initialisation de modèles de pré-séquence principale homogène.

Alrd*** : Opacités OPAL "Allard 96".

C95*** : Opacités OPAL "Cox 95".

COX*** : Opacités OPAL "Cox".

EOS*** : Données pour l'équation d'état OPAL.

GN*** : Opacités OPAL "Grevesse Noels".

GS*** : Données pour les opacités OPAL, *cf.* § 8.137 (Page 240).

Gz*** : Données pour les opacités OPAL, *cf.* § 8.137 (Page 240).

HOUDEK_17.03.06.tar.gz : Données pour les opacités Houdek, *cf.* § 2.2.3.1 (Page 16)

IEOS*** : Données pour l'équation d'état OPAL.

W95*** : Opacités OPAL "Weiss", *cf.* § 8.137 (Page 240).

ZFSinterppeos.f : Programme de création de tables d'équation d'état OPAL.

Z_interp_IEOS : Programme de création de tables d'équation d'état OPAL.

ascii2bin_opa : Programme de création de tables d'opacité OPAL en binaire.

datai_gr2.tar.gz : Données pour les accélérations radiatives Alécian2, *cf.* § 8.57 (Page 197).

extract_opa_yveline.explik : Explications pour la création de fichiers d'opacité exploitables par `opa_yveline.f`.

fesh*** : Données pour les tables de loi $T(\tau)$ de type `roger**`, *cf.* § 8.121 (Page 229).

m*.zams.gz** : Fichiers ASCII d'initialisation de modèles de séquence principale d'âge zéro homogène.

mhd1.tab*** : Données pour l'équation d'état MHD, *cf.* § 8.52 (Page 194).

opa_yveline.data.gz : Tables d'opacité Yveline. *cf.* § 8.84 (Page 212)

opal*** : Fichiers ASCII pour la création de tables d'équation d'état OPAL.

opint_v9f.tar.gz : Package d'opacité Houdek, *cf.* § 8.80 (Page 209),

peos*** : Tables pour l'équation d'état OPAL.

phi_psi2*** : Données pour le calcul des accélérations radiatives Alécian1, *cf.* § 8.5 (Page 170).

Bibliographie

- Adelberger, E., et al. : Solar Fusion Rates. *Rev. Mod. Phys.* **70**, 1265–1291 (1998)
- Alécian, G. : Approximate formulae for radiative acceleration in stars. *A&A* **145**, 275–277 (1985)
- Alécian, G., Artru, M.C. : Radiation forces on highly ionized elements in stellar envelopes. *A&A* **234**, 323–331 (1990)
- Alécian, G., Michaud, G., Tully, J. : Radiative accelerations on iron using Opacity Project Data. *ApJ* **411**, 882–890 (1990)
- 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)
- Burgers, J.M. : Flow equations for composite gases. Academic Press, New York and London (1969)
- Canuto, V.M., Mazitelli, I. : Stellar turbulent convection - A new model and applications. *ApJ* **370**, 295–311 (1991)

- Canuto, V.M., Goldman, I., Mazitelli, I. : Stellar Turbulent Convection : A Self-consistent Model. *ApJ* **473**, 550 (1996)
- 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)
- Clayton, D.D. : *Principles of Stellar Evolution and Nucleosynthesis*. Mc Graw-Hill, New York and London (1968)
- Conte, S.D., de Boor, C. : *Elementary Numerical Analysis*. McGraw-Hill Book Company, third edition (1987)
- Cox, J.P., Giuli, R.T. : *Principles of Stellar Structure*, vol I & II. Gordon & Breach, New-York, (1968)
- Cox, A.N., Guzik, J.A., Kidman, R.B. : Oscillations of solar models with internal element diffusion. *ApJ* **342**, 1187–1206, (1989)
- De Boor, C. : *A Practical Guide to Splines* Springer, third ed., Berlin (1978)
- Eggleton, P. : The evolution of low mass stars, *MNRAS* **151**, 351 (1971)
- Eggleton, P. : Composition changes during stellar evolution. *MNRAS* **156**, 361 (1972)
- Eggleton, P.P., Faulkner, J., Flannery, B.P. : An Approximate Equation of State for Stellar Material, *A&A* **23**, 325 (1973)
- Fowler, W.A., Caughlan, G.R., Zimmerman, B.A. : Thermonuclear Reaction Rates, II. *ARA&A* **13**, 69 (1975)
- 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)
- Hairer, E., Wanner, G. : *Solving Ordinary Differential Equations II*, Springer-Verlag, Berlin (1991)
- Heiter, U., Kupa, F., van't Veer-Menneret, C., et al : New grids of ATLAS9 atmospheres I : Influence of convection treatments on model structure and on observable quantities. *A&A* **392** 619(2002)

- Heney L.G., Le Levier R., Levee R.D. : Evolution of Main-Sequence Stars. *ApJ* **129**, 2 (1959)
- Heney, L.G., Vardya, M.S., Bodenheimer, P.L. : Studies in Stellar Evolution III. The Calculation of Model Envelopes. *ApJ* **142**, 841 (1965)
- Henrici P. : 1962, Discrete Variable Methods in Ordinary Differential Equations. John Wiley & Sons, New-York, (1962)
- Houdek, G., & Rogl, J. : On the accuracy of opacity interpolation schemes. *Bulletin of the Astronomical Society of India* **24**, 317 (1996)
- Iben, I. : Stellar evolution I The approach to the main sequence. *ApJ* **141**, 993–1018 (1965)
- Iben, I. : *ApJ* **196**, 525, 1975
- Iben, I. : Thermal pulses, p-capture, alpha-capture, s-process nucleosynthesis, and convective mixing in a star of intermediate mass. *ApJ* **196**, 525–547 (1975)
- Iben, I., MacDonald, J. : The effect of diffusion due to gravity and due to composition gradients on the rate of oxygen burning in a cooling degenerate dwarf. I the case of a thick helium buffer layer. *ApJ* **296**, 540–553 (1985)
- Iglesias, C.A., & Rogers F.J. : Opacities for the solar radiative interior. *ApJ* **371**, 408–417 (1991)
- Kippenhahn, R., Weigert, A., Hofmeister, E. : *Methods of Computational Physics*. **7**, 129 (1968)
- Kippenhahn, R., Weigert, A. : *Stellar Structure and Evolution*, Springer-Verlag Berlin (1991)
- Lang, K.R. : *Astrophysical formulae*, Springer-Verlag Berlin (1968)
- Langer, N., Fricke, K. J., & Sugimoto, D. : Semiconvective diffusion and energy transport, *A&A* **126**, 207–208 (1983)
- Marchouk G., Agochkov, V. : *Méthode des Eléments Finis*. Mir Moscou(1985)
- 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)
- Michaud, G., Proffitt, C.R. : Particle transport processes. In : Baglin & W.W. Weiss (eds.) *Inside the Stars*, San Francisco : ASP conf. ser. **40**, 246–249 (1993)
- Mihalas, D. : *Stellar Atmosphere*. Freeman and Cie, New York (1978)
- Mihalas D., Weibel-Mihalas B. : *Foundations of Radiation Hydrodynamics*, Oxford University Press, Oxford (1984)

- 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)
- Morel, P., van't Veer, C., Provost, J. Berthomieu, G., Castelli, F., Cayrel, R., Lebreton, Y. : Incorporating the atmosphere in stellar structure models : the solar case. *A&A* **286**, 91–102 (1994)
- Morel, P. : CESAM a code for stellar evolution calculations. *A&AS* **124**, 597–614 (1997)
- Morel, P., Thévenin, F. : Atomic diffusion in stellar models of type earlier than G, *A&A* **390**, 611–620 (2002)
- 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 μ gradient in main sequence and sub giant Pop I stars. *A&A* **399** 603–617 (2003]
- Paquette, C., Pelletier, C., Fontaine, G., Michaud, G. : Diffusion coefficients for stellar plasmas. *ApJS* **61**, 177–195 (1986)
- Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T. : Numerical Recipes, Cambridge University Press, Cambridge (1986)
- Proffitt, C.R., Michaud, G. : Diffusion and mixing of lithium and helium in population II dwarfs. *ApJ* **371**, 584–601 (1991)
- 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)
- Thoul, A.A., Bahcall, J.N., Loeb, A. : Element diffusion in the solar interior. *ApJ* **421**, 828–842 (1994)
- Trenoguine V. : Analyse Fonctionnelle. Mir Moscou (1980)
- Schatzman, E., Praderie, F. : Les Etoiles, Ed. InterEditions/Editions du CNRS (1990)
- Schumaker, L. : Splines Functions : Basic Theory. John Wiley, New-York (1981)
- Strittmatter, P.A., Faulkner, J., Robertson, J.W., Faulkner, D.J. : A Question of Entropy. *ApJ* **161**, 369 (1970)
- Stoer, J., Bulirsch, R. : Introduction to Numerical Analysis. Springer-Verlag, Berlin (1979)
- 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)

Wagoner, R.V. : Synthesis of the Elements Within Objects Exploding from Very High Temperatures. *ApJS* **18**, 247 (1969)

Zahn, J.P. : Convective penetration in stellar interiors. *A&A* **252**, 179–188 (1991)