PIES Code Improvements

Don Monticello, Allan Reiman,
and Peter Merkel

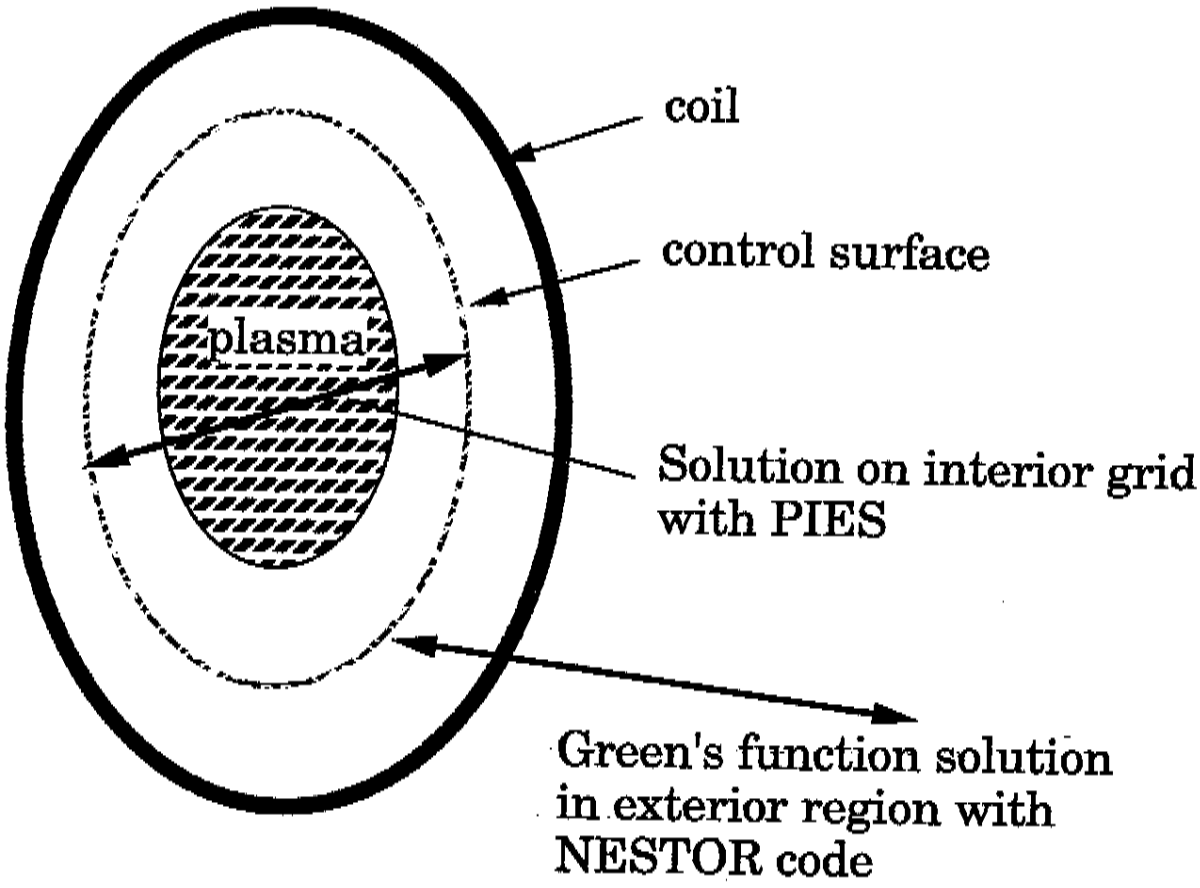
Outline

- New Free Boundary Algorithm
- Improved Treatment of Out-of-Phase Islands
- Bootstrap Model
- Porting to Various Platforms
- F90 Version
- Future Plans

New Free Boundary Algorithm

- A new, non-iterative, free-boundary algorithm has been implemented and first tests indicate the implementation has been successful. More testing is needed.
- Benchmark runs on W7X model show significant improvement over previous free-boundary algorithm.

Free-Boundary PIES



Match exterior Green's function solution to interior solution at "control" surface. Want **B** continuous across control surface.

Boundary Conditions at Control Surface

- Control surface is computational boundary for interior PIES solution on grid. Interior solution determined by $\hat{\mathbf{n}} \cdot \mathbf{B}$ on boundary.
- Boundary condition for exterior solution: $\hat{\mathbf{n}} \cdot \mathbf{B}$ on control surface, $B \rightarrow 0$ at ∞.
- $\hat{\mathbf{n}} \cdot \mathbf{B}$ continuous at control surface.
- Solutions give $\hat{\mathbf{n}} \times \mathbf{B}$ outside and inside control surface. Jump in $\hat{\mathbf{n}} \times \mathbf{B}$ = surface current in control surface.
- Adjust $\hat{\mathbf{n}} \cdot \mathbf{B}$ to zero out surface current.

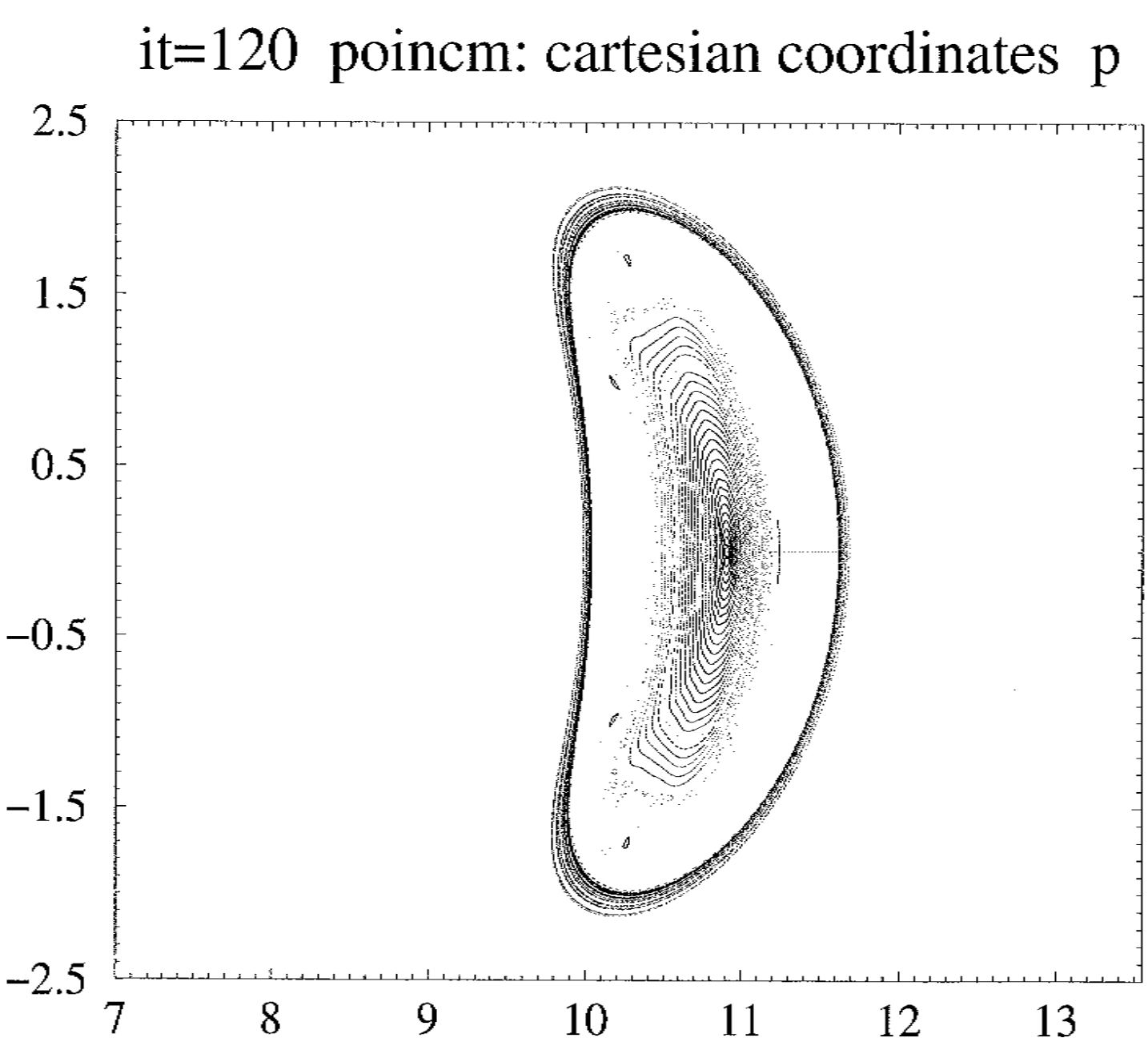
Use Superposition to Adjust $\hat{\mathbf{n}} \cdot \mathbf{B}$

$$\nabla \times \mathbf{B} = \mathbf{j}_{\text{plasma}} + \mathbf{j}_{\text{coils}} + \mathbf{j}_{\text{surface}}, \quad B \to 0 \text{ at } \infty$$

- Want to subtract off $\mathbf{B}_{\text{surf}}$ (where $\nabla \times \mathbf{B}_{\text{surf}} = \mathbf{j}_{\text{surface}}$, $B_{\text{surf}} \to 0$ at ∞).
- Solve for $\mathbf{B}_{\text{surf}}$ using Green's function. Take $\hat{\mathbf{n}} \cdot \mathbf{B} \to \hat{\mathbf{n}} \cdot \mathbf{B} - \hat{\mathbf{n}} \cdot \mathbf{B}_{\text{surf}}$.

Old method: uses $\hat{\mathbf{n}} \cdot \mathbf{B}_{\text{surf}}$ from previous iteration.

New method: does two Ampere solves to get self-consistent $\hat{\mathbf{n}} \cdot \mathbf{B}$ in each iteration. Stored matrix inverses minimize additional cost.



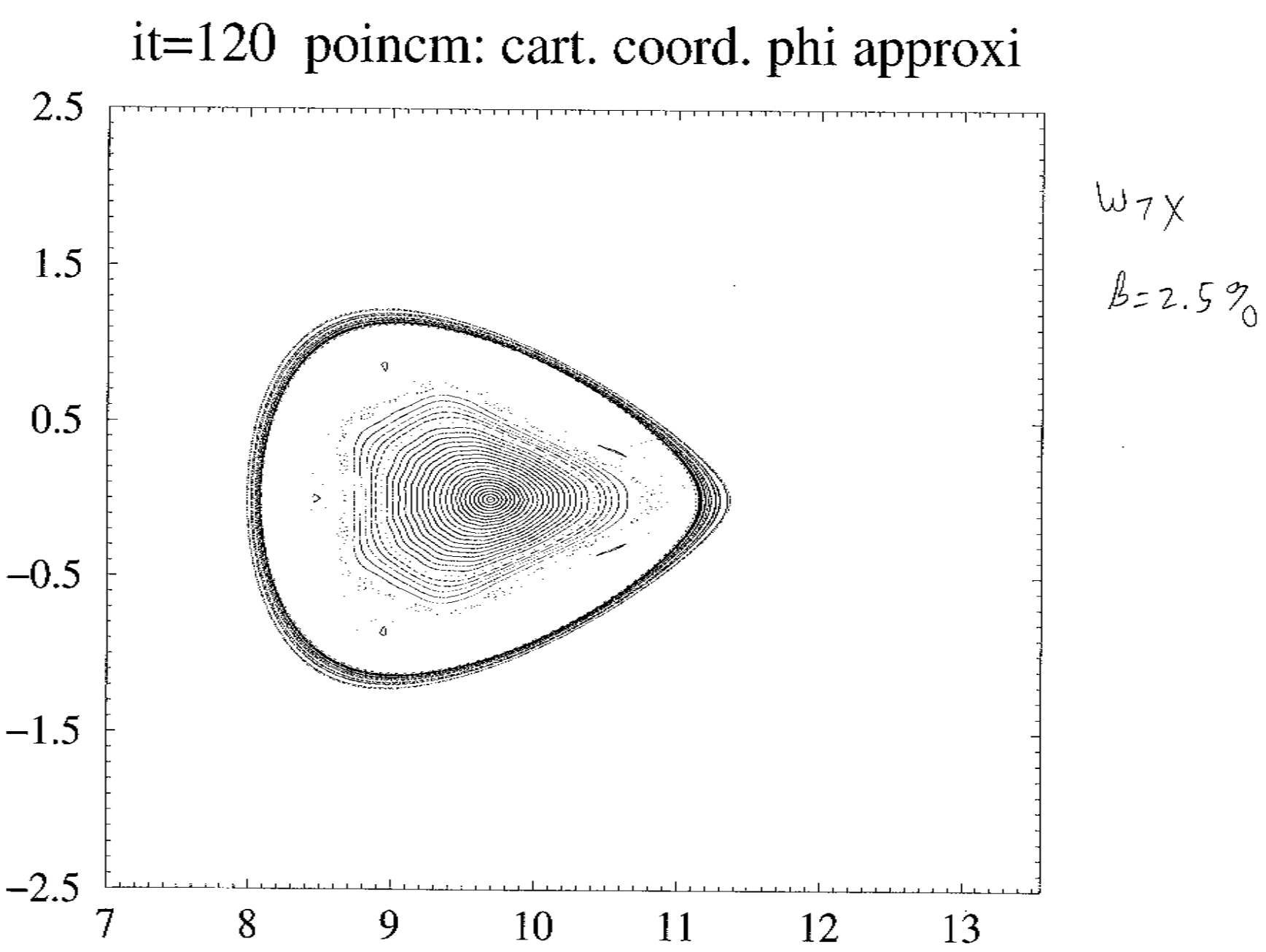
W7X

$\beta = 2.5\%$

it=120 poincm: cartesian coordinates phi=0

p65_120

Plot 139



$w_7 X$

$\beta = 2.5\%$

it=120 poincm: cart. coord. phi approxi. eq. pi/nper

p65_120

Plot 141

Improved Treatment of Out-of-Phase Islands

- A new treatment of islands with x-points on the mid-plane has been implemented and debugged. This algorithm treats the in-phase and out-of-phase islands more symmetrically than the old algorithm.
- This new treatment is necessary for the implementation of a bootstrap model for current within the islands. More testing is of the algorithm is needed.

Bootstrap Model

- A model of the bootstrap effect has been implemented and debugged. This model zeroes out the current flowing in the islands and should lead to smaller equilibrium islands.
- More testing is required.

Porting to Various Platforms

- The PIES code has been successfully ported to the PPPL Dec-Alpha workstations.
- These Dec-Alpha workstations are twice as fast as the j90 machines. The latest version of the Dec-Alpha workstation, which we are acquiring, is another factor of two faster than its predecessor.

F90 Version

- Considerable progress has been made on the conversion of PIES from RATFOR to F90. The first phase should be completed by mid-April.
- This conversion will greatly aid the portability of the PIES code to various platforms. In addition, new users will find the code more readable.

Future Plans

- Application of PIES to NCSX using the new free boundary algorithm and the new bootstrap model
- Application of PIES to QOS
- Benchmarking with VMEC, M3D and HINT