ESS/ILL User Meeting

BigDFT

New on-line DFT approach to analyse neutron diffraction crystallography and SANS data

Viviana Cristiglio, Luigi Genovese

Institut Laue Langevin, Laboratoire de Simulation Atomistique - L_Sim

January 21, 2021

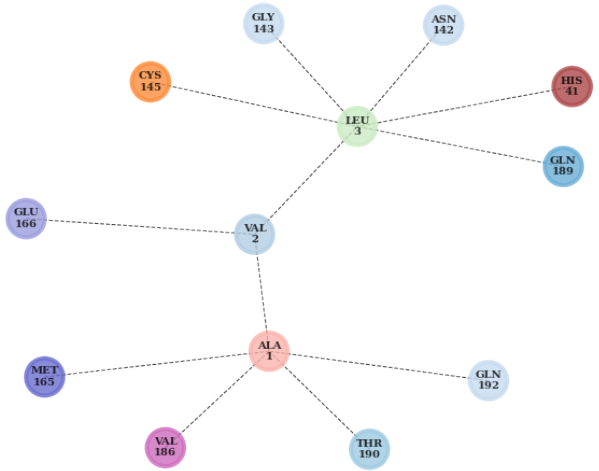
Grenoble

Complexity of biological systems

Various conditions

- Large systems of many thousand atoms
- Atomistic models should model atomic interactions, yet electronic information may be important
- Difficult to obtain predictivity without arbitrariness

✘ A number of important implications



What we will illustrate today

Quantum Mechanics (DFT) may be needed

- Whenever DFT is necessary to study the electronic structure of the systems, it is important to provide the tools to *interpret* experimental data
- Need of new tools developed especially for the study of biological systems

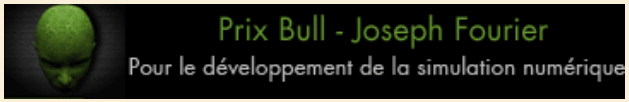
☞ The BigDFT code provide a new paradigm of analysis

Main ingredients

- PDB files from neutron crystallography, Cryo-TEM, MD simulations, ...
- Remotely accessible (super) computing platform
- A post-processing infrastructure easy-to-use

BigDFT: DFT with wavelets

A DFT code conceived for HPC (`www.bigdft.org`)

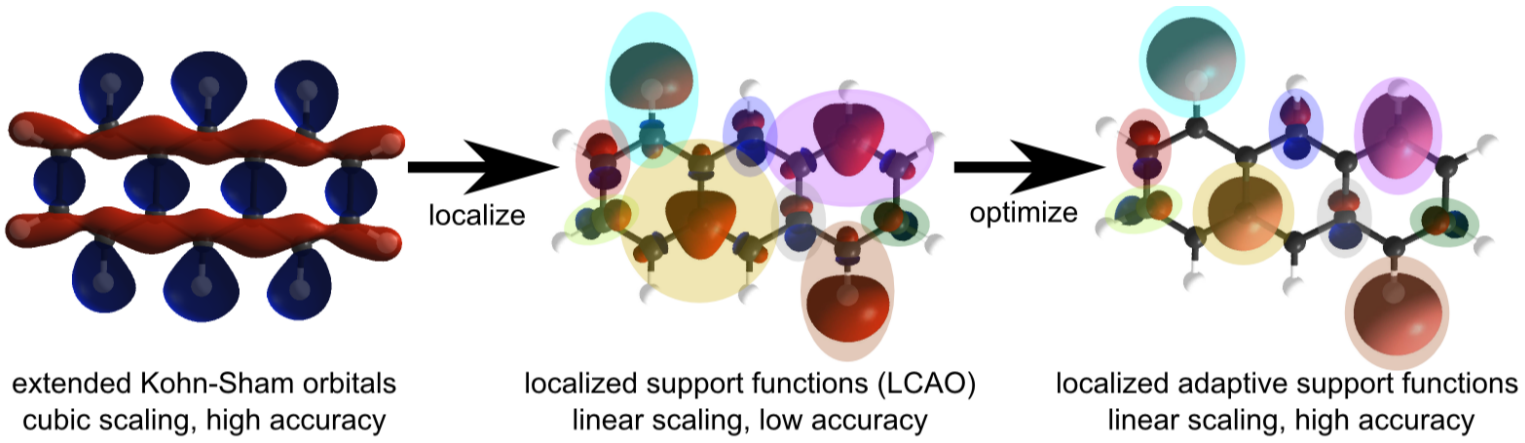
- DFT calculations up to many thousands atoms
- 👉 An award-winning HPC code 

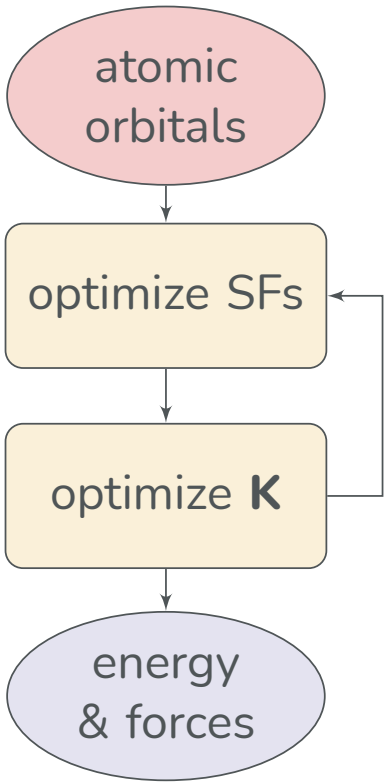
- BigDFT has been conceived for massively parallel etherogeneous architectures since more than 10 years (MPI + OpenMP + GPU)

Code able to run routinely on different architectures

- GPU accelerators since the advent of double-precision GPGPU (2009)
- Various large calculation projects since 10 years
- ✔ A code conceived for supercomputers

A Linear Scaling Algorithm





Accurate Minimal Basis

- Employs optimized DoF to express accurately and efficiently the DFT solution
- Able to treat systems up to tens of thousand atoms
- Ideal framework to reduce the complexity of the description

How DFT should be used at this scale?

In practice:

- O(N) codes have been able to compute large systems for many years now.
- And yet, how often do you encounter research being done with DFT involving systems of many tens of thousands of atoms?

Issues related to DFT in large systems

- Enthalpy challenge: are DFT functionals really more accurate than a well tuned forcefield?
- Entropy challenge: can DFT capture the full set of conformations of a large system?

The key added-value of DFT is not accuracy, but insight

Complexity Reduction: QM Fragments

- Consider a system of interest, which we have arbitrarily partitioned in to two sets of atoms. We can call these sets QM Fragments.
- We introduce the concept of *Complexity Reduction* - dividing large, complex systems into chemically meaningful fragments and measuring their interaction.

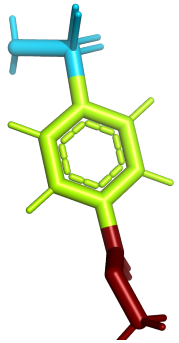
Robust and well-defined indicators

- Purity Indicator - similar to the classical concept of Atomic Valence.
- Fragment Bond Order - similar to standard atomic bond order.

Automatic Fragmentation of Systems

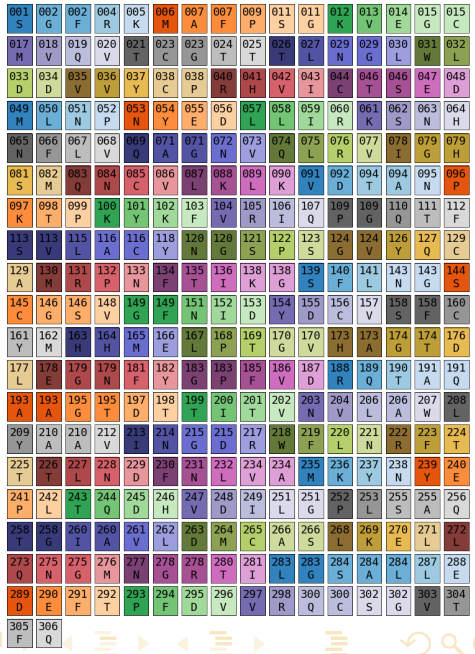
Small Molecules - From Atoms Up

- Automatic: We can re-organize a system into fragments without prior knowledge.
- Robust: Non-expert DFT users can interpret the information coming out of DFT calculations.



Large Biomolecules

- Proteins are often already divided into fragments based on their Amino Acids.
- Yet not all amino acids are equally good fragments. We can combine them together to build a more coherent picture.



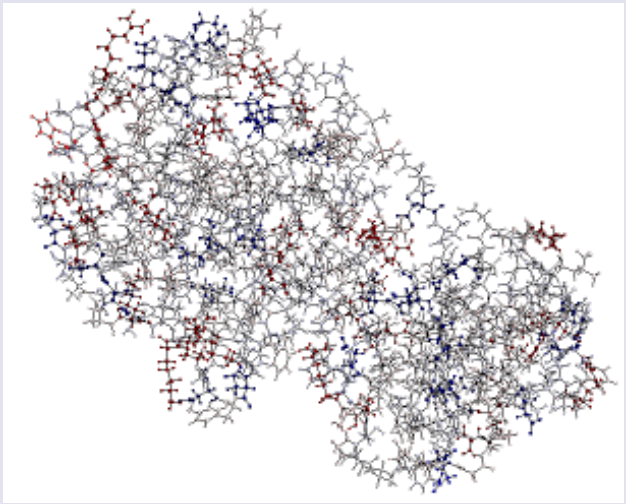
How this is related to Neutrons?

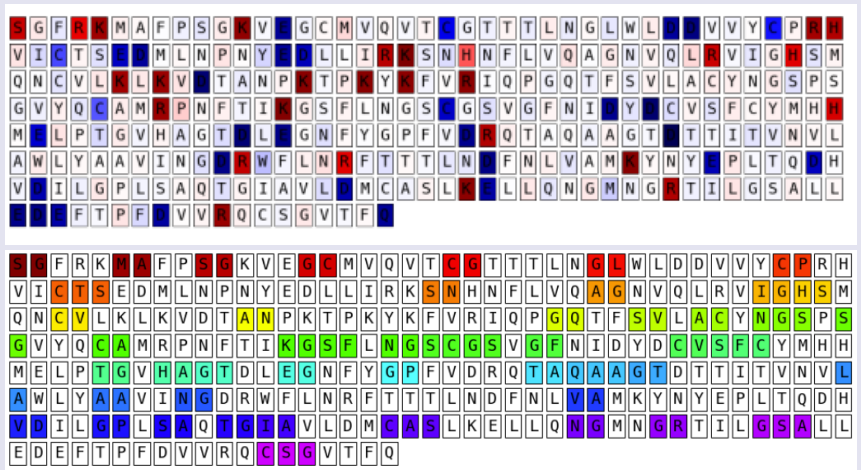
From Neutron crystallography to DFT

- Direct process of PDB
- ✎ Neutron Crystallography files contain also the positions of the deuterium → protonation state is given

Amino acid charges of SARS-CoV-2 MPro

(PDB 7JUN - Oak Ridge)





Workflow ready to be extended

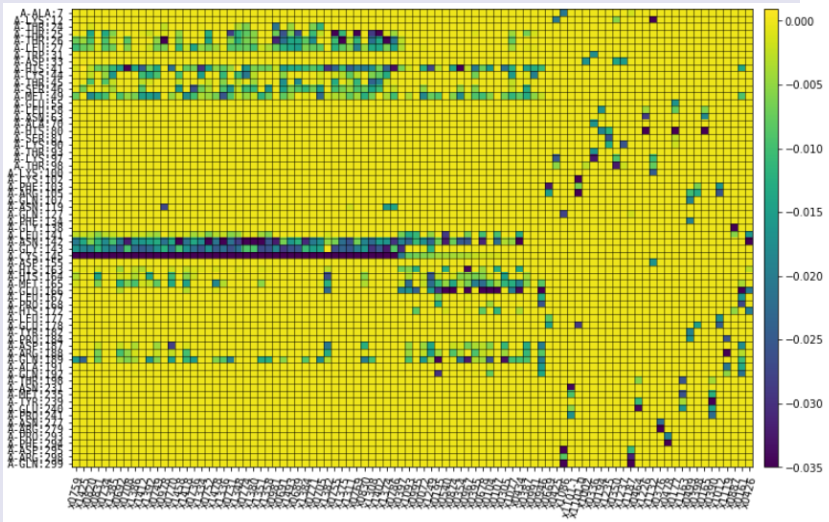
Other QM observables can be analyzed

- Interaction maps can be represented (eg. role of the crystallographic solvent)
- High throughput analysis can be performed on NC databases

COVID XChem fragments project

MPro crystal structures of 92 different small inhibitors

☛ Map the interactions of inhibitors with enzyme



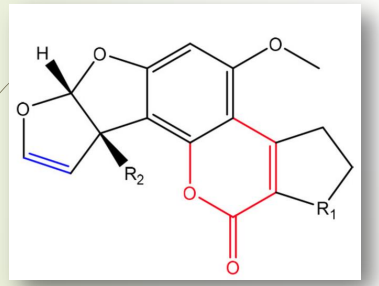
Enzymatic Activity of Laccase

Case Study

Laccase is a potential bioremediator for Aflatoxin contamination

AFLATOXIN B1 (AFB1)

- PRODUCERS: Genus *Aspergillus*
- STRUCTURAL TRAITS: Difuranocoumarin derivatives produced by a polyketide pathway.



R_1 = C=O, -(C=O)-O- or C-OH

R_2 = H or OH

Aflatoxicosis:
Chronic: Liver cancer, Immunosuppression, Death.
Acute: Death.

- TOXICITY:
  The most carcinogenic metabolite in nature. The lactone ring is the main responsible for toxicity, and secondarily the terminal double bond of the difuran moiety.
  Cytochrome P450 activates AFB1 into a highly reactive epoxide that **easily binds to DNA and proteins**, inducing mutagenicity and cytotoxicity.

Combination of DFT and Structural caracterization

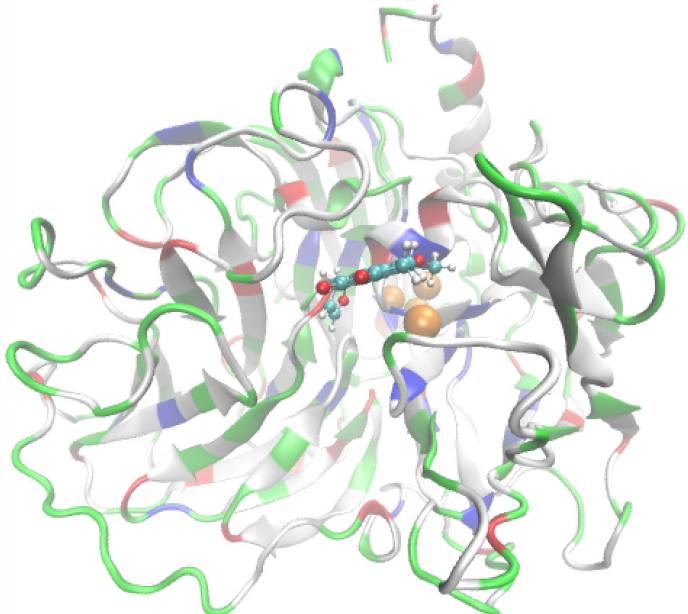
Structural analysis informed by SANS

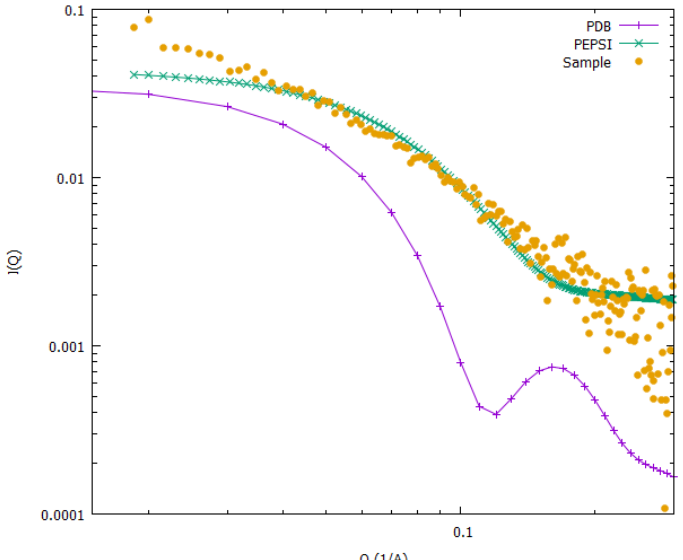
Aflatoxin Degraded by Laccase Enzyme

Open Questions and present projects

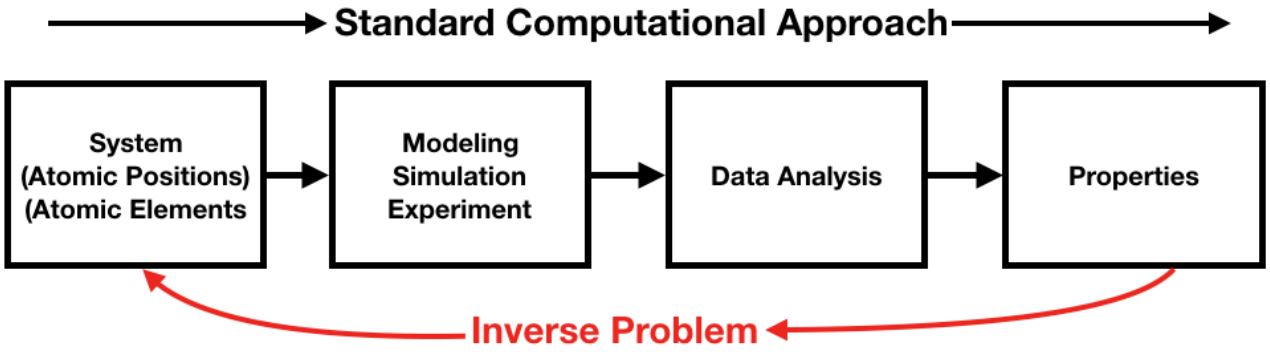
- How can we break down this enzyme into smaller parts, and which of those parts are relevant for the reaction?
- What interplay between toxin and environment?
- Why are some toxins easier to degrade than others?

☛ SANS data can be used to validate DFT structures





Going from Understanding to Design



The Inverse Problem of Computational Science

- Mechanistic models might be used to gain detailed insight into individual systems.
- However, the goal of bioremediation is not understanding, but design.
- Inverse problem: given a set of desirable properties, how to we build a system that produces those properties?

Towards a new scientific paradigm

- Systems of interest in biology may *start* to be modelled
- Structural information coming eg. from Small-Angle Scattering might be put in relation to such investigations

Implications

Development of a method capable of informing biological problems without arbitrariness (e.g. antibiotic resistance, antibodies and drug development etc.)

- First steps towards a rational, iterative amelioration of laccase focused on substrate specificity.

Quantum-as-a-Service approach

A dedicated VISA platform is under implementation
Collaboration L_Sim (CEA Grenoble) and CS Group (ILL)

PyBigDFT

Pre- and Post-processing of simulations are performed via a `Python` module

HPC

Calculations triggered remotely on a super-computer from a Jupyter notebook (AiiDA framework)

Userclub

Simulation can be processed from a platform next to experimental data (ILL User Club access)

Database

Large databases of biological systems can be created

New insights for (neutron) data analysis

- Dedicated routine for neutron crystallography data interpretation
- Possible improvement from SANS data

In a nutshell

Summary

- DFT shouldn't be employed for large systems just on the hope of accuracy, but instead with the goal of insight.
- Complexity Reduction - We have developed a way to use information from DFT to generate coarse-grained views of a system by defining reliable fragments and measuring their interaction.

Packaging

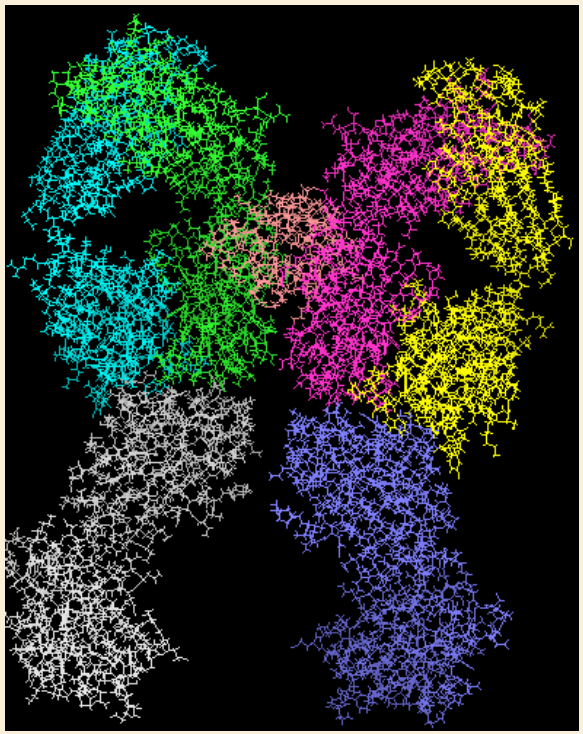
- The complexity reduction framework presented here is available through our PyBigDFT package.
- Postprocessing can be done even for large systems on a typical workstation.
- This combination create interesting opportunities for interdisciplinar collaborations

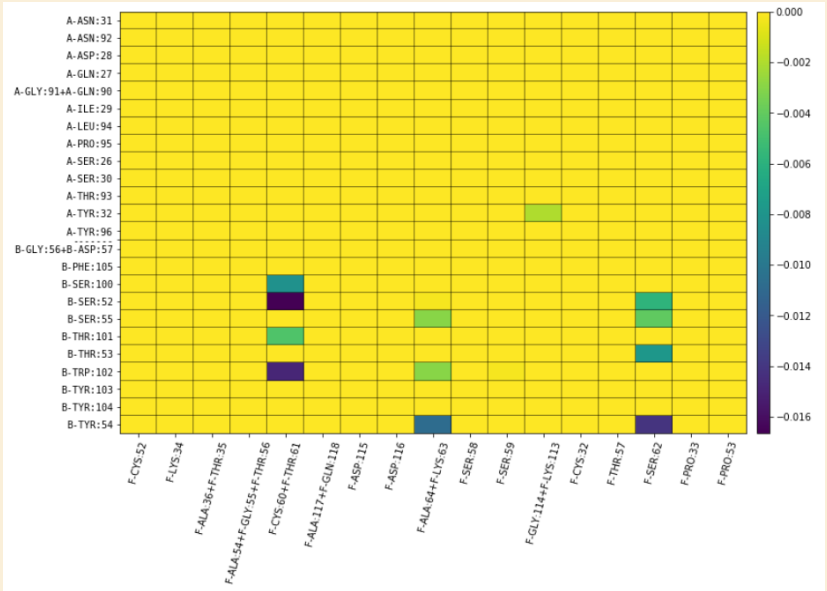
Lots of Systems of interest in Biology

Large systems are routinely accessible

Linear Scaling code → CPU time *per atom*
Example: 1400 Residues (One Monoclonal Antibody). 3h of walltime on 32 nodes of IRENE-Rome Machine

Reduce the interactors in a biological system





Acknowledgments

London L. E. Ratcliff, M. Stella

Kobe W. Dawson, T. Nakajima

Basel S. De, S. Goedecker

Barcelona S. Mohr

Grenoble A. Degomme, D. Caliste, T. Deutsch, LG

Boston, ILL M. Zaccaria, M. Reverberi, **V. Cristiglio**

Rome M. D'Alessandro

Catania G. Fisicaro

Advantage of aiida technology

We have implemented the "traditional" flavour of AiiDA plugin. `pip install aiida-bigdft`

▸ BigDFT plugin

Integrated in PyBigDFT

▸ PyBigDFT AiiDA runs

A technology that makes the notebook a *console* to launch the job and to analyze production data

`AiidaCalculator` ☛ used to remotely submit the job

```
from BigDFT import AiidaCalculator as A
study=A.AiidaCalculator(code="bigdft@localhost",
         num_machines=1,mpiprocs_per_machine=1,
         omp=1,walltime=3600)
%load_ext jupyternotify
%notify
hgrid_cv.wait()
>>> '0 processes still running'
```

Employed for (Virtual?) Benchmarks, Production.

Future scientific directions

Linear-Scaling DFT calculations based on wavelets

- Robust convergence, high accuracy and flexibility
- Reduction in degrees of freedom $\rightarrow$ large systems
- Different level of descriptions (controlling the precision) QM $\supset$ Fragments $\supset$ Atomic charges
- Opens up new possibilities

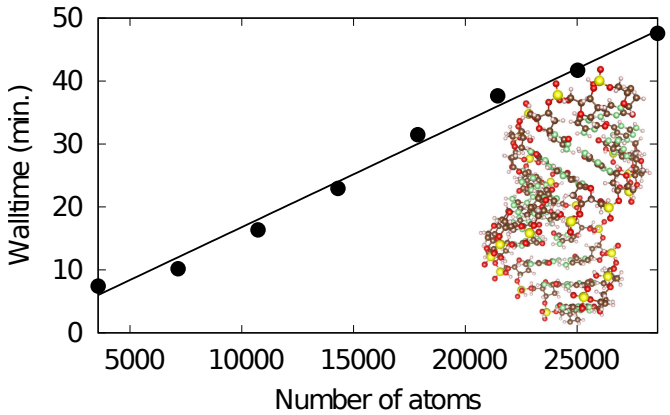
Challenges and future directions

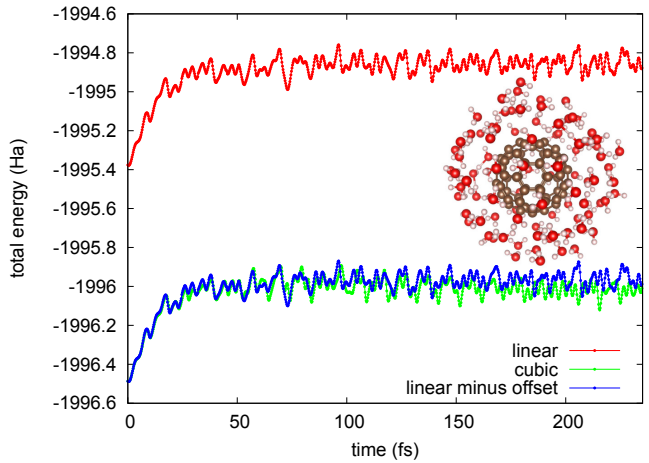
- Explore interplay environment $\leftrightarrow$ electronic excitations (CDFT, QM/MM, statistics...)
- Provide high quality back end for different communities (Biology, Electro-Chemistry, ...)

Accurate DFT for 1000s of atoms

Linear Scaling DFT with Wavelets

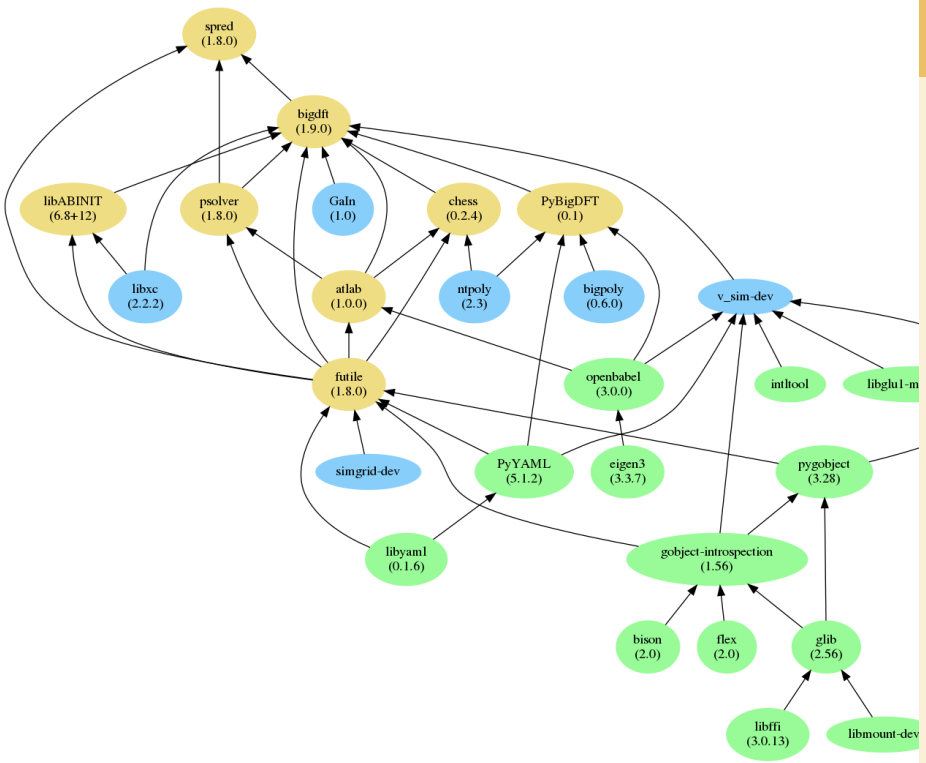
- minimal, localized basis with wavelet accuracy – can treat $> 10,000$ atoms
- applicable to a range of materials, including metals
- accurate forces – geometry optimizations, molecular dynamics





LS-BigDFT References: { Mohr, Ratcliff Boulanger, LG *et al.*, J. Chem. Phys. **140**, 204110 (2014)
Mohr, Ratcliff, LG *et al.*, Phys. Chem. Chem. Phys. **17**, 31360 (2015)
LS-BigDFT Metals: Mohr, Eixarch, Amsler *et al.*, J. Nucl. Mater. Energy **15**, 64 (2018)
CheSS: Mohr, Dawson, Wagner *et al.*, J. Chem. Theory Comput. **13**, 4684 (2017)

Code release and distribution



Modularity first

BigDFT-suite is a collection of different independent libraries with own build system. Enables to include third-party libraries (green) and upstream modules (blue)
☛ Dependencies can be expressed easily in the jhbuild-based bundler.

Containered `hub.docker.com/r/bigdft`

BigDFT suite is relased as a `docker` container (GPU-direct ready) ☛(also on NVidia NGC repository)
Different Flavour of containers: Runtime, SDK, Console