Building ONETEP
Software design, development practices and capabilities

James C. Womack

University of Southampton, UK

ONETEP masterclass 2019
University of Warwick
Preamble

Slides will soon be available online at
http://www.onetep.org/Main/Workshops

See slides online for bibliography
Outline for this talk

How we build ONETEP and how you can use it

1. Thoughts on scientific software development
   ▶ What defines scientific software?
   ▶ What is “good” scientific software?

2. ONETEP as a research tool
   ▶ The motivating problem
   ▶ Linear-scaling density functional theory
   ▶ A tour of capabilities

3. ONETEP as a software package
   ▶ Distribution, usage and governance
   ▶ Design and supporting infrastructure
   ▶ Software development practices
Scientific software
**What defines scientific software?**

Some common attributes of scientific software packages:

1. Motivated by needs of developers
   - Developers and users often overlap

2. Not developers’ main “product”
   - A means to perform research

3. Continually developed, extended
   - Always a prototype, never finished
   - Nature of problem changes

These are some general characteristics I have observed: they are not unique to scientific software (e.g. Linux kernel), nor are they universal to all scientific software.
What defines scientific software?

Contrast this with software to control an ATM . . .

1. Designed for users, not developers
2. Software is developers’ product
3. Clear, fixed specification

Image: “Nice ATM” by Flickr user Jason Cupp, used under CC BY 2.0
Scientific software: some key principles

**Version control**
- Scientific record
- Multiple developers
- Avoid regression

**Modularity**
- Simplified debugging
- Code reuse
- Scientific record

**Testing**
- Confidence in results
- Validate new functionality
- Help future developers

**Documentation**
- Easier to extend
- Support users
Scientific software is scientific infrastructure

A scientific software package is similar to a large shared scientific facility:

1. Designers (of experiments) are users
2. The product is research, not the instrument itself
3. New experiments add new capabilities

Image: Google Maps, Imagery (c)2018 Google, Map data (c)2018 Google
Scientific software is scientific infrastructure

Not just a metaphor. . .

- Research tools for scientific communities
- Significant investment of human-time
- Requires maintenance and long-term planning
- Enables applications beyond those possible for any individual researcher or group

e.g. EPSRC’s “Software as Infrastructure” strategy

Image: Google Maps [with logo overlaid], Imagery (c)2018 Google, Map data (c)2018 Google
Scientific software: the good, the bad and the ugly

A. Developer

*J. Gen. Sci. 2018*

A B C D E F

?!

#?*!

?!

science.py

science2.py

A. Developer

J. Gen. Sci. 2018
Scientific software: the good, the bad and the ugly
Scientific software: the good, the bad and the ugly

- Be experimental, but then refine and clean up
- Plan development to avoid unmaintainable messes
- Well-written code is easier to extend and maintain
ONETEP as a research tool
Quantum chemistry

Time-independent electronic Schrödinger equation

\[ \hat{H}\Psi = E\Psi \]
- Solutions provide information about properties of materials and molecules
- Intractable many-body problem for all but the smallest atomic system

Self-consistent field approach (SCF)

\[ \hat{H}\psi_i = \varepsilon_i\psi_i \]
- Convert $N$-body problem into $N$ more-tractable 1-body problems
- Electrons interact through a mean field which is self-consistently solved for
Density functional theory (DFT)

\[ E[n] = T_s[n] + E_{\text{es}}[n] + E_{\text{xc}}[n] \]

- A popular and widely adopted SCF method
- Balances accuracy and computational expense
- Conventional DFT: \( O(N^3) \) scaling

The \( O(N^3) \) scaling of conventional DFT rapidly becomes problematic for > 1000 atoms...

Image: Fig. 1 from D.J. Cole et al., J. Phys. Chem. Lett. 4, 4206 (2013) [CC-BY-4.0 license] [1]
Linear-scaling DFT in ONETEP

Motivation: Practical quantum chemical methods for modelling large systems

Premise: Reformulate DFT to achieve $O(N)$ scaling

▶ Possible due to “nearsightedness” [2] of electronic interactions
▶ Locality of interactions manifested in the density matrix, $\rho(r, r')$
▶ ONETEP expresses DFT as a direct energy minimization wrt $\rho(r, r')$

$$E = \min_{\rho} E[\rho(r, r')]$$

Many interesting theoretical and computational techniques are used to implement this in practice [3, 4].

We will focus on the result. . .
Linear-scaling DFT in action (2005)

Linear-scaling DFT in action (2016)

Iridis 4
512 cores
13696 atoms
A tool for researchers

ONETEP computes and analyses quantities relevant to physical scientists

- Many distinct capabilities
- All built upon LS-DFT framework
- See website for documentation

And now, a brief tour...
Linear-Response TDDFT

Capabilities:

▶ Calculate localised excitations and optical spectra
▶ Analysis of excitations via Quantified Natural Transition Orbitals
▶ Absorption spectra used to predict colours and exciton dynamics

People: Tim Zuehlsdorff, Jian-Hao Li, David Turban, Matt Turner, Peter Haynes, Nicholas Hine

References: 5, 6 (LR-TDDFT); 7 (QNTO); 8, 9 (colour prediction); 1, 10 (exciton dynamics)
Electron Energy Loss Spectroscopy

**Capability:** Predict spectra from EELS measurements

**People:** Edward Tait, Laura Ratcliff, Nicholas Hine

**References:** 11

\[
e_{\ell}(\omega) = \frac{1}{\Omega} \sum_{\ell} \sum_{i} |q \cdot \langle \psi_{i} | r \rangle |^{2} \delta(E_{i} - E_{0} - \omega)
\]

\[
\langle \psi_{i} | r | \psi_{i} \rangle = \langle \tilde{\psi}_{i} | r | \psi_{i} \rangle + \sum_{\ell} \langle \tilde{\psi}_{\ell} | r | \psi_{i} \rangle \langle \langle \tilde{\psi}_{\ell} | r | \psi_{i} \rangle - \langle \tilde{\psi}_{\ell} | r | \psi_{i} \rangle \rangle
\]
Implicit solvent

**Capability:** Simulate molecules, nanostructures etc in a polarizable dielectric medium, calculate $\Delta G_{\text{solv}}$

**People:** Jacek Dziedzic, James Womack, Arash Mostofi, Chris Skylaris

**References:** 12, 13, 14, 15

**Coming soon:** Solvation in solvent + ions (Poisson-Boltzmann model)
Metallic Systems

**Capability:** Treatment of large metallic systems, e.g. metallic nanoparticles, using ensemble DFT (EDFT)

**People:** Jolyon Aarons, Alvaro Ruiz-Serrano, Chris Skylaris

**References:** 16, 17, 18; 19 (AQuA-FOE)

**New linear-scaling method:**
“Annealing and QUenching Algorithm FOE”
Advanced exchange-correlation models

**Capability:** Functionals with orbital dependence via exact exchange (e.g. B3LYP) or $\tau$ (e.g. B97M-rV)

**People:** Jacek Dziedzic, James Womack, Chris Skylaris

**References:** 25 (hybrids) 26 (meta-GGAs)

**Coming soon:** Greatly reduced computational cost!

**Capability:** Fully non-local van der Waals density functionals, e.g. VDW-DF1/-DF2, VV10

**People:** Lampros Andrinopoulos, Gabriel Constantinescu, Fabiano Corsetti, Nicholas Hine, Arash Mostofi

**References:** 20; used in 21

\[
E_{c}^{nl} = \frac{1}{2} \int \int drdr' \rho(r)\phi(r, r')\rho(r') \quad \Rightarrow \quad E_{c}^{nl} = \frac{1}{2} \int \int drdr' \rho(r)\phi(q, q', r)\rho(r')
\]
External Pressure

**Capability:** Simulated systems under external pressure, via enthalpy method

**People:** Niccolo Corsini, Nicholas Hine, Peter Haynes

**References:** 22, 23, 23
Projector Augmented Wave

Capability: Use PAW treatment of core electrons rather than norm-conserving pseudos

Effectively an all-electron treatment. Enables accurate calculations on high-Z elements & transition metals.

People: Nicholas Hine

References: 24; used in 21
Molecular Dynamics

**Capability:** Perform dynamics on large systems, with density matrix extrapolation

**People:** Valerio Vitale, Simon Dubois, Chris Skylaris

**References:** 27, 28

\[
1: \quad v' = v_n + \frac{\Delta t}{2m} \times F_n \\
2: \quad r_{n+1} = r_n + \Delta t \times v' \\
3: \quad \text{Compute ionic forces } F_{n+1} \\
4: \quad v_{n+1} = v' + \frac{\Delta t}{2m} \times F_{n+1}
\]
Capability: Compute electronic transport for multi-lead devices on a large scale

People: Simon Dubois, Rob Bell, Arash Mostofi

References: 29
Population Analysis & Energy Decomposition

**Capability:** Natural Population Analysis, Hirschfeld, DDEC, etc, for forcefield fitting and electrostatics

**People:** Louis Lee, Danny Cole, Max Phipps, Chris Skylaris

**References:** 30, 31, 32, 33, 34
Strong Correlations

Capability:
DFT+U (penalise non-integer occupations); Dynamical Mean Field Theory

People:
David O’Regan, Cedric Weber, Nicholas Hine, Arash Mostofi

References:
35, 36, 37, 38, 39, 40
Constrained DFT

**Capability:** Constrain populations & spins of certain regions of a system (e.g. for studying charge transfer reactions)

**People:** Gilberto Teobaldi, David O’Regan, David Turban, Nicholas Hine

**References:** 41, 42
Spectral Function Projection

**Capability:** Perform supercell calculation, project result onto primitive bandstructure

**People:** Gabriel Constantinescu, Nicholas Hine

**References:** 21, 43, 44
Phonons

**Capability:** Calculate vibrational frequencies, free energies, phonon modes etc, via a) finite-difference approach, b) linear-response methods

**People:**
Fabiano Corsetti, Gabriel Constantinescu

**References:** 45
**New in v5.2: Electron localisation descriptors**

**Capability:**
Visualisation of electron pair localisation, providing a quantum VSEPR-like representation for prediction of interactions

**People:**
Rebecca Clements, James Womack, Chris Skylaris

**References:**
Article in preparation
New in v5.2: Quantum embedding

Capability:
Embedded mean-field theory (EMFT) treats parts of a system with different levels of theory (e.g. hybrid + GGA), reducing cost whilst maintaining accuracy

People:
Joe Prentice, Rob Charlton, Peter Haynes, Arash Mostofi

References:
Article in preparation
ONETEP as a software package
# Distribution and licensing

<table>
<thead>
<tr>
<th>License</th>
<th>Cost</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaborator - Individual, specific project</td>
<td>-</td>
<td>Individual, specific project</td>
</tr>
<tr>
<td></td>
<td>Includes source</td>
<td>Includes source</td>
</tr>
<tr>
<td>Academic</td>
<td>£500</td>
<td>Entire research group</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Includes source</td>
</tr>
<tr>
<td>Commercial</td>
<td>Purchase of Materials Studio</td>
<td>Non-academic users</td>
</tr>
<tr>
<td></td>
<td>GUI interface</td>
<td></td>
</tr>
</tbody>
</table>

- All members of the [CCP9](#) community are entitled to collaborator agreements.

- Daussault Systèmes [BIOVIA Materials Studio](#) packages ONETEP into a suite of materials science tools with a single GUI.
Usage

150 academic licenses
(individuals and groups)

20-30 academic collaborations
(PIs and groups)

Daussault Systèmes BIOVIA’s products have high market penetration in top chemical companies (pharma, oil, automotive, materials, etc.)

Data from ~2016–17
## ONETEP Developer’s Group (ODG)

Group of core developers who manage licensing, releases, relationship with BIOVIA:

- Jacek Dziedzic\(^1,\)\(^2\) Peter D. Haynes\(^3\) Nicholas D. M. Hine\(^4\)
- Arash A. Mostofi\(^3\) Mike C. Payne\(^5\) and Chris-Kriton Skylaris\(^1\)

1 University of Southampton, 2 Gdańsk University of Technology,
3 Imperial College London, 4 University of Warwick, 5 University of Cambridge

<table>
<thead>
<tr>
<th>Code Contributors</th>
<th>Active Users of Repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>37</td>
</tr>
</tbody>
</table>

(code contributors (past and present))
(active users of repository (developers and expert users))
Source code

Linear-Scaling Ab Initio Total Energy Program
Release for academic collaborators of ODG
Version 5.1.1.1.

- Written in Fortran (2003 standard), built using GNU Make
- Bundled support utilities written in other languages (e.g. Bash scripts)
- Fortran modules used to encapsulate related procedures and data types
Parallelism

- Designed to run on parallel (super)computers
- Necessary for large-scale calculations
- Combined MPI [46] and OpenMP [47] parallelism.

**MPI** Work divided between processes which communicate by *message passing*

**OpenMP** Processes subdivided into threads, with *shared memory* access
Interaction with other software

Designed for Unix-like systems: Linux, Mac OS X (Windows via WSL)

Supporting software is required to build and run ONETEP:

<table>
<thead>
<tr>
<th>Software/library</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fortran compiler (F2003, OpenMP)</td>
<td>Intel Fortran, GNU Fortran</td>
</tr>
<tr>
<td>MPI library</td>
<td>Intel MPI, OpenMPI</td>
</tr>
<tr>
<td>Linear algebra libraries</td>
<td>BLAS, LAPACK</td>
</tr>
<tr>
<td></td>
<td>ScaLAPACK, Intel MKL</td>
</tr>
<tr>
<td>Fast Fourier transform (FFT) library</td>
<td>FFTW, FFTW3</td>
</tr>
<tr>
<td></td>
<td>Vendor-specific FFT libraries</td>
</tr>
<tr>
<td>Multigrid solver library</td>
<td>DL_MG</td>
</tr>
</tbody>
</table>
DL_MG

DL_MG [15] is an open-source Poisson solver library distributed with ONETEP

- Solves several variants of the Poisson equation in real-space:

\[ \nabla \cdot (\varepsilon(r) \nabla \phi(r)) = -4\pi (n(r) + n_{\text{ions}}[\phi](r)) \]

- Designed for use in large scale electronic structure calculations
- Multigrid approach [48] scales well with problem size and converges rapidly
- Iterative high-order defect-correction reduces discretization error

Enables open BC electrostatics and implicit solvent functionality in ONETEP

Source code available at www.dlmg.org
Hardware

- ONETEP runs well on conventional CPU-based machines
- Scales to 1000s of cores [47]
- More cores and more memory per core are generally advantageous
- GPU-port is a work-in-progress [49]
- Can also run on desktops/laptops
How do we manage day-to-day development?

Follow the key principles:

1. Version control
2. Testing
3. Documentation
4. Modularity
Version control

We have a Git repository hosted on Bitbucket and use a forking workflow:

- Single “official” server-side repository
- Private fork for each contributor
- Changes contributed via pull request
- Pull requests undergo review

New developers don’t need to worry about breaking the official repository!
We have a growing suite of regression tests:

- When new functionality is added, require a new regression test (“QC” test)
- Each test runs a calculation to produce known results (benchmarks)
- QC test suite is run by a Python program called testcode
- A test fails if outputs are outside of allowed tolerance (wrt benchmarks)

The test suite allows new builds of ONETEP to be validated.
Testing

We use the Buildbot package to automate testing of the software:

- A **Buildbot** instance regularly pulls the official master branch and starts a build
- Each build includes compilation, code quality checks and QC test suite
- If any step fails, core developers are notified by e-mail
Documentation

Developer documentation:

- Template module file to encourage consistent documentation
- Each new module/procedure should be documented
- Encourage developers to comment their code as they write it

User documentation:

- User documentation is required for substantial new functionality
- Distributed with the code and at www.onetep.org
- Daussault Systèmes BIOVIA provide documentation for Materials Studio
ONETEP’s source code is organized as a loose hierarchy of modules:

- Modules encapsulate related functionality
- High-level modules depend on low-level modules
- Low-level modules provide basic functionality
- High-level modules perform sophisticated tasks

A module can often be developed without needing to consider the behaviour of the entire program.
Closing remarks
Summary

Building ONETEP

1. ONETEP is a tool for large-scale quantum simulation of matter
2. An extensive and expanding list of capabilities
3. Development of ONETEP follows a few key principles
4. These principles keep ONETEP usable and maintainable
Introducing ONETEP: Linear-scaling density functional simulations on parallel computers
Acknowledgements

Funding and compute resources

- EPSRC-funded CCP9 flagship (EP/P02209X/1)
- UK MMM Hub “Thomas” (EP/P020194/1)
- IRIDIS HPC Facility (Southampton)
- ARCHER UK National Supercomputing Service (UKCP consortium EP/P022561/1)

Thank you for your attention!

...any questions?
Bibliography I


Bibliography III


