STABILITY MODULE FOR THE NATIONAL TRANSPORT CODE COLLABORATION LIBRARY*

A. Pletzer, J. Manickam, S.C. Jardin, D. McCune, Ch. Ludescher, S. Klasky, L. Randerson and R. Ziemlinski, Princeton Plasma Physics Laboratory

APS
DPP1999
41st Annual Division of Plasma Physics
Seattle
November 15-19 1999

*Work supported by U.S. Department of Energy

Abstract

There is a need to provide numerical tools to the fusion community that are robust, portable, easy to use, documented, and reviewed by independent peers. A web site (http://w3.pppl.gov/NTCC) where modules can be freely downloaded has been set up for that purpose [Status of the NTCC Modules Library (D McCune)]. The existence of such a library is in addition motivated by the increasing demand for programs that can be plugged into large packages with minimal effort. In particular, there has been some requests to make MHD stability codes such as the PEST, which are capable of simulating large scale plasma phenomena, available at the NTCC module library. Progress on the work to convert PEST to satisfy the NTCC module standards is presented. The resulting, new PEST interface is a collection of subroutines, which initialize, modify and extract data. Dynamic memory allocation is introduced to minimize memory requirements and allow for multiple runs. Embedded graphics routines are disabled and dependence on native binary files replaced by portable NetCDF files. To illustrate the flexibility of the module approach, numerical results obtained by integrating PEST-3, the mapping code DMAP and the equilibrium JSOLVER modules into a C++ and Java environment with remote database connectivity are presented.

1. Goal

There has been requests to provide modules capable of performing detailed ideal and nonideal stability calculations as part of larger, more comprehensive computing packages.

For instance, the determination of stability from within the TRANSP transport code system could provide important information about the future time evolution of a plasma.

In nonlinear codes such as the MH3D, a linear stability calculation could provide a valuable estimate for the most unstable waveform, which can then be used as a seed to initialize the perturbation.

Other applications involve large parametric studies where the gradual influence of a parameter requires hundreds of runs.

In order for such a stability module to be widely used, the following criteria have been identified:

- **Robustness**. The module should never crash but return an error flag in case of failure.
- **Portability**. The module should run on all UNIX based workstations and supercomputers and give consistent results up to the last digit.
- **Re-entrant capability**. The package should be able to call the module as often as desired $\Rightarrow$ avoid memory leaks!
- **Ease-of-use**. The module should come with a list of reasonable default parameters, which can be modified upon request.
- **Flexibility**. The size of arrays should be dynamically allocatable to make efficient use of the stack.
- **Reduction of the number of input/output data files**. These can unnecessarily fill up disk space and cause disk crashes. When file communication cannot be avoided then the file format should be binary and platform portable (e.g. netCDF).
- **Accessibility from the NTCC web site**.
- **Ease of maintenance**.

All these criteria contribute to improving the quality of codes. More about the NTCC standards at `http://w3.pppl.gov/NTCC`.

2. Strategy

- Use Fortran 90, the least disrupting approach for PEST3.
- Use IMPLICIT NONE everywhere. This catches many bugs.
- Make REAL(KIND=r8) synonymous to 8-byte long reals. Use default (4- or 8-byte) INTEGERs.
- Use suffix `_r8` to denote "double" constants as in `1.0_r8`.
- Rely whenever possible on freely available, well tested and optimized libraries (LAPACK, BLAS, PSPLINE etc).
- Perform automatic name mapping (e.g. DSCAL on workstations → SSCAL on CRAYs) using PPPL developed FPREPROC tools, also available at `http://w3.pppl.gov/NTCC`.
- Convert all input/output non-portable binary files to NetCDF.
- Prefer the use of the memory heap over scratch files to store large amount of data.

More about portability issues at `http://w3.pppl.gov/~pshare/help/port_strategy_hlp.html`

.

3. Overview of PEST3

PEST3 solves the Sturm-Liouville equation

$$L\xi \equiv -(\partial_\psi \mathcal{D}_\theta + \mathcal{Q}^\dagger)\mathcal{G}(\mathcal{Q} + \mathcal{D}_\theta \partial_\psi)\xi + \mathcal{K}\xi = 0 \quad (1)$$

in toroidal geometry.

Here, L is the linearized force operator and ξ the normal displacement. The "force" is zero because we consider the limit where the growth rate is so small that inertial terms can be neglected.

The operator L has regular singularities at the rational surfaces $q = m/n$ where $\mathcal{D}_\theta = \partial_\theta - inq$ has zero eigenvalues. The asymptotic form of ξ x^α is power-like there with α taking the following values:

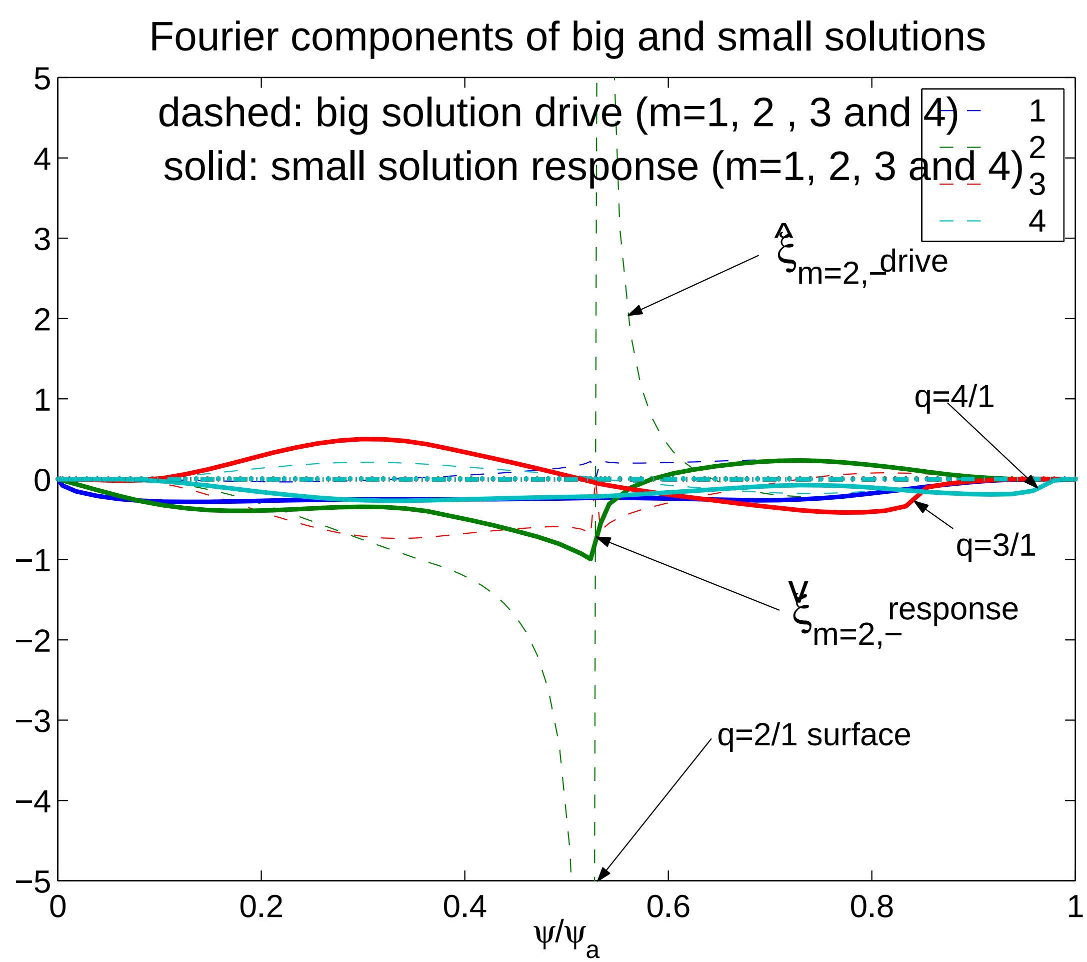
α	'big'	'small'	'regular'
resonant mode $l = m$	$-\frac{1}{2} - \mu$	$-\frac{1}{2} + \mu$	0, 1
nonresonant $l \neq m$	$\frac{1}{2} - \mu$	$\frac{1}{2} + \mu$	0, 1

with $\mu \equiv \sqrt{-D_I}$ related to the Mercier index.

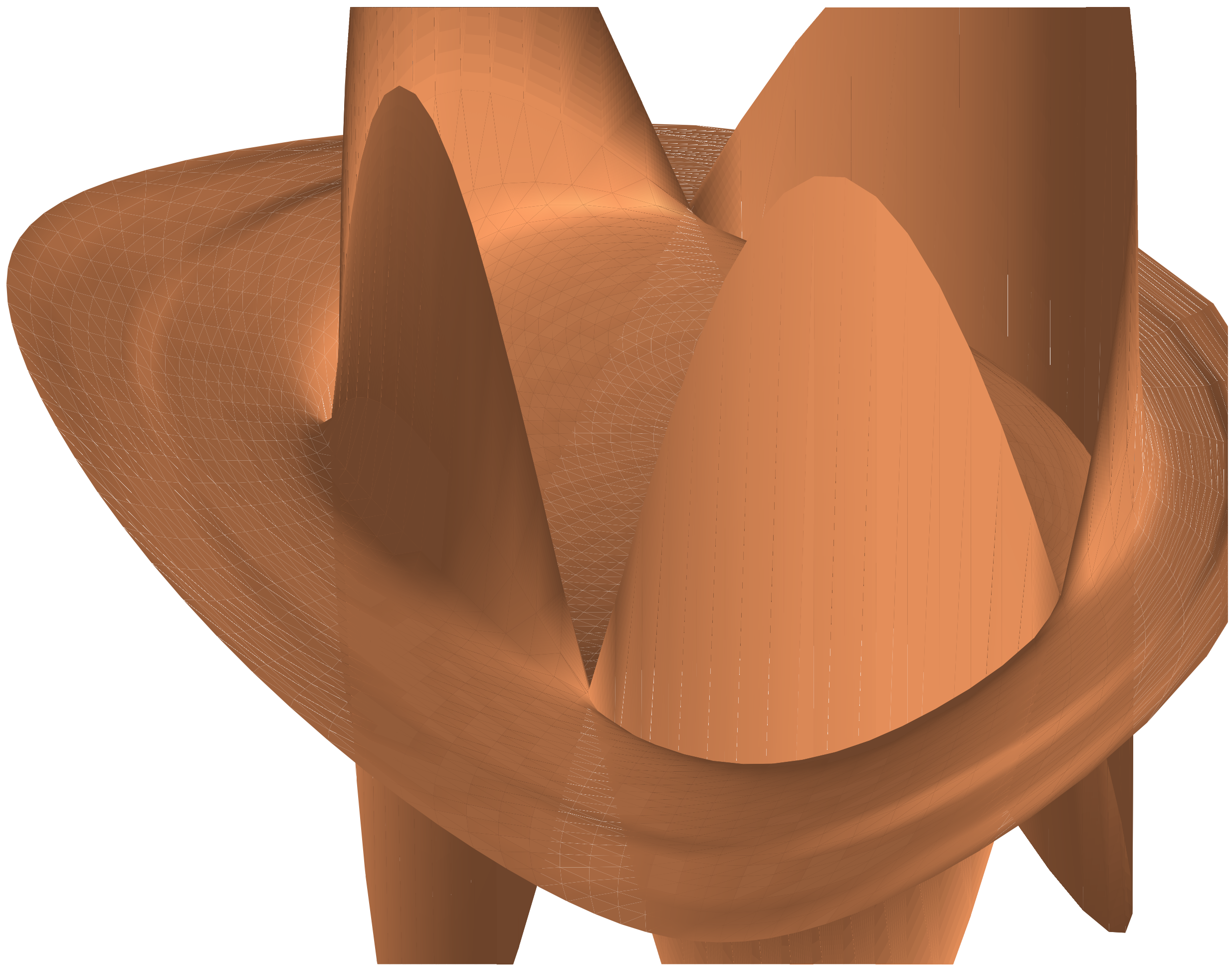
4. Singular solutions

For N rational surfaces there are $2 \times N$ prescribed $\hat{\xi}$'s and computed $\check{\xi}$'s with even and odd parity.

Example of a single helicity big solution and associated small response solution in a finite β plasma:



The total displacement solution for the same case.



5. Matching data

The singularities of the ξ's are resolved by matching to an inner layer model (not part of PEST3). This can be done in a general context given the ratio Δ', B' etc. of small and big solutions about each rational surface $i = 1, \cdots N$.

$$\xi_+ = \sum_i \left\{ \xi_{i+}^{(b)} + \sum_j \xi_{j+}^{(s)} \frac{\mathrm{A}'_{ji}}{2} + \xi_{j-}^{(s)} \frac{\mathrm{B}'_{ji}}{2} \right\} C_{i+}$$

$$\xi_- = \sum_i \left\{ \xi_{i-}^{(b)} + \sum_j \xi_{j+}^{(s)} \frac{\Gamma'_{ji}}{2} + \xi_{j-}^{(s)} \frac{\Delta'_{ji}}{2} \right\} C_{i-}$$

or more concisely

$$\xi = \left(\xi_{ip}^{(b)} + \xi_{jq}^{(s)} D'_{jq,ip} \right) C_{ip}. \tag{2}$$

An example of inner layer model consisting of a single helicity neoclassical tearing mode of width w:

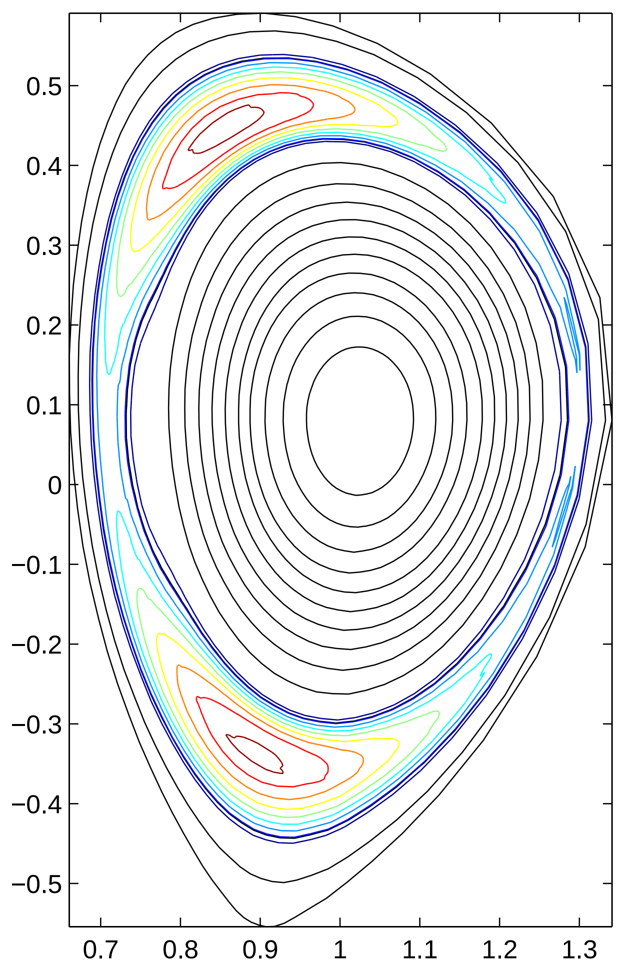
$$
\begin{aligned}
1.6\frac{\tau_R}{r_s}\frac{dw}{dt} &= r_s\Delta' + 1.6\kappa^2 w \log w \\
&+0.8\kappa\left(\Gamma' - \frac{\kappa}{1-2\mu}\right) w + 2.3\sqrt{\epsilon}\beta_p \frac{L_q}{L_p w}
\end{aligned}
\tag{3}
$$

PEST3 computes Δ', Γ' etc.

All other parameters are assumed to be supplied by the user and/or equilibrium code.

6. Tearing mode

Contours of the sum of the equilibrium and perturbed helical fields. The perturbed helical field is an ideal MHD solution from PEST3; its amplitude A being determined by the saturated island width $w(\Delta') \sim \sqrt{A}$.



8. Extended energy principle

Let

$$D' = \begin{pmatrix} \mathrm{A}' & \mathrm{B}' \\ \Gamma' & \Delta' \end{pmatrix} \tag{4}$$

denote the full $2N \times 2N$ matching matrix. It can be shown that D' derives from an energy principle:

$$D'_{ip,jq} = W(\check{\xi}_{ip}, \check{\xi}_{jq}) + \tfrac{1}{2}(\hat{\xi}_{ip}, L\hat{\xi}_{jq}) \tag{5}$$

where $i, j = 1, \cdots N$ and $p, q = \pm$ for the parity.

$$W(\check{\xi}_{ip}, \check{\xi}_{jq}) : \text{ideal energy function} \tag{6}$$

We have $W > 0$ definite for ideal stability. Hence PEST3 also determines ideal stability.

$$\tfrac{1}{2}(\hat{\xi}_{ip}, L\hat{\xi}_{jq}) : \text{nonideal correction to the energy} \tag{7}$$

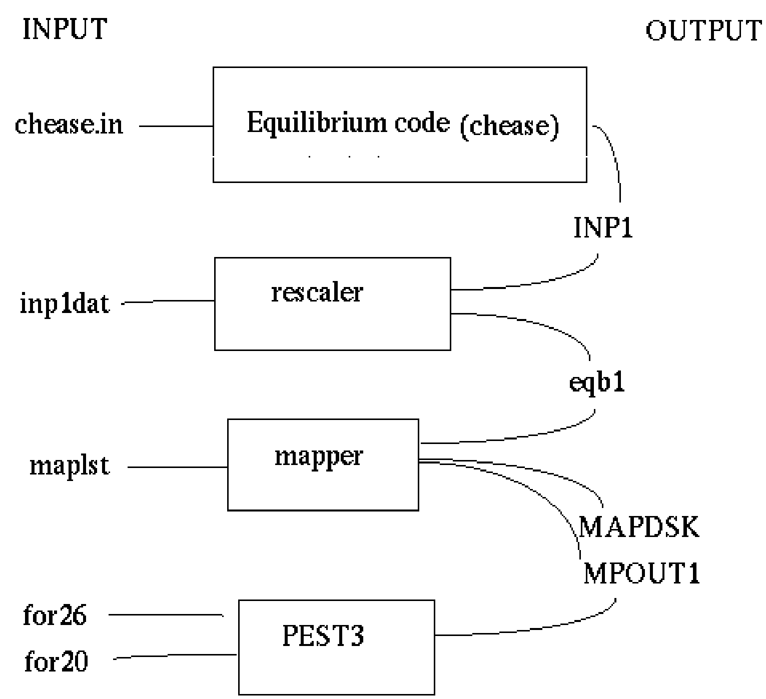
7. Old PEST3 layout

The traditional PEST design suffers from a number of shortcomings.

A complete stability computation typically involves running 3-4 codes in sequence. These codes communicate through I/O files. Some parameters need to be matched across codes.

The files MAPDSK and MPOUT1 are binary and contain various equilibrium and metric information with some redundancy. Unfortunately, these files, written using a set of routines called BZIO, are not portable. Furthermore the source code for the BZIO is not portable so that there are presently at least 3 versions of BZIO for CRAYs, ALPHAs and other workstations coexisting. Some data are default integer so that the resulting MAPDSK and MPOUT1 files can only be read provided the same compiler switches are used across the sequence of programs.

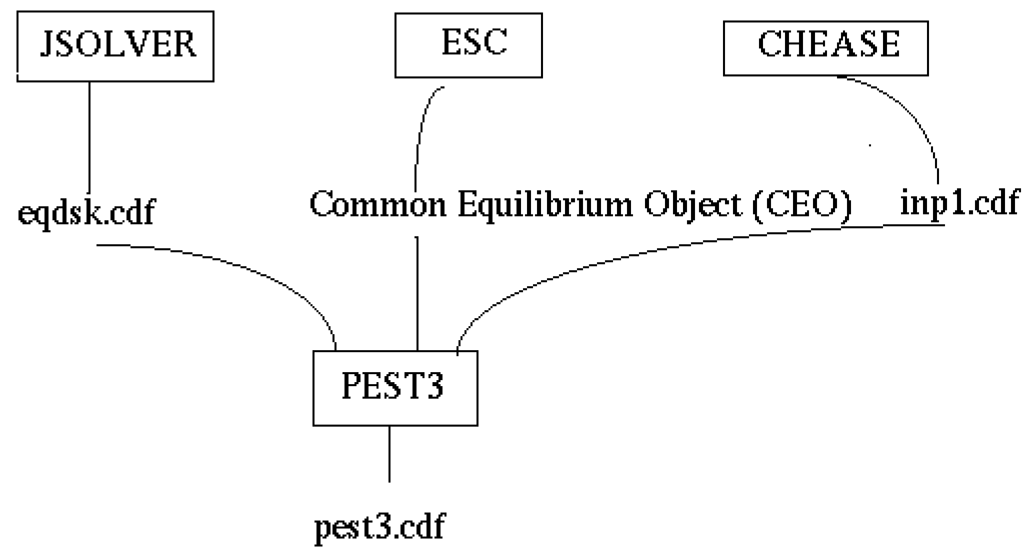
INP1 and eqb1 are unformatted fortran binary files. All other input files are ASCII namelists.


INPUT
OUTPUT
chease.in
Equilibrium code (chease)
INP1
inp1dat
rescaler
eqb1
maplst
mapper
MAPDSK
MPOUT1
for26
for20
PEST3

8. Module PEST3 design

To overcome the shortcomings of the traditional PEST design we want:

- PEST3 to take a minimal, primitive set of data from various equilibrium codes as input (avoiding redundancies which complicate the interfacing to other equilibrium codes).
- The Mapper should be called from within PEST3 (thus avoiding the creation of the intermediate files MAPDSK and MPOUT1).
- Dimensions should be adjusted automatically, thus drastically reducing common user errors.
- Input/output data should be passed as subroutine arguments rather than via namelists.



All output files are in netCDF format and produced by EZCDF—an easy-to use set of routines developed at PPPL. There is no loss of precision since the format is binary. Many visualization packages (AVSExpress, IDL, Matlab, OpenDX etc) can directly access these data.

10. Example of PEST3 calls

```
character*10 filename
integer, parameter :: r8=selected_real_kind(12,100), ns = 5
integer, dimension(5) :: msin(ns)
real(r8), dimension(5) :: deltap(ns,ns)
...
call pstInit ! initialize
!
! set input parameters
!
call pstSetN(1)      ! toroidal mode number (optional)
call pstSetLmax(10) ! poloidal modes range from -10...+10 (optional)
call pstSetM(200)    ! number of radial modes (optional)
msin = (/0, 1, 0, 0, 0 /) ! tag resonant surfaces where reconnection occurs
call pstSetMsin(ns, msin) ! (optional)
```

```
!
! execute
!
call pstExec
!
! access the results
!
call pstGetDeltap(ns, deltap)
!
! clean up
call pstFree
...
```

11. Summary

This work describes the present status of the transformation of PEST3 from a research tool that 'works sometimes' to a production tool that should give consistent results.

In the process of becoming more resilient, it is hoped that PEST3 will become more widely used. A greater exposure will most certainly benefit PEST3 in turn through the many suggestions for improvement and through the hands of other programmers. The success of this process relies on a good modular design.

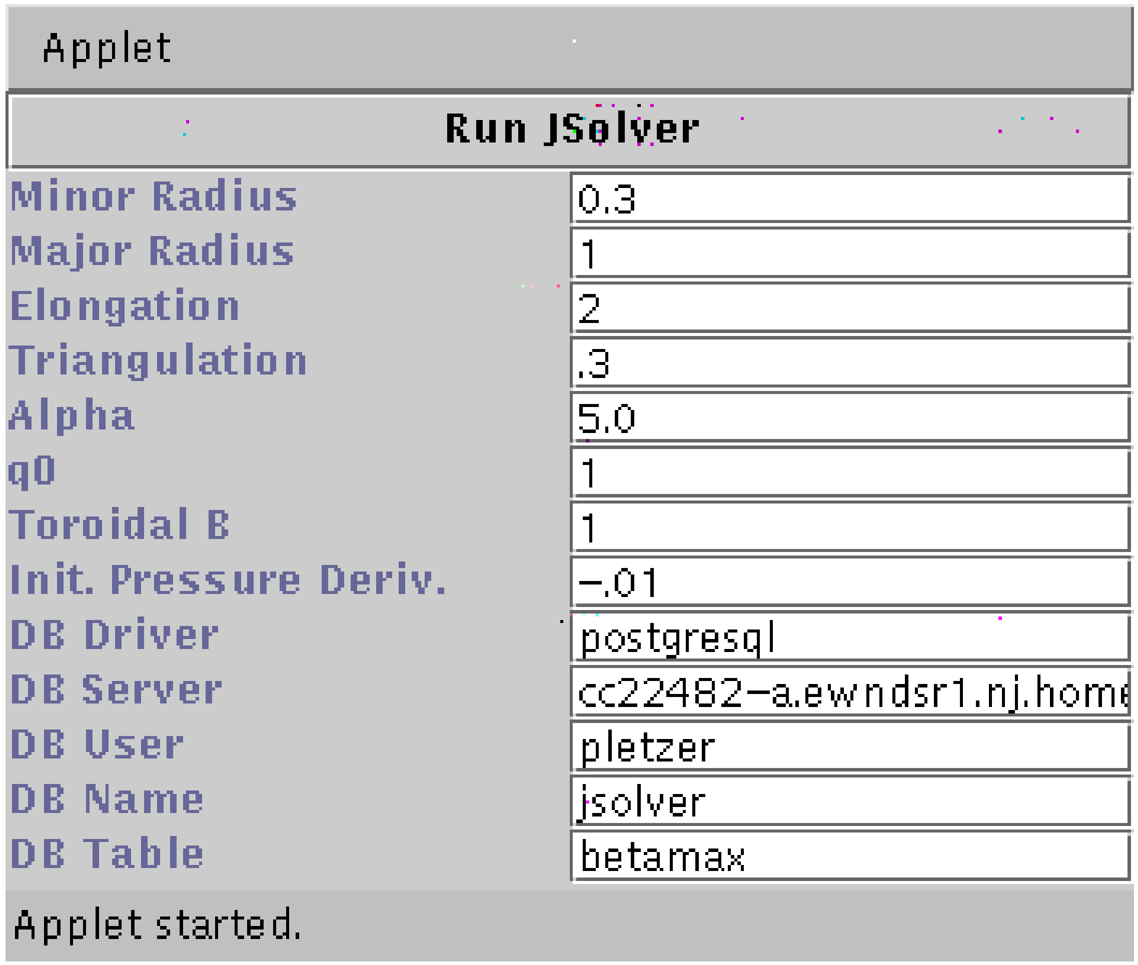
The future of computing in plasma physics can be speculated from the observation that programming algorithms tend not to evolve at the same speed as hardware. Thus, the longevity of codes should not be underestimated. It is most likely that the algorithms used in 20 years will be very similar to the ones used today. However, the increasing computational power already available now allows us to integrate what used to be standalone applications into computing environments such as TcL/TK, Python, Java etc. In the present work PEST3 reduces to a mere (shared) library that can be **dynamically linked**. As an example we give here a GUI Java applet interface that sends MHD equilibrium data to a remote SQL database on a home PC.

The new version of PEST3 and the mapper DMAP has been tested on SUN, ALPHA and LINUX (Nag f95). The dependencies on namelists and BZIO files have been replaced by netCDF files. Interfaces to equilibrium codes need, however, to be further developed and carefully tested. The rate of progress will depend on the users' interest and help (please send an email to pletzer@pppl.gov if you are interested to contribute). The estimated cost to produce a downloadable beta version is 3 man-months.

On the physics front it is worthwhile to note that the asymptotic matching procedure generally fails for the $m = 1/n = 1$ mode, which tends to be ideally unstable although experiments suggest otherwise. The reason may be that kinetic effects play here an important role. Perhaps the most natural way to add these effects is in the form of an energy correction to the extended energy functional.

References and related efforts:

- Official PEST3 reference. Pletzer, Bondeson & Dewar, J. Comput. Phys. 115, 530 (1994)
- Recent advances in distributed computing (this conference). [UP1.60] Status of NTCC Python Physics Server
- A library of plasma physics codes (this conference). [UP1.77] National Transport Code Collaboration (NTCC)


Applet
Run JSolver
Minor Radius
0.3
Major Radius
1
Elongation
2
Triangulation
.3
Alpha
5.0
q0
1
Toroidal B
1
Init. Pressure Deriv.
-.01
DB Driver
postgresql
DB Server
cc22482-a.ewndsr1.nj.home
DB User
pletzer
DB Name
jsolver
DB Table
betamax
Applet started.