Improving the Usability of Physics Modeling Codes

Making the rant3d code faster and easier to use

Katherine White
University of Tennessee, Knoxville

Mentor: Mark Carter, Fusion Energy Division,
Oak Ridge National Laboratory

Purpose of the rant3d Code

- The rant3d code models antennas used for heating plasma in controlled nuclear fusion experiments.
- This model keeps track of the geometry of recesses in the antenna, and current straps within the recesses.
- The code solves Maxwell's equations for connected rectangular waveguide sections (the recesses) with a plasma boundary.

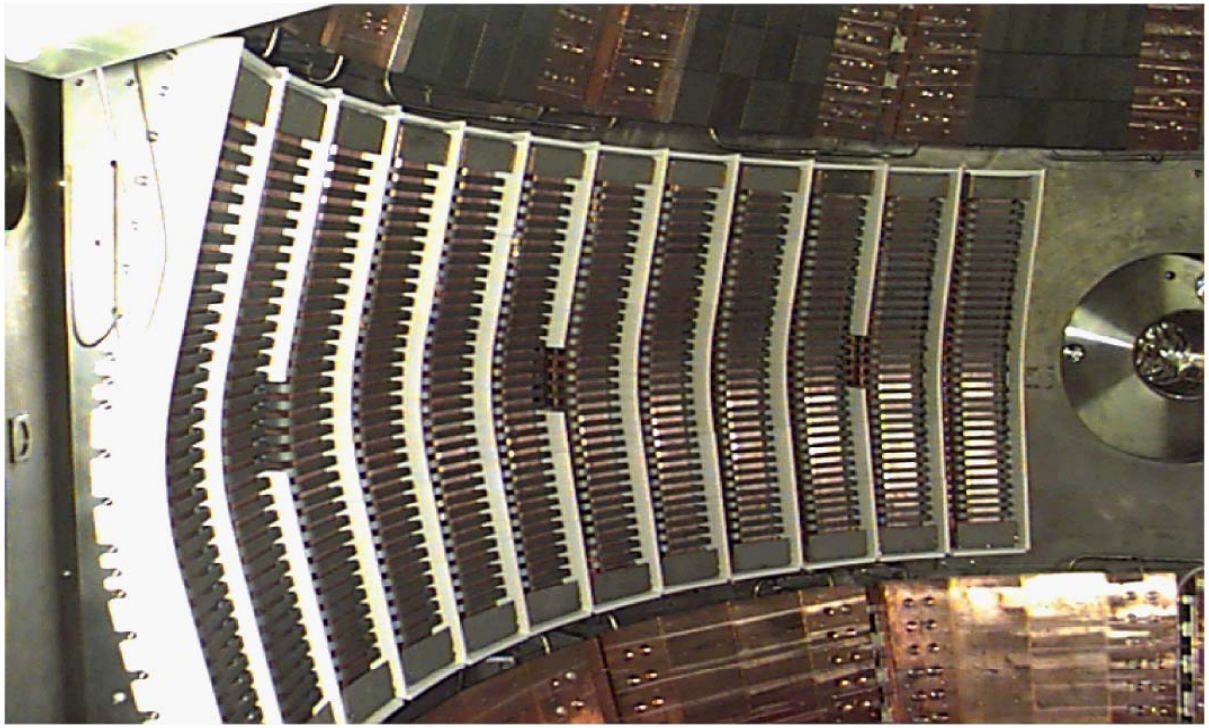
Issues addressed for the rant3d code

- Performance optimization
  - Performance was improved by analytically solving Fourier transforms and replacing the two dimensional fast Fourier Transform methods in the code.
  - Timings of the code were taken using various performance improvement parameters.
- User interface for code inputs
  - Previously inputs were made in a text file and were difficult to read and understand
  - A Graphical User Interface (GUI) was written in Java
- Portability
  - The rant3d code was ported from IBM AIX to Linux.
  - The GUI was tested on several different operating systems.

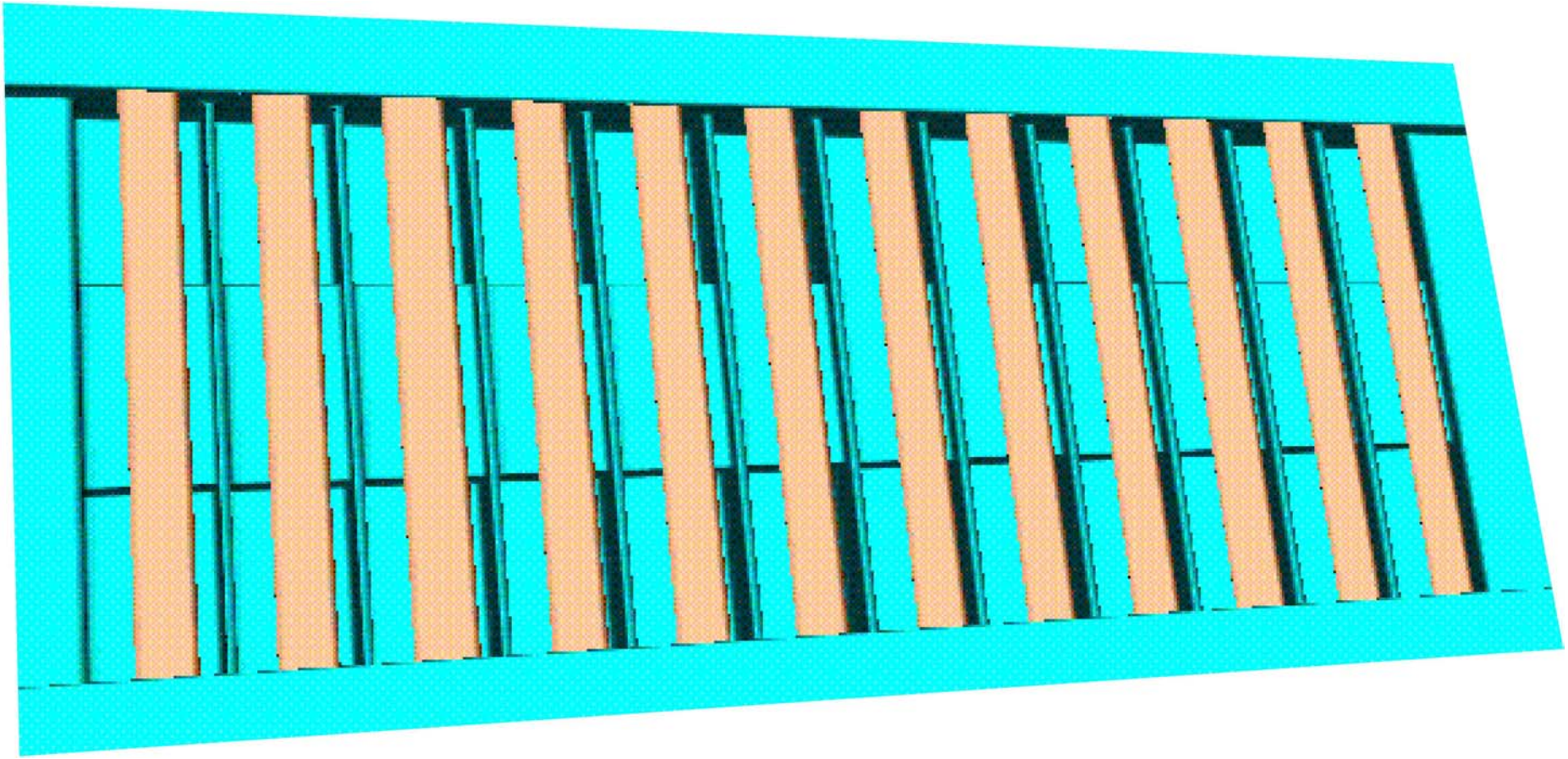
NSTX Antenna

National Spherical Torus Experiment

(High harmonic fast wave antenna) at Princeton Plasma Physics Laboratory



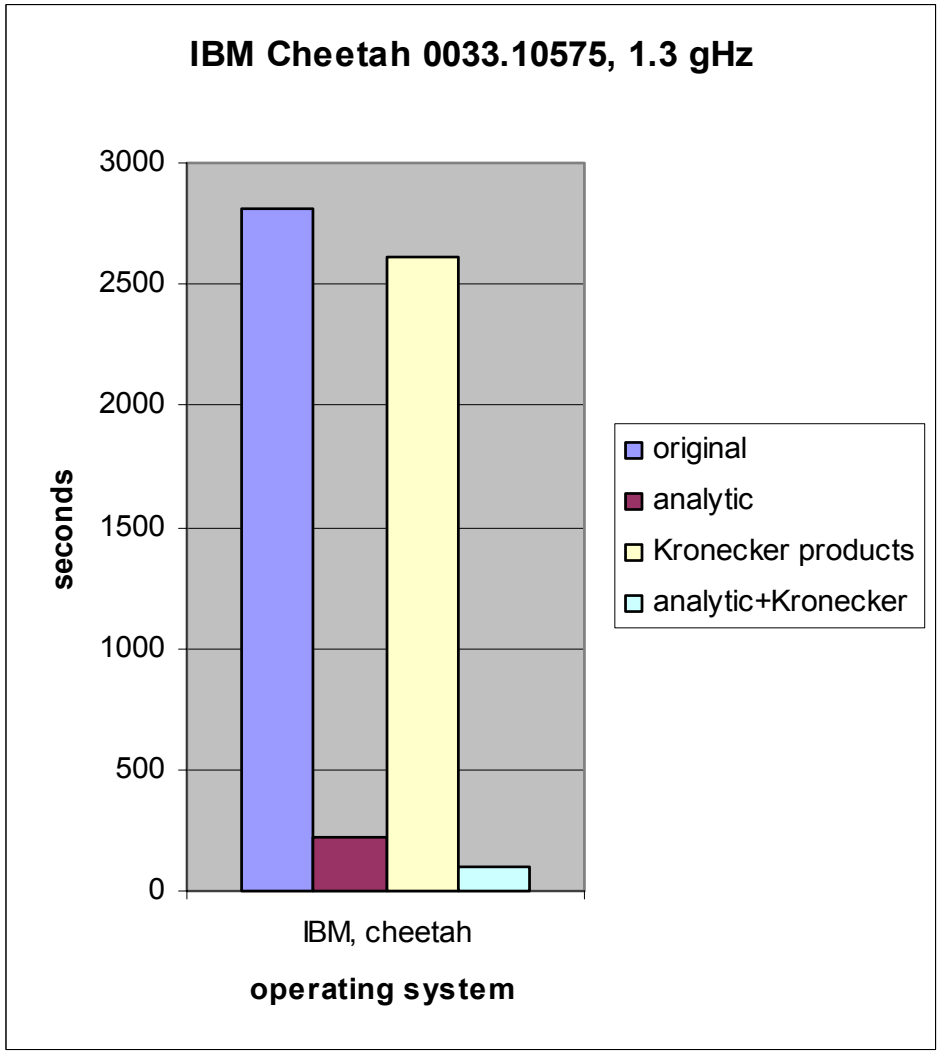
NSTX Model Geometry

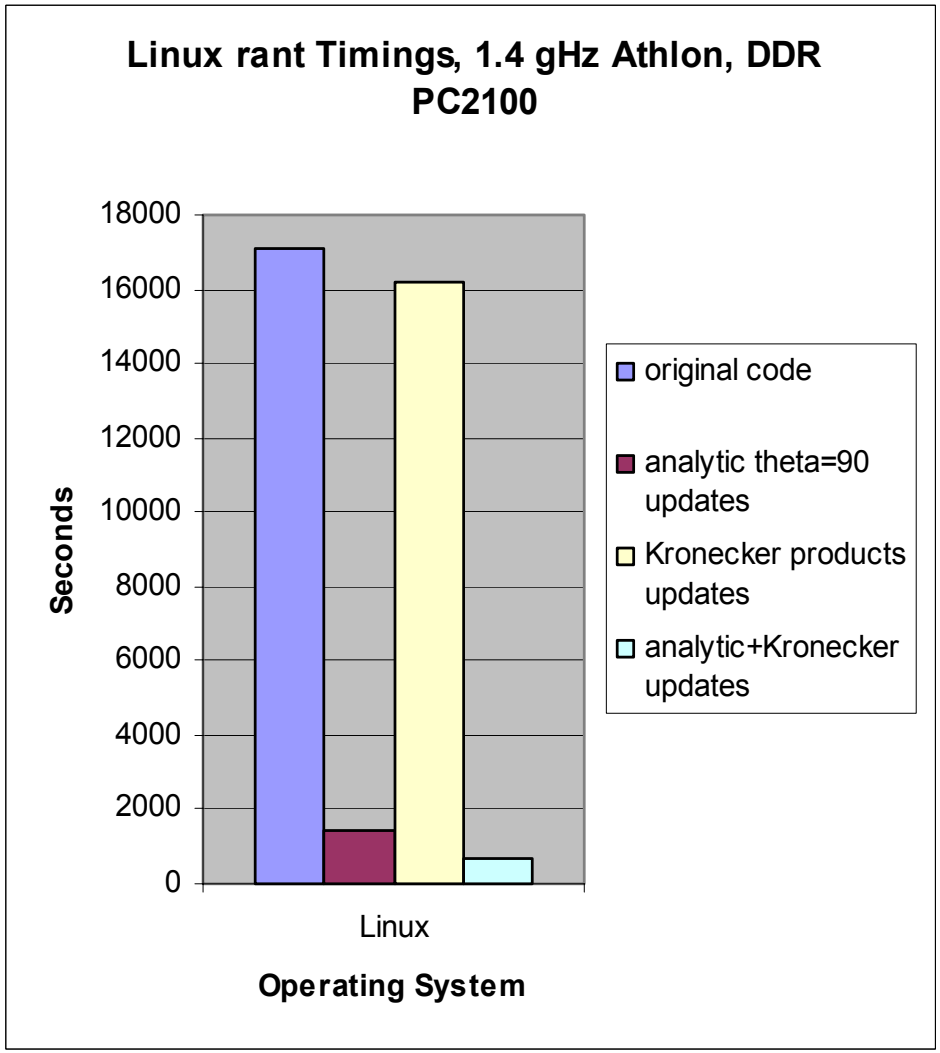


Performance Optimization

- Used analytical solve for theta=90 case to speed up fast Fourier transforms
- Extended the analytical solution for any value of theta using Mathcad and Mathematica
- Examined code for the possible further implementation of Kronecker product algebra. Found that further improvements using Kronecker products would involve major restructuring of the code.
- Took timing data to measure efficiency improvements made using different speed up methods.

Performance Optimization Data





Performance Optimization Conclusions

- The largest improvement in performance came from solving the fast Fourier Transforms analytically.
- Kronecker product implementation also improves the efficiency of the code.
- Using both the Kronecker product improvements and analytical solve greatly improves performances using parallel processors.
- Now a matrix with 14,000 unknowns can be generated and solved in less than 1 hour using the Cheetah.

Original User Interface—UGLY!

Recess_input

```
enter range for this recess, closest x value first
0.006 0.055
Enter height (poloidal) and width (toroidal) of cavity
0.85 3.518
Enter periodic recess coordinates for lower left corner
-0.425 0.2940
Enter boundary condition type for poloidal direction
Enter either waveguide or periodic
Waveguide
Enter boundary condition type for toroidal direction
Enter either waveguide or periodic
Waveguid
Enter number of modes in y’ and z’ dirctions
3 8

Enter range for this recess, closest x value first
0.0810 0.140
Enter height (poloidal) and width (toroidal) of cavity
0.275 0.175
```

…Continued—1 entry for each recess (615 lines total!)

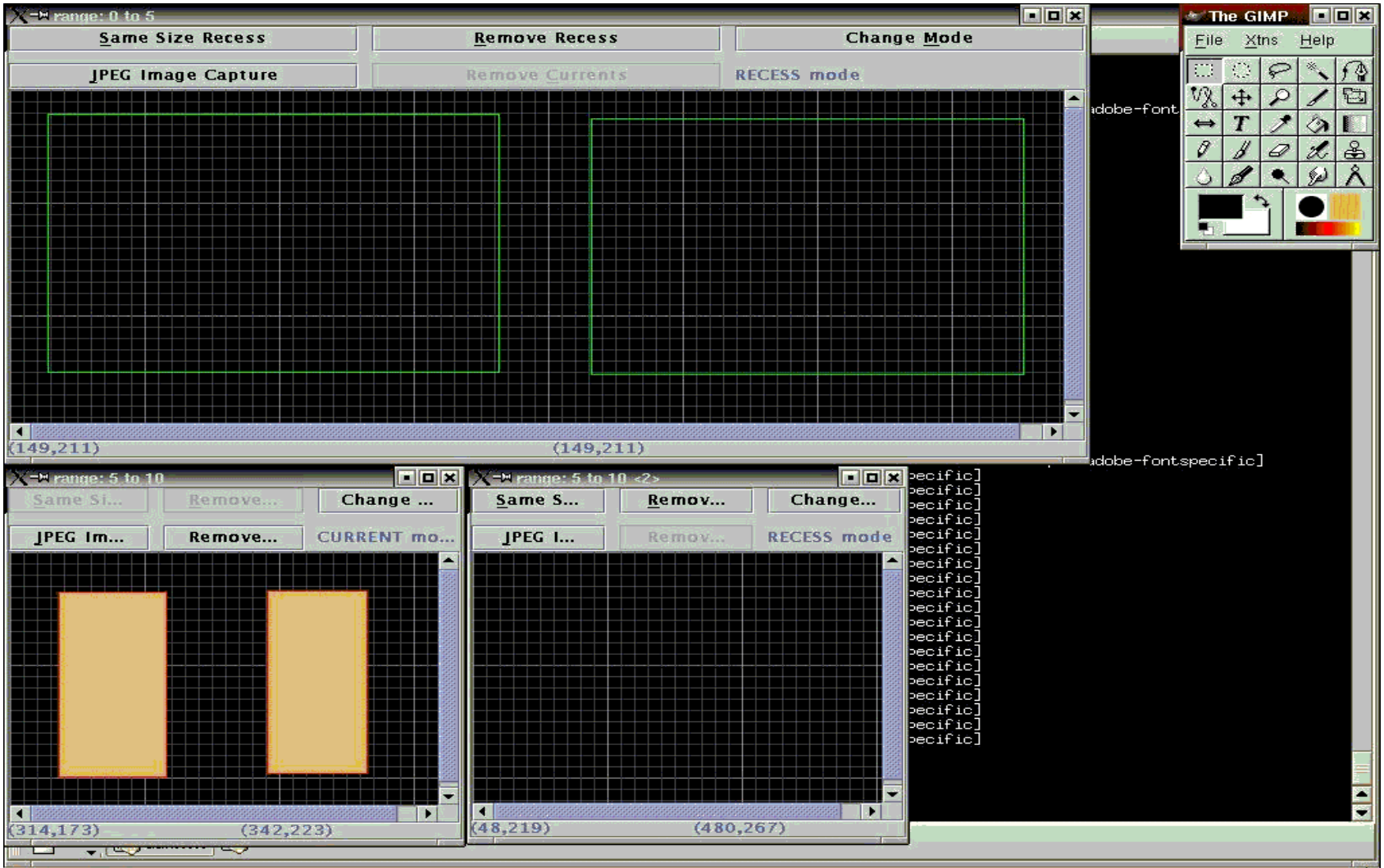
Current_element_input

```
Enter feeder type (o constant, 1 standing, -1 stop input)
And traveling wave parameter (0.0 gives half standing..)
0 0.00
Enter distance from plasma to strap in meters, plasmad
0.0194
Enter amp, tphaz, yc, zc, theta, (theta in degrees)
1.00 0.0 0.0 0.194 90.0
Enter width, height, phazi, phazs
0.0762 0.95 0.0 –0.475

Enter height (poloidal) and width (toroidal) of cavity
And traveling wave parameter (0.0 gives half standing..)
0 0.00
Enter distance from plasma to strap in meters, plasmad
0.0194
Enter amp, tphaz, yc, zc, theta, (theta in degrees)
1.00 0.0 0.0 0.387 90.0
Enter width, height, phazi, phazs
0.0762 0.95 0.0 –0.475
```

…Continued—one entry for each current strap (120 lines total)

Screen Shot of GUI



User Interface Improvements

- The GUI is much easier to use than text based input. Before, a pencil and paper was needed for laying out antennas before putting inputs into the text file.
- Some error checking is built into the GUI, so it is more difficult to enter erroneous data.
- The information obtained from the GUI is written to recess_input and current_element_inputs files in the same format as the old input files, so the GUI is completely backward compatible.

Portability

- rant3d code Portability
  - Ported rant3d from IBM AIX to Linux and now have one copy of code that runs on both operating systems.
- GUI Portability
  - Written in Java, so it should run on all platforms which have a Java runtime environment installed.
  - Tested on Linux, Windows, and Macintosh operating systems.

Conclusions

- Performance of rant3d code was improved significantly
  - Analytic solutions of fast Fourier transforms showed the most improvements in small cases.
  - Exploitation of Kronecker products also sped up calculations and showed notable improvement in larger cases.
- A graphical user interface makes inputs for the code both easier to understand and less prone to errors.
- Porting the code to Linux allows more users access to the code.

Acknowledgements

- ERULF, US Department of Energy
- Mark Carter, FED
- Ed D'Azavedo, CSM
- Adam Griffith, Rice University
- Phil Ryan, FED
- Russ Overbey, FED