

Abstract Submitted  
for the DFD17 Meeting of  
The American Physical Society

**The Julia programming language: the future of scientific computing**<sup>1</sup> JOHN GIBSON, University of New Hampshire — Julia is an innovative new open-source programming language for high-level, high-performance numerical computing. Julia combines the general-purpose breadth and extensibility of Python, the ease-of-use and numeric focus of Matlab, the speed of C and Fortran, and the metaprogramming power of Lisp. Julia uses type inference and just-in-time compilation to compile high-level user code to machine code on the fly. A rich set of numeric types and extensive numerical libraries are built-in. As a result, Julia is competitive with Matlab for interactive graphical exploration and with C and Fortran for high-performance computing. This talk interactively demonstrates Julia's numerical features and benchmarks Julia against C, C++, Fortran, Matlab, and Python on a spectral time-stepping algorithm for a 1d nonlinear partial differential equation. The Julia code is nearly as compact as Matlab and nearly as fast as Fortran.

<sup>1</sup>This material is based upon work supported by the National Science Foundation under Grant No. 1554149

John Gibson  
University of New Hampshire

Date submitted: 04 Aug 2017

Electronic form version 1.4