

Binary STARS

a.k.a. **bs**

R. J. Stancliffe

July 21, 2008

1 History

This document sets out to describe the latest generation of the Cambridge version of the evolution code originally developed by [Eggleton \(1971\)](#). The last major overhaul of the code was carried out by [Pols et al. \(1995\)](#) some time ago. Ostensibly, the standard version of the code (and certainly the one on the STARS website, <http://www.ast.cam.ac.uk/~stars>) is from 2000. That still leaves about 6 years worth of modifications to deal with.

2 Modifications

There have been some major changes to the code since the 1995 version. These are detailed below.

2.1 Opacity Updates

As part of his Ph.D. thesis John Eldridge overhauled the opacity tables for the code. These modifications are detailed in [Eldridge & Tout \(2004\)](#). The new tables now use the latest OPAL opacities ([Iglesias & Rogers, 1996](#)) - a change that was long overdue. In addition, the tables also deal with changes in the opacity due to enhancements of carbon and oxygen abundances.

In addition, we now have a means for accounting for molecular opacities. Ideally, we'd want these opacities in the tables but the people who make such things have been slow in supplying them. I decided not to wait and have followed the approach of [Marigo \(2002\)](#). Briefly, this involves the computation of the abundances of the molecules H_2 , H_2O , OH , CO , CN and C_2 in the equation of state and adding their contribution on to the opacity taken from the regular opacity tables. The form of the opacity for the molecules H_2 , H_2O , OH and CO are taken from [Keeley \(1970\)](#), while for CN the polynomial fit of [Scalo & Ulrich \(1975\)](#) is used. The C_2 opacity is assumed to follow that of the CN opacity.

2.2 Nucleosynthesis

The STARS code originally contained routines to follow the evolution of minor elements not important to the structure of a star. These routines fell into disrepair and were only ever

modified to enable the code to compile and run properly. This was changed in 2004 when I resurrected them¹.

The basic idea behind the nucleosynthesis subroutines remains the same. The evolution of 46 elements can be followed. Unlike the main evolution routines, the nucleosynthesis routines make use of analytic derivatives in order to speed up computation.

2.3 Binary evolution

This is perhaps the most extensive rewrite and some aspects of it are incomplete. Most importantly, it allows the solution of the evolution of two stars and their orbital configuration simultaneous. It is based on the ideas in Peter Eggleton's TWIN evolution code, though I have made some changes and not necessarily followed in Peter's footsteps exactly. I have tried to keep the two relative close though (or at least close to the Utrecht variant).

One area in which this version differs from TWIN is that I have added a separate subroutine to deal with the surface boundary conditions that may affect the interaction of the two stars. This subroutine, `massloss.f`, contains the mass loss prescriptions as well as the computations for tidal interactions and angular momentum loss from both the stars and the orbit due to mass being transferred from one star to the other or lost from the system.

2.4 AGB evolution

In order to be able to compute the evolution of asymptotic giant branch stars, a few modifications to the code have had to be made. Most of these came about as a result of my thesis work.

Firstly, as the occurrence of third dredge-up is sensitive to the mixing employed we need to use a mixing coefficient as close to the MLT theory as possible. Of course, we still want the models to be able to converge!

2.5 Extra-mixing mechanisms

3 Input and Output

Dealing with binary stars creates one obvious problem from the outset - you need to double the number of input and output files. I have attempted to rationalise the structure of the unit numbers and these are outlined below. Physical input data start from unit 10. For star 1, the files start in the 30s and work from there. For star 2, the files start in the 50s.

3.1 data – fort.1

Unsurprisingly, the data file has undergone considerable revision. It now looks like:

```
NH2 ITER1 ITER2 JIN JOUT NCH JP ITH IX IY IZ IMODE  
ICL ION IAM IOP INUC IBC ICN IML1 IML2 ISGTH IMO IDIFF  
NVRT1 NVRT2 NVRT3 NVRT4 NVRT5 NSAVE NMONT  
EPS DEL DH0 DT3 DDD
```

¹I should add that my motive for doing this was so that I could get a trip to Vancouver (to attend Nuclei in the Cosmos VIII) out of my Ph.D.

NE1 NE2 NE3 NB NEV NF J1 J2 IH JH
 ID(30) - 3 lines
 NE1 NE2 NE3 NB NEV NF J1 J2 IH JH
 ID(90) - 3 lines
 ISX(45) - 3 lines
 DT1 DT2 CT(10)
 ZS ALPHA CH CC CN CO CNE CMG CSI CFE
 RCD OS RML RMG ECA XF DR
 RMT RHL AC AK1 AK2 ECT TRB
 IRAM IRS1 VROT1 IRS2 VROT2 FMAC FAM
 IVMC TRC1 IVMS TRC2 MWTS IAGB ISGFAC FACSGMIN SGTHFAC

The entries are defined as follows:

NH2 – The desired number of mesh points. If this is different from that in modin, the code should interpolate the given model to give the new one, provided that NCH is greater than or equal to 1. (199)

ITER1 – The maximum number of iterations allowed on the first timestep. (10)

ITER2 – The maximum number of iterations allowed on later timesteps. (10)

JIN – The number of independent variables of the H, DH arrays to be read in. (11)

JOUT – The number of independent variables written out to output models. (11)

NCH – This affects the remeshing of the model. For an NCH of 2 or 3, the model will alter the spacing of the mesh points so that they are optimally spaced according to the mesh spacing function. NCH=3 is used for changing the composition of a model. (1,2 or 3)

JP – This affects the guessed increments for DH for the next timestep. If it is set to 1, the values used are those of the previous timestep. If it is zero, then the increments are set to zero. (0, sometimes 1)

ITH – The thermal energy generation rate is $ITH * T * DS / Dt$. Setting ITH to zero enables $T * DS / Dt$ to be ignored. (1, sometimes 0)

IX – $DX / Dt = IX * (\text{burning rate of Hydrogen})$. Setting this equal to 0 allows the model to ignore composition changes whilst still burning hydrogen. This is useful for creating initial models. (1, sometimes 0)

IY – as for IX, but relates to helium-4 instead. (1, sometimes 0)

IZ – as for IX, but relates to carbon-12 and oxygen-16. (1, sometimes 0)

IMODE - Determines whether the code operates in pseudo-binary mode, solving the equations only for one star (1) or in full binary mode (2).

ICL – If ICL=1, then PRESSI will include the effects of the Coloumb interaction. (1)

ION – Used by STATEF. This is the number of elements that that the routine will calculate ionizations for. (2)

IAM – If set to zero, the program will use integer atomic weights. (1)

IOP – Used to select spline interpolation in opacity. If set to one or less the old style opacity tables are used. If IOP is 1, spline interpolation will be used. If set to zero, a linear interpolation in opacity will be used. Setting this to 5 will use the new opacity tables with variable C/O composition. (5)

INUC – If set to 10 or greater, weak screening is employed by the routine `nucrat.f`. (0)

IBC – No current function.

ICN – Used for CNO equilibrium on the main sequence. If ICN=1, a baryon correction is applied to the hydrogen evolution equation. The timestep used is also affected by ICN=1.

(0)

IML1 IML2 – Specifies the mass-loss rate to use for star 1 and star 2 respectively. Zero is no mass loss, 1 is Reimers mass loss, 2 is Blöcker, 3 is Vassiliadis & Wood and 4 is Wolf-Rayet mass loss.

ISGTH – Turns on thermohaline mixing. Used if you want to compute models with accretion or to provide extra mixing on the giant branch in low mass stars. See also **SGTHFAC**. (1)

IMO – Turns on the molecular opacity routines in STATEF if IMO is set to 1. These are only really important for cool ($T < 10^4$ K) stars that are carbon-rich, such as low-mass AGB stars and carbon-enhanced metal-poor stars. (0)

IDIFF – Turns on gravitational settling, as well as atomic and thermal diffusion. This is probably only important for low-mass stars without significant convective envelopes. (0)

NWRT1 – Prints the internal details of every NWRT1'th model to OUT. (e.g. 100)

NWRT2 – Prints the internal of every NWRT2'th meshpoint when the internal model is printed to OUT. (e.g. 1 or 2)

NWRT3 – Affects the number of 'pages' printed out for every NWRT1'th (i.e. detailed) model. (1-3)

NWRT4 – Prints a short summary of every NWRT4'th model. (e.g. 1 or 4)

NWRT5 – Prints a one-line summary of each iteration of each model, excluding the first NWRT5 iterations of each model. (e.g. 0 or 2)

NSAVE – An output model is saved to MODOUT every NSAVE'th timestep, in the same format as the input model, so that it can be used for a further run. The final model is automatically saved. (300)

NMONT – A detailed model for use as an input file for the **Montage** post-processing code is printed out every NMONT models. If set to zero, no output is produced. (0)

EPS – The accuracy to which SOLVER is expected to solve the equations. EPS must be greater than the DH0 – i.e. you can't solve to a greater degree of accuracy than your derivative. (10-6 or 10-7)

DEL – This is the maximum value in ERR for which the whole correction is applied in SOLVER. Above this limit, the correction applied is reduced by a factor of ERR/DEL. (0.01)

DH0 – Affects the value of the increments of the variables during the numeric differentiation. It is no longer so important, now that everything is in double precision. (10-7)

DT3 – No current function.

DDD – Sets the modulus of the total increment that is desired in one timestep. (0.5 to 4)

NE1 – The number of 1st order equations the code will use. (6)

NE2 – The number of 2nd order equations the code will use. (5)

NE3 – This defines a subset of the 1st order equations that may be defined at 3, rather than two, adjacent meshpoints. At present, this is not implemented in the code. (0)

NB – The number of boundary conditions possessed by the 1st order equations at the stellar surface. (3)

NEV – The number of 'eigenvalues' (i.e. quantities that don't vary with the mesh) used by the model. This is usually the mesh spacing function. (1)

NF – This affects the number of variables being passed into the EQUNS routines. (30)

J1, J2, IH, JH – These variables are used for debugging. Suitable choice gives an output via PRINTC that can be used to see if FUNCS and EQUNS are setting up the difference equations correctly. (0,0,0,99 – this suppresses the debugging output.)

ID(90) – There are two blocks of numbers for ID, one for the evolution package (FUNCS1,

EQUNS1) and the other for the nucleosynthesis package (FUNCS2, EQUNS2). Together with the preceding line (containing NE1, NE2, NB, NEV, NF, J1, J2, IH, JH), these define what order the variables are solved for (first line), what order the equations are solved in (second line) and what order the boundary conditions are solved in (third line). For the evolution of single stars the usual values are: first line (1, 2, 4, 5, 3, 9, 10, 8, 7, 6, 0, 0, 0, 0, 0); second line (6, 7, 8, 9, 4, 2, 1, 3, 5, 0, 0, 0, 0, 0, 0); and the last line (4, 5, 6, 7, 2, 3, 1, 2, 3, 1, 0, 0, 0, 0, 0).

ISX(45) – 3 lines of 15 numbers. These values affect the variables of the internal structure that are printed to OUT. The first 15 values define what will be placed on the first 'page', with the next two sets of 15 defining the output to the extra pages. The options are:

$$\begin{aligned}
 &1 - \psi, 2 - P, 3 - \rho, 4 - T, 5 - \kappa, 6 - \nabla, 7 - \nabla_{\text{ad}}, 8 - \nabla_{\text{rad}} - \nabla_{\text{ad}}, 9 - m, \\
 &10 - {}^1\text{H}, 11 - {}^4\text{He}, 12 - {}^{12}\text{C}, 13 - {}^{14}\text{N}, 14 - {}^{16}\text{O}, 15 - {}^{20}\text{Ne}, 16 - {}^{24}\text{Mg}, 17 - r, \\
 &18 - L, 19 - E_{\text{th}}, 20 - E_{\text{nuc}}, 21 - E_{\nu}, 22 - \delta m, 23 - k^2, 24 - \frac{n}{n+1}, 25 - U_{\text{hom}}, \\
 &26 - V_{\text{hom}}, 27 - U, 28 - S, 29 - \frac{L}{L_{\text{Edd}}}, 30 - \mu, 31 - \mu, \text{ assuming the gas is ideal,} \\
 &32 - \sigma_{\text{thermohaline}}, 33 - E_{\text{bind}}, 34 - \beta = P_{\text{rad}}/P_{\text{tot}}
 \end{aligned}$$

23 is the square of the radius of gyration; 24-26 are homology invariants.

DT1 – Places a lower limit of DT1 * current timestep on the size of the next timestep. (0.8, 0.95 or 1.0)

DT2 – Places an upper limit of DT2 * current timestep on the size of the next timestep. If both DT1 and DT2 are set to 1, then the timestep is constant, unless the model fails to converge (in which case it will become 0.8 times the current value for the next run). (1.2, 1.05 or 1)

CT(10) – These 10 numbers are coefficients used in the mesh spacing function, Q.

ZS – This is the star's metallicity. See the recipes section if you wish to change the metallicity of the model you currently have. (e.g. 0.02, 0.001)

ALPHA – The mixing length. The value chosen should be based on calibration to a solar model. (2.00)

CH-CFE – These are the values for initialising the abundances of ${}^1\text{H}$ through to ${}^{56}\text{Fe}$ of a model. The metals are expressed as a fraction of the total metallicity. These are only used for ZAMS models (or better still, pre-MS models), and with NCH set to 3. (0.7, 0.176, 0.502, 0.092, 0.034, 0.072, 0.072)

RCD – The diffusion coefficient for convective mixing is $\text{RCD} * (\nabla - \nabla_{\text{ad}})^2 / t_{\text{nuc}}$. (106)

OS – A convective overshoot parameter. Zero implies no overshoot. (0)

RML – Used to set the amount of [Reimers \(1975\)](#) mass loss. This value is now just the parameter η . The code multiplies whatever you feed it by 4×10^{-13} before using it in the mass-loss subroutines. (0.4)

ECA – Used in the evolution of EC for pre-MS construction. (0)

XF – Defines the boundary of a core, for printout purposes only, to be when the abundance equals XF. (0.1,0.3)

DR – Defines the boundary between a convection and semiconvection zone, for printout purposes only, to be at $\nabla_{\text{rad}} - \nabla_{\text{ad}} = \text{DR}$. (0.01)

RHL – Defunct. (0)

AC – Defunct. (1.0)

AK1 – Used in the AGB mixing formula (see [Stancliffe et al., 2004](#), for details). Sets β for the H-rich regions. (1.0)
AK2 – Used in the AGB mixing formula. Sets β for the H-poor regions. (10^{-4})
ECT – A constant logarithmic increase or decrease for EC (see the section on MODIN), which can be used to push a ZAMS star back up it’s Hayashi track to make a pre-MS star. (0, 10^{-4})
TRB – Used in setting the surface conditions of binaries. Effectively places the star in a radiation bath of temperature TRB (0).
IRAM – Used to reset the orbital angular momentum. If it is equal to 1, then the code will recalculate the orbital angular momentum using the period from modin. (0,1)
IRS1 – Used to reset the spin angular momentum of star 1. If it is equal to 1, the code will set the spin of star one using the rotational speed provided in VROT1. (0,1)
VROT1 – The desired rotational speed of star one, in kilometers per second. Used only if IRS1=1.
IRS2 – As for IRS1, but for star 2.
VROT2 – As for VROT1, but for star 2.
FMAC – The fraction of matter accreted during mass transfer. Any matter not accreted will be lost from the system, and will carry away angular momentum (1.0)
FAM – The fraction of angular momentum that is transferred in the accreted matter that is transferred to the accreting star. (1.0)
IVMC – Used to turn on the viscous mesh. Used in conjunction with TRC1. Useful for evolving AGB stars and possibly in other situations. (0)
TRC1 – The mesh point at which the code switches from viscous to non-viscous meshing. Note that the transition must be smooth, so there is probably some degree of viscosity at $k=TRC1$. If TRC1 is greater than NMESH, there will be no viscous meshing. It is also unwise to try and lock the majority of the star with the viscous mesh. You have been warned. (700)
IVMS – Used to turn on the viscous mesh at the star’s surface. Used in conjunction with TRC2. This is useful for suppressing loops in the HR diagram for post-AGB objects (and possibly other shell-burning stars) (0).
TRC2 – As TRC1. (300)
MWTS – Unlike the core viscous meshing, which is designed to be invoked at low timesteps, the surface viscous mesh takes a constant mesh weighting. If this is set high enough, the surface mesh points will only move around slowly, thus reducing the problem of numerical diffusion. If you set this to 1, the points will not move at all. This will limit numerical diffusion, but can cause convergence failures. (0.75)
IAGB – Switches on the AGB-specific modification of [Stancliffe et al. \(2004\)](#), including the more realistic mixing, the AGB timestep control and the AGB specific mesh spacing function. Note that you still have to change the coefficients of the mesh spacing function! (0 for nearly all stars or 1 for AGB stars).
ISGFAC – Switches on the reduced mixing convergence aid. Used with FACSGMIN. (0)
FACSGMIN – Sets the initial mixing reduction factor for assisted convergence. Setting this to 1 is equivalent to setting ISGFAC to zero (10^{-2}).
SGTHFAC – Boosts the efficiency thermohaline mixing by a factor of SGTHFAC. It has been suggested that 10^2 is appropriate for getting ^3He -induced extra mixing on the giant branch in low-mass stars. (10^2)

3.2 COtables – fort.10

This file contains the opacity tables used for calculating the opacity with variable C and O composition. The format of the tables is as follows. Each table begins with a row of five numbers. The first three give the hydrogen, helium and metal abundance of the table. The next two numbers give the scaled carbon and oxygen abundance. For carbon, if this number is 0, then there is no carbon enhancement. If it is one, then the abundance is $(1-X-Z)$. Does this strike anyone else as being wrong? Surely, helium should be included in there too?

There then follows 141 lines of 32 numbers. The first number in each line is the temperature in $\log_{10} T$. The remaining numbers are the opacities running from $R(= \rho/T^3) = -8$ to $R = 7$.

3.3 physn.dat - fort.11

This contains the input physical data, including the original style of opacity table, the neutrino loss rates and the charged particle reaction rates. It starts with a single number defining how many opacity tables of different composition there are. This is then followed by a line defining the composition of each of the table using the formula $m = 2X + Y + 1$. The n tables then follow in 10 by 9 blocks. After this comes the neutrino loss rate data, followed by the charged particle reaction rates. These latter have been updated to use the latest rates (as of 2004, that is).

3.4 nrate.dat – fort.13

This contains the reaction rates for the neutron capture reactions used in the nucleosynthesis subroutines. Each set of reaction rates comes as a block of 200 numbers in 20 rows and 10 columns, as with the charged particle reaction data.

3.5 modin and modin2 – fort.30 and fort.50

These are the input model files for two stellar models to be evolved (if desired). They start with a single line of data that gives: the star's mass, the current timestep size, the model's age, the period of the binary (remember, the code always thinks you are dealing with a binary, even if you only care about evolving one star!), the total mass of the binary, an artificial energy generation term, the number of mesh points in the model, the desired number of models to be computed, the starting model number, a number determining whether this is star 1 or star 2 of a binary, the pressure in the H-burning shell and the pressure in the helium burning shell.

There then follows the lines for the model itself. There can be up to 15 variables in this file at present and all are currently used². In order, these are:

1. $\ln f$ - a parameter related to the electron degeneracy
2. $\ln T$ - \ln of the temperature in Kelvin
3. X_O - the mass fraction of ^{16}O

²This is not strictly true. In modin2, the value of the orbital angular momentum is not used. I supposed one could replace this with the eccentricity of the orbit if you wished to include this too (one day, maybe...) though that would mean you could never switch over modin and modin2.

4. $\ln m$ - \ln of the mass in units of 10^{33} g
5. X_H - the mass fraction of ^1H
6. QK, the gradient of the mesh spacing function
7. $\ln r$ - \ln of the radius in units of 10^{11} cm
8. L - luminosity in units of 10^{33}erg s^{-1}
9. X_{He} - the mass fraction of ^4He
10. X_C - the mass fraction of ^{12}C
11. X_{Ne} - the mass fraction of ^{20}Ne
12. X_N - the mass fraction of ^{14}N
13. H_{orb} - the orbital angular momentum in code units
14. H_{spin} - the spin angular momentum of the star
15. $X_{^3\text{He}}$ - the mass fraction of ^3He

3.6 nucmodin and nucmodin2 – fort.31 and fort.51

These are the input model files for the nucleosynthesis routines. The first line of each file is the same as the first line of the modin file (or it should be!). There then follows the lines of the model itself. Each line has 50 numbers to it - these are the abundance by mass fraction of all the elements used by the nucleosynthesis subroutines. In order, these are:

1. g, gallinoes
2. n
3. ^2H
4. ^3He
5. ^7Li
6. ^7Be
7. ^{11}B
8. ^{13}C
9. ^{14}C
10. ^{15}N
11. ^{17}O
12. ^{18}O

13. ^{19}F
14. ^{21}Ne
15. ^{22}Ne
16. ^{22}Na
17. ^{23}Na
18. ^{24}Mg
19. ^{25}Mg
20. ^{26}Mg
21. $^{26}\text{Al}^{\text{m}}$
22. $^{26}\text{Al}^{\text{g}}$
23. ^{27}Al
24. ^{28}Si
25. ^{29}Si
26. ^{30}Si
27. ^{31}P
28. ^{32}S
29. ^{33}S
30. ^{34}S
31. ^{56}Fe
32. ^{57}Fe
33. ^{58}Fe
34. ^{59}Fe
35. ^{60}Fe
36. ^{59}Co
37. ^{58}Ni
38. ^{59}Ni
39. ^{60}Ni
40. ^{61}Ni

41. ^1H
42. ^4He
43. ^{12}C
44. ^{14}N
45. ^{16}O
46. ^{20}Ne
47. Not used...yet!
48. Not used...yet!
49. Not used...yet!
50. Not used...yet!

3.7 out and out2 – fort.32 and fort.52

This file is useful as a reference for what is going on as a star evolves. The file starts with a copy of the data block used for the star’s evolution. Below this is a print out of the first line of MODIN, so you know exactly where the model started from. The structure of the remaining output depends on the values that were chosen in DATA. Typically, the code will produce a short four line summary of the model (the frequency depends on the value of NWRT4), like the one below:

```

447 dt/age/MH/MHe  tn/tKH/Mb  P/rlf/dM LH/LHe/LC Lth/Lnu/m  H1      He4      C12      N14      O16      Ne20      He3      psi      dens      temp      cntr
1.0000 1.977573088E+08 1.067E+10 1.443E+81 9.094E-01-2.764E-04 0.37009 0.60941 0.00001 0.00570 0.00895 0.00198 0.00001 -1.539 2.1457 7.1818 7.1818
0.0001 4.541134415E+09 1.306E+07-1.253E+02 1.620E-33 2.154E-02 0.70000 0.28000 0.00346 0.00112 0.00964 0.00198 0.00003 -15.823 -6.7799 3.7594 srfc
0.0000 0.6645 0.3145 2000.0000 0.000E+00 0.000E+00 0.0001 0.37009 0.60941 0.00001 0.00570 0.00895 0.00198 0.00001 -1.539 2.1457 7.1818 Tmax

```

The first column contains the number of the model (in this case 704), then the mass of the star being evolved, in solar units (1.0000). The next number is the location in mass of the boundary of the hydrogen burning shell, and the last is the location in mass of the boundary of the He-burning zone. In the second column, we have the current timestep of the model and below it the current age of the model - both in years. The two numbers below these are the masses of H and He in the star, in solar units. The next column gives the nuclear and Kelvin-Helmholtz timescales in years, and the mass of the binary system in solar units. In column four, we find the period of the binary system in days, the difference between the stellar and Roche Lobe radii (in units of the stellar radius) and the rate of mass transfer. Next come two columns giving information about the luminosity of the system. First up are the luminosities from hydrogen, helium and carbon (actually, any nuclear burning that isn’t H or He!), followed by the thermal and nuclear luminosities. The units are L_{\odot} and the m at the base of this column is the mass location of the region of maximum temperature. The remaining entries give the abundances of nuclear species at the three points noted in the far right column.

After this block comes another 4 lines providing information about the model. It will take a form something like:

```

0.0000  0.976 -0.976  1.000 -1.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000 -1.1941  0.0753 -0.4571 -0.0009 -0.0110  447
199 199   109  -109    3   -3    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  1.261E+01  7.306E-31  0.000E+00
1      -1      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0

```

On the first line, the first number is the mass interior to the point at which carbon burning occurs. The next twelve are the masses at which the model changes from convective regions to radiative regions (and vice versa). The next 3 numbers are: $\log_{10} mr^2 k^2$ evaluated at some point close to the surface (the moment of inertia of the star³), the homology constant k^2 ,?). Then comes the base-10 log of the radius in solar units, followed by the base-10 log of the luminosity in solar units. The final number is the number of the model.

On the second line, the first number is the mesh point marking the H-shell burning boundary. The next number is the same quantity, but for the He-burning region. The last 12 numbers of the second line give information on where the boundaries between convective and radiative regions are. Boundaries between convective and semi-convective regions are marked by negative signs; convective/radiative boundaries have no sign.

On the third line, the first three numbers are the mass-locations of the points of maximum hydrogen, helium and carbon energy generation. The next 12 numbers are the mass-locations of the boundaries of burning regions. The final three numbers on this lines are the values of the energy generation rates for hydrogen, helium and carbon respectively, at their maximum points.

On the last line, the first three numbers are the the mesh point numbers of the points of maximum hydrogen, helium and carbon energy generation. The next twelve numbers are the mesh point numbers of the location of burning shell boundaries. The last value is $d(\log r)/d(\log m)$.

Every NWRT1th model, the code will output details of the model. The printed details are selected via ISX in DATA.

3.8 plot and plot2 – fort.33 and fort.53

These are output files that are useful for plotting the characteristics of the models as they evolve, or for making Kippenhahn plots of the interior. The entries are:

- 1 - Model number
- 2 - Age
- 3 - $\log_{10} R/R_{\odot}$
- 4 - $\log_{10} T_{\text{eff}}$
- 5 - $\log_{10} L/L_{\odot}$
- 6 - Stellar mass in M_{\odot}
- 7 - Mass of H-exhausted core in M_{\odot}
- 8 - Mass of He-exhausted core in M_{\odot}

³I honestly have no idea why you would want this and I'm not even sure if I've interpreted it right – I may even change it.

- 9 - Hydrogen luminosity
- 10 - Helium luminosity
- 11 - Carbon luminosity
- 12-23 location in mass of the convective boundaries
- 24 location in mass of point of maximum energy generation from H-burning
- 25 - location in mass of point of maximum energy generation from He-burning
- 26 - log-10 of the opacity
- 27 - timestep in years
- 28 - surface abundance of hydrogen
- 29 - surface abundance of helium
- 30 - surface abundance of carbon
- 31 - surface abundance of nitrogen
- 32 - surface abundance of oxygen
- 33 - surface abundance of ^3He
- 34 - radius as fraction of Roche lobe radius
- 35 - Spin angular momentum of star
- 36 - Period of binary in days
- 37 - Separation of binary
- 38 - Binary mass
- 39 - Orbital angular momentum
- 40 - Total spin angular momentum
- 41 - Total angular momentum
- 42 - Angular freq. of orbit
- 43 - Angular freq. of star
- 44 - Moment of inertia of star
- 45 - Moment of inertia of orbit
- 46 - Mass loss rate
- 47-58 Mass location of burning shell boundaries

- 59-70 Location of boundaries of regions where thermohaline mixing is active
- 71 - The mass in the convective envelope (if there is one)
- 72 - The radius of the base of the convective envelope (if there is one)
- 73 - $\log_{10} \rho_{\text{central}}$
- 74 - $\log_{10} T_{\text{central}}$

3.9 modout and modout2 – fort.34 and fort.54

These are where the code outputs models to. To restart a run from a given point, one need only extract the appropriate model from modout and put it into modin. Typically one will simply use the last output model. The format of modout is exactly the same as that for modin. The frequency with which models are output is controlled by NSAVE.

3.10 nucmodout and nucmodout2 – fort.35 and fort.55

Nucmodout is to nucmodin what modout is to modin.

3.11 The synth files – fort.36-38 and fort.56-58

These contain the detailed models for the nucleosynthesis files. There are 3 synth files: a, b and c. The first has the lightest species in it starting from the gallinoes and going up to ^{22}Ne . Synthb has ^{22}Na up to ^{34}S and synthc has the iron group elements as well as the nucleosynthesis code's values of the structure isotopes ^1H , ^4He , ^{12}C , ^{14}N , ^{16}O and ^{20}Ne . All these abundances are plotted with the meshpoint number in the first column and the mass co-ordinate in the last column.

Detailed nucleosynthesis models are printed out with the same frequency and number of mesh points as the detailed structure models. These settings are controlled by NWRT1 and NWRT2. They will not be printed if you are not using the nucleosynthesis routines.

3.12 surface and centre – fort.39-40 and fort.59-60

These files both contain the abundance of all the isotopes in the structure code at the surface and centre, in the order they are listed above in the section on nucmodin. They are preceded by the model number and the age to aid with plotting.

In addition, the surface file also comprises some data that is useful for computing yields. The abundances are followed firstly by the wind mass-loss rate in $M_{\odot} \text{yr}^{-1}$ and then the mass-loss rate from non-conservative mass transfer (or mass lost during common envelope evolution). The final entry is the timestep in years. From all this data, one can compute the yield for every species using a simple Fortran program (which I might have bundled with this package).

3.13 `sprocess` – `fort.41` and `fort.61`

Originally, these were going to be used to supply input to a post-processing code to compute the *s*-process in AGB stars. This has largely been abandoned in favour of using the `Montage` code to do post-processing in general (see the next subsection). However, this file does contain useful information like the temperature in the various burning shells. The entries are as follows:

- 1. The model number.
- 2. The model age in years.
- 3. The mass co-ordinate of the H-burning shell.
- 4. The mass co-ordinate of the He-burning shell.
- 5. The intershell mass (i.e. 3-4, remember this was originally designed for AGB stars!)
- 6. The stellar mass.
- 7. The abundance of ^{14}N in the ashes of the H-burning shell.
- 8-19. The temperature at the convective boundaries, in units of 10^8 K.
- 20-31. The density at the convective boundaries.
- 32. The temperature at the base of the convective envelope.
- 33. The temperature in the H-burning shell (at the point that $X_{\text{H}} = 0.345$).
- 34. The temperature at the base of the H-burning shell (at the point that $X_{\text{H}} = 0.05$).
- 35. The temperature in the He-burning shell (at the point that $X_{\text{He}} = 0.45$).
- 36. The temperature at the base of the He-burning shell (at the point that $X_{\text{He}} = 0.05$).
- 37. The Mira pulsation period in days, based on the pulsation law in [Vassiliadis & Wood \(1993\)](#).
- 38. The absolute value of the mass-loss rate in solar masses per year.

3.14 `montage` – `fort.42` and `fort.62`

This is currently an experimental output. The hope is to be able to take output from the stellar evolution code and feed it in to a post-processing nucleosynthesis routine (specifically Ross Church’s `Montage` program) in order to calculate nucleosynthesis for a wider range of isotopes (crucially including the *s*-process isotopes) than the evolution code can handle (hey, it’s easier than expanding funcs2!).

The output first throws out a line including the number of mesh points, the model number, age in seconds followed by the hydrogen and helium luminosities in L_{\odot} . There then follows a detailed model with the temperature (K), density (g cm^{-3}), mixing length (cm), convective velocity (cm s^{-1}), radius (cm), mass (M_{\odot}) and the ^1H , ^4He , ^{12}C , ^{14}N and ^{16}O abundances printed for each mesh point.

4 Known bugs and issues

The code is not perfect and I do not guarantee its functionality in anyway. The above is only a guide to the code – there is no substitute for you looking at the code and working out what it does yourself!

This section is here to highlight some known issues – you may like to try fixing them yourself!

4.1 The helium flash

The code is currently not able to evolve through the core helium flash. This is my albatross – I was given this problem to do at the beginning of my Ph.D. and though I thought I managed it once, I never got it to work (to my personal satisfaction, none of the Ph.D. students who came after me have manage to do it either) and I still go back to try again every now and then.

References

- Eggleton P. P., 1971, MNRAS, 151, 351
- Eldridge J. J., Tout C. A., 2004, MNRAS, 348, 201
- Iglesias C. A., Rogers F. J., 1996, ApJ, 464, 943
- Keeley D. A., 1970, ApJ, 161, 643
- Marigo P., 2002, A&A, 387, 507
- Pols O. R., Tout C. A., Eggleton P. P., Han Z., 1995, MNRAS, 274, 964
- Reimers D., 1975, Circumstellar envelopes and mass loss of red giant stars. Problems in stellar atmospheres and envelopes, pp 229–256
- Scalo J. M., Ulrich R. K., 1975, ApJ, 200, 682
- Stancliffe R. J., Tout C. A., Pols O. R., 2004, MNRAS, 352, 984
- Vassiliadis E., Wood P. R., 1993, ApJ, 413, 641