1 Introduction

The following is a guide to running the various nonradial pulsation codes. We describe the basic features of each code, including the required input, a brief description of the output, and give some insight to the numerical method used. We use the set of equations listed in Saio & Cox (1980) and we advise you to have a copy of this paper handy to explain some of the symbols we use here. Other general references are Cox (1980) and Unno et al. (1989). The first book has a skimpy coverage of nonradial oscillations, but describes things in physical terms. The latter book gives an encyclopedic coverage of nonradial oscillations, but does not provide as much physical insight.

There are three general nonradial pulsation codes available for use; two of these exist in both an automated version (for batch jobs) and an interactive version that allows you to obtain more detailed results for selected modes. The codes will solve for p-modes, the f-mode, and g-modes, the only difference is the input period guess required.

In the succeeding sections, we describe each pulsation code in turn.

2 ANRO.F (interactive) and ANROAUT.F (batch)

This code uses a General Newton-Raphson (GNR1) relaxation scheme to solve the full fourth order adiabatic nonradial oscillation equations, given a guessed ℓ value and pulsation period. The code requires an input model in the form given by a "prep" code. The prep code reads an equilibrium model, computes additional pulsation quantities, and arranges everything in a format that can be read in by ANRO.F. The input files are tape18.dat and tape19.dat.

Input period guesses may be obtained by computing the discriminant for a given ℓ value. See Osaki & Hansen (1973) for details of how the discriminant is used. When you look at the discriminant output (discr.dat) there are two columns: the trial period and the value of the discriminant. The value of the discriminant goes to zero when the trial period corresponds to an actual eigenperiod of the model. Thus, you can scan a range of periods for each ℓ value, locate the period where the discriminant crosses zero, and use these as guessed periods for the adiabatic pulsation code. This is the procedure I implemented for ANROAUT.f and CJHANRO.f.

The original version of this code was written by N. Baker & K. von Sengbush (1969) using the GNR1 integrator written by N. Baker (handwritten notes). The linear finite difference technique employed by this code gives it second order convergence properties. The code cannot add in extra shells to improve the accuracy of its results. Practical experience suggests the results start to suffer from resolution problems past $k = 10$ (10th radial overtone mode) in models with 200+ shells.

The current version of the code computes the kinetic energy of oscillation associated each mode given by:

$$KE = \frac{1}{2}\sigma^2 \int_0^{\mathrm{M}_\star} |\delta \mathbf{r}|^2 dm$$

It also computes the first order correction to the frequency caused by rotation $C_{\ell,k}$ and the associated corrections to the eigenfunctions. A further description and relevant formulae are given in Saio (1981). As a check on the accuracy of the computed results, we use a variational principle to estimate both the pulsation period and $C_{\ell,k}$ from the converged eigenfunctions. Because we use the converged eigenfunctions and numerically integrate them, we expect the computed errors to overestimate the actual errors. Tests with various stellar models show that the agreement between the eigenvalue (GNR1) period and the integrated (variational) period can show considerable disagreement. Our preferred way to estimate the numerical errors in the period is to change the zoning of the input model and recompute the periods. These kind of tests on white dwarf models suggest that ANRO gives periods good to about 1 % for low order modes.

The output from this code is in the files:

- *discr.dat*: listing of the discriminant used for period guesses.
- *output.dat*: complete listing of pulsation period, eigenfunctions, and rotational correction to the eigenfunctions.
- *summary.dat*: brief listing with pulsation period, kinetic energy, and $C_{\ell,k}$ values.
- *eigen.dat*: lists the eigenfunctions y_1 (radial perturbation) and y_2 (related to horizontal perturbation) versus $-\log(1 - m/\mathrm{M}_\star)$.

contents of ANROIN.DAT

60800 3	Sequence, mixing-length version
80. 600. 260 10	Starting period, ending period, # of points in discriminant, max. number of modes to calculate.
0	Not used (was a print option switch)

- *weight.dat*: lists the weight functions that are proportional to: the kinetic energy density $\mathrm{T}(y,r)$ (most people call this the "weight function"); the square of the acoustic frequency $\mathrm{C}(y,r)$; and the Brunt-Väisälä frequency $\mathrm{N}(y,r)$. All of these are listed versus $-\log(1-m/\mathrm{M}_\star)$.
- *gravity.dat*: lists the eigenfunctions y_3 (grav. potential perturbation), y_4 (related to grav. force perturbation) and the grav. field weight function $\mathrm{G}(y,r)$ versus $-\log(1-m/\mathrm{M}_\star)$.

See Kawaler, Hansen, & Winget (1985) for the form of the weight functions and their application to PNN models.

ANROAUT.F is structurally the same as ANRO.F except that it is automated to compute a set number of modes in a period range read in from ANROIN.DAT I display the format of ANROIN.dat in the table below.

2.1 QANRO.F (interactive)

This code is structurally identical to ANRO.F, with the addition of a subroutine that estimates the pulsational stability of the mode in the quasi-adiabatic approximation. In the quasi-adiabatic approximation, one uses the adiabatic eigenfunctions to compute the temperature and luminosity perturbations. (See Osaki & Hansen 1973 for details.) In real life, I've not used this code much, because the nonadiabatic one is readily available and it doesn't take that long to run.

The qanro subroutine also includes a computation of the gravitational radiation luminosity (L_{gw} for quadrupole $\ell = 2$ modes. Typically, gravitational radiation luminosities dominate for low order p-modes in white dwarf models. More generally, one can expect large values of (L_{gw} whenever y_3 has a large value at the surface of a model, which usually means the mode has a large amplitude in the core.

3 RKFANRO.F (interactive) and CJHANRO.F (batch)

This program also computes the adiabatic pulsation properties of a given input model, but solves the pulsation equations in a different manner than GNR1. This code uses a Runge-Kutta-Fehlberg (4,5) integrator to "shoot" trial solutions from the center to the surface. The code does this in an iterative manner until the built in convergence criteria are met. The guts of the routine (which we call RKF) were written by H.A. Watts and L.F. Shampine of Sandia Laboratories for geophysical applications and adapted for stellar pulsations by C.J. Hansen. The latter originally wrote this code for neutron star calculations; the current version works on any fluid model of a star that is not too centrally condensed. The code has the capability of interpolating the equilibrium model quantities between zones by means of cubic splines to resolve closely spaced modes and satisfy the requirements of the integrator. Because of this feature, the results of this code are insensitive to the initial zoning of the model, so long as no important physics are left out in the original model.

RKF is like GNR1 in that it reads in an input model from a "prep" code (on tape28.dat and tape29.dat) and uses the discriminant for period guesses. It also computes an integrated (variational) period as a check on the internal accuracy, which is 0.01 % at worst in most cases. Modes with very low kinetic energies can show poor agreement between the eigenvalue and integrated period, which is a result of the eigenfunction having a nonzero amplitude in a limited region of the model.

RKF also requires a guess for Y3, which is the guessed value of y_3 at the center. For g-modes, Y3 is a small negative number, starting at about -0.1 for $k = 1$ and moving down to less than -0.0001 for $k > 10$. CJHANRO.F is preloaded with an array of guessed Y3 values appropriate for g-modes. When analyzing p-modes, Y3 is a large negative number, starting at -0.5 for the f-mode, increasing to -0.8 for $k = -1$, and it continues a rapid rise to -3 ($\ell = 1$), -1.5 ($\ell = 2$), and -1 ($\ell = 3$).

The output of these codes is similar to their GNR1 counterparts except that RKF does not calculate the rotational correction to the eigenfunctions and it does print out the values of the various weight function integrands.

The output from this code is in the files:

contents of PERIOD.DAT

60800 3	Sequence, mixing-length version
80. 600. 260	Starting period, ending period, # of points in discriminant,
0	When zero, compute discriminant and obtain period guesses. When 1, read in # of modes, period guesses, and Y3 guesses from DISCR.DAT.

- *discr.dat*: listing of the discriminant used for period guesses.
- *check.dat*: complete listing of pulsation period, eigenfunctions, and rotational correction to the eigenfunctions.
- *results.dat*: brief listing with pulsation period, kinetic energy, and $C_{\ell,k}$ values.
- *eigen.dat*: lists the eigenfunctions y_1 and y_2 versus $-\log(1 - m/\mathrm{M}_\star)$.
- *weight.dat*: lists the weight functions mentioned in ANRO.f
- *gravity.dat*: lists the eigenfunctions y_3, y_4 and the weight function $\mathrm{G}(y, r)$ versus $-\log(1 - m/\mathrm{M}_\star)$.

CJHANRO is structurally the same as RKFANRO except that it is automated to compute a set number of modes in a period range read in from PERIOD.DAT CJHANRO can handle up to 100 period guesses for each ℓ value, although the arrays can be redimensioned to handle more. I also gave CJHANRO the ability to compute the g-mode properties within a given period range for several ℓ values by hard wiring a parameter called “lmax”. Normally, when I run the code, I set lmax to 2 or 3, but one can start and end with pretty much any ℓ value except 0. If you run with more than 3 different ℓ values, you will have to redimension the arrays handling the period guesses and results. CJHANRO reads in the period range and sequence information from PERIOD.dat, whose format is presented below.

contents of NONFLAG.DAT

0 0	(iprint, iplot) These control output of detailed pulsation information and plottable output files. If 0, no output. If 1, you get output.
15	This number is the total number of files with adiabatic results that you want nonadiabatic results for.
70 3	The first number specifies the tape number of the adiabatic results file for that mode; the second number specifies the radial overtone number (k-value) of the mode.

4 NADRS.F (batch)

This program is the one described by Saio & Cox (1980) that solves the fully nonadiabatic pulsation equations for a given stellar model. It uses a complex version of the GNR1 equation solver, which we call GNR1C.

As input, it requires the same equilibrium model files as ANRO (tape18.dat and tape19.dat), input files containing the adiabatic period and eigenfunctions (tape70-99.dat), and a flag file called nonflag.dat. Because NADRS reads in a tape file for each mode, there is a limit on the number of modes one can run through NADRS at any given time. Right now, the upper limit is 30, which utilize the names tape70.dat through tape99.dat. This is a holdover from the good old days of batch jobs and memory limitations of the original computer these programs ran on.

The output files for NADRS and their contents are listed below. I do not include the eigenfunctions y_3 and y_4 in the output, although it is a simple matter to write them out if you want them. For my work, they are generally uninformative.

- *nonout.dat*: Complete listing of nonadiabatic eigenfunctions, work integral information, and various perturbed quantities. Generates about 2000 lines of output for each mode.

- *nonsum.dat*: Summary listing of period, growth rate, and cumulative nonadiabatic quantities.

- *save.dat*: very brief listing with mode id, pulsation period, growth rate, and integrated and eigenvalue stability coefficient values.

- *diff.dat*: lists the work per zone (real) versus $-\log(1 - m/\mathrm{M}_\star)$. Also called the differential growth rate.
- *growth.dat*: lists the cumulative work up to a given zone (real) versus $-\log(1-m/\mathrm{M}_\star)$. At the surface, this is equal to the integrated stability coefficient.
- *y1.dat*: lists the radial perturbation eigenfunction y_1 (complex) versus $-\log(1 - m/\mathrm{M}_\star)$.
- *y2.dat*: lists the dimensionless pressure perturbation eigenfunction y_2 (complex) versus $-\log(1-m/\mathrm{M}_\star)$. This eigenfunction is related to the horizontal perturbation.
- *y5.dat*: lists the entropy perturbation eigenfunction y_5 (complex) versus $-\log(1 - m/\mathrm{M}_\star)$.
- *y6.dat*: lists the radiative luminosity perturbation eigenfunction y_6 (complex) versus $-\log(1 - m/\mathrm{M}_\star)$.
- *deltp.dat*: lists the temperature, density, and pressure perturbations (complex) versus $-\log(1 - m/\mathrm{M}_\star)$.
- *work.dat*: lists the thermal work, nuclear energy generation (or loss) work, and PdV work per zone (complex, except for nuclear work) versus $-\log(1 - m/\mathrm{M}_\star)$.

I urge you to treat the nonadiabatic pulsation information with extreme caution, because the results tend to be very sensitive to the construction of the model and the physics that you do or do not include in the models. The problems arise from several sources, and I will describe them briefly here. I will describe the problems in more detail when I discuss the sample results.

5 REFERENCES

Baker, N., handwritten notes

Baker, N. & von Sengbush, K. 1969, Mitt. Astron. Ges., 27, 162

Cox, J.P. 1980, Theory of Stellar Pulsation, (Princeton: Princeton Univ. Press)

Kawaler, S.D. 1986, Ph. D. thesis

Kawaler, S.D., Hansen, C.J., & Winget, D.E. 1985, ApJ, 295, 547

Osaki, Y. & Hansen, C.J. 1973, ApJ, 185, 277

Saio, H. 1981, ApJ, 244, 299

Saio, H. & Cox, J.P. 1980, ApJ, 236, 549

Unno, W., Osaki, Y., Ando, H., Saio, H., & Shibahashi, H. 1989, Nonradial Oscillations of Stars, (2nd ed.), (Tokyo: Tokyo Univ. Press)