Update on the Psi4 Project

C. David Sherrill

School of Chemistry and Biochemistry

School of Computational Science and Engineering

Georgia Institute of Technology

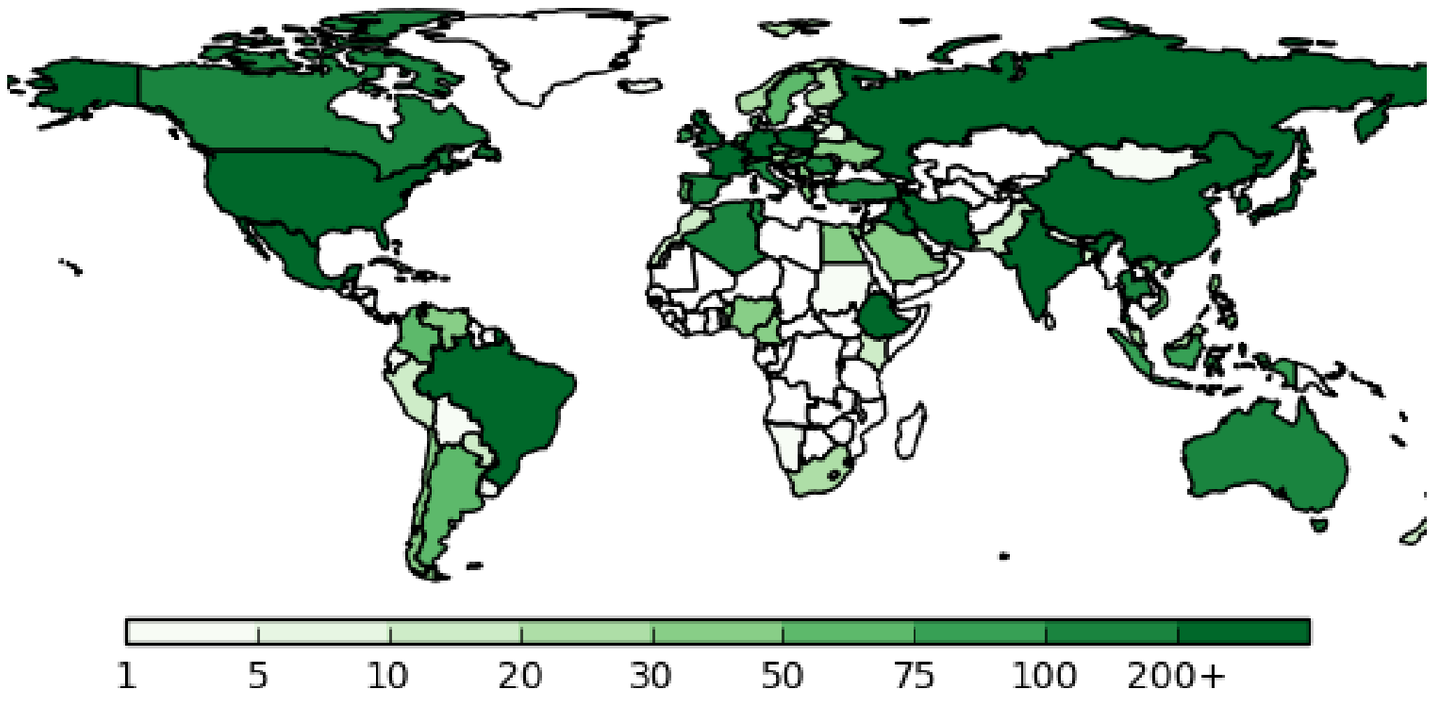
Georgia Institute of Technology
Center for Computational Molecular Science and Technology

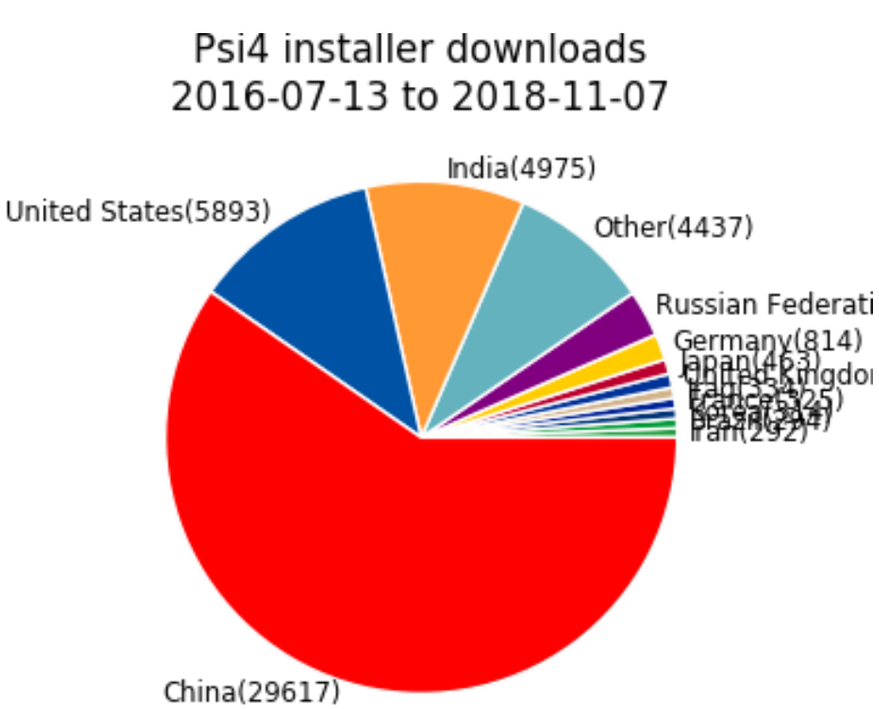
STATUS

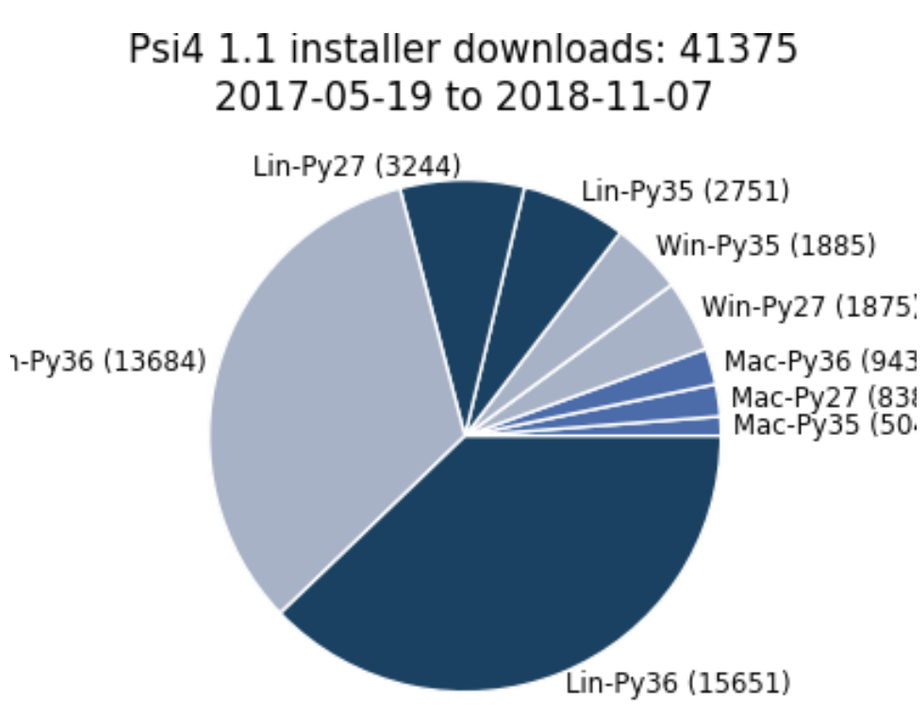
Psi4: The Papers

- J. M. Turney et al., *WIREs: Comput. Mol. Sci*. **2**, 556 (2012)
  - ISI “Highly Cited Paper”: In the top 1% of the field of Chemistry based on a highly cited threshold for the field and publication year
  - 379 citations as of November 2018
- R. M. Parrish et al., *JCTC* **13**, 3185 (2017)
  - 98 citations as of November 2018
- Psi4NumPy: D. G. A. Smith et al., *JCTC* **14**, 3504 (2018)
  - 1 citation as of November 2018

Psi4 Downloads



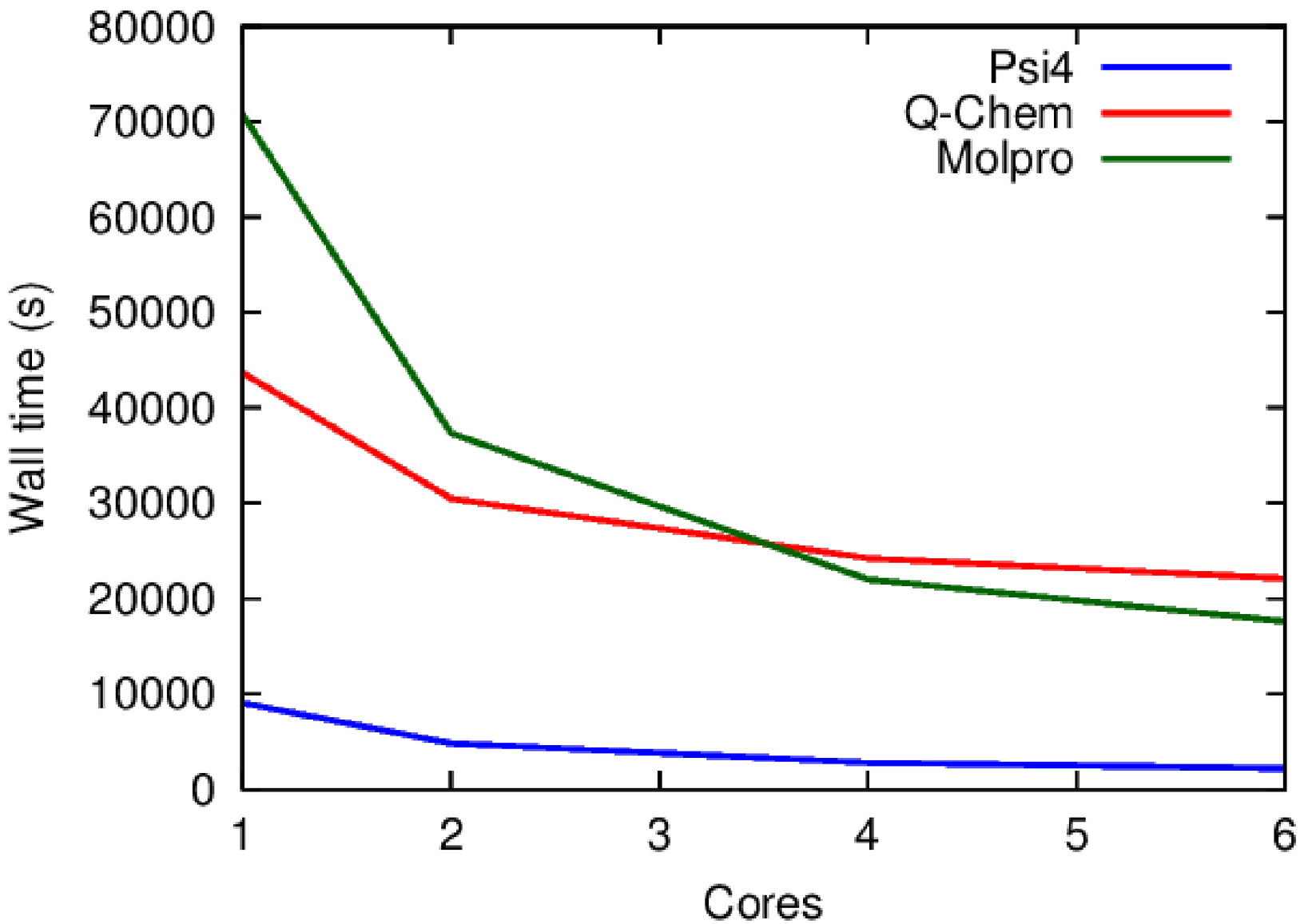




Performance

- For many types of computations, we are competitive with or better than commercial codes

Benzene dimer, DF-FNO-CCSD(T)/aug-cc-pVDZ with 99.5% occupancy (conventional for Molpro)



Anecdotal Feedback

- Lots of favorable comments from users and developers
- Psi4 seems to getting a lot of attention
- Positive comments about ease of installation

STRATEGY

Goals of the Psi4 Project

- Provide free, open-source software for quantum chemistry
- Make the code fast
- Make the code easy to use
- Provide good documentation for users
- Provide a robust infrastructure for rapid development of new theoretical methods
- Make the code friendly to reusable libraries, add-ons, plugins, etc. Promote development of reusable software components
- *Make Psi4 the premier free quantum chemistry program*

Who is in our Target Audience?

In priority order…

1. End users who want to do chemistry with a QC program
2. Educators who want to train students in QC
3. Developers who want to create / implement a method

Help these groups before helping others

Strengths

- Open-source license earns us points with users, developers, collaborators, and funding agencies; we were one of the first serious QC codes in this space
- Generally fast code
- Easy to use
- Compared to most other QC codes, cleaner code and better software development practices
- Good documentation for users

Weaknesses

- Not yet fully featured --- some QC projects can’t be done (or done as easily) with Psi4
  - Analytic DFT Hessians
  - IR intensities
  - ECP’s (in progress)
  - TDDFT
  - GPU capabilities
  - Gradients for solvent models

Opportunities

- Interoperability (with plugins, libraries, other QC codes, MM codes, …)
- Massive-scale automation
- Educational outreach (free, WebMO integration, Jupyter integration, …)

Strategic Aims

- Make Psi full featured (e.g., analytic SCF Hessians, IR intensities, ECP's, TDDFT)
- Make Psi the best code for multi-core architectures
- Add at least reasonable GPU support
- Make Psi truly user-friendly by adding GUI capabilities (e.g., interface to Avogadro 2)
- Leverage existing libraries wherever possible; let others solve some of our problems
- … and what then??