Session 1: towards better benchmark sets (was: “status of the Delta project”)

From the discussion, a clear wish emerges for more/larger test sets, that are well-document via all-electron DFT codes. The requirements for those test sets vary:

- There is a need for test sets that scan several oxidations states per element, as currently under development (see talk of M. Callsen). With these concerns:
  - Some want a set with predominantly crystals that do exist in nature (“what’s the use of a Ba-crystal with oxidation state +5?“), while others claim that crystals that do not make sense chemically are a good stress test for the transferability of PAWs and pseudopotentials as well as for revealing bugs.
  - Some want a set with different crystal structures for the same oxidation state (➔ this is what the GBRV-set currently delivers).
- A set of high-symmetry crystal is perhaps not needed at all. Why can’t we fill random boxes randomly with atoms, and use these as test set? Such a set would lend itself easily to verify forces.
- Another way to have lower symmetries in the set, is to consider small molecules, surfaces or vacancies.
- Not all functionals go together well with all elements. For instance, with PBE f-electron crystals are difficult to converge. Shouldn’t one avoid f-electrons in a set meant for PBE?
- A set specifically containing magnetic systems would be useful too.

Having a larger set, with multiple crystals containing the same element, offers new possibilities. For instance, formation energies can be calculated. This is new information, that was lacking in the initial Delta-set with only one crystal per element.

By calculating free atom energies for every element (which can be painful for some codes), also cohesive energies become readily available.

Taking all the above considerations into account would lead to a very large test set, too large for routine practical use. It would also be paralyzingly large to start building it. We need instead several smaller sets, each built for a specific purpose, independent from other sets.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 2: pseudopotential development

A discussion developed around the deliberately provocative question: “Can this community afford to continue building ever more pseudopotential libraries? Shouldn’t we focus more on all-electron methods instead?” Some arguments that play a role:

- Having multiple approaches (pseudopotentials vs. all-electron, one pseudolibrary vs. another, …) is an asset. It shows this community is healthy, and it offers a biodiversity on which we can draw when confronted with new situations.
- In all-electron methods, the challenge is shifted from creating a good pseudopotential to creating a good basis set. This is a kind of conservation law of hard work.
- It is not such that all-electron codes are static and stable, while pseudopotentials keep developing. Also in all-electron codes new evolutions are happening (cfr. HDLOs in LAPW).
- The value of pseudopotentials depends to some extent of the property one is interested in. As a corollary, benchmark sets that are focusing on a particular property could be useful for pseudopotential development.

Benchmark sets that are specifically designed for testing pseudopotentials require different features. It could help, for instance, to keep the k-mesh coarse. As long as the all-electron reference uses the same coarse k-mesh, this is fine. It would make rapid testing much easier. More generally, for future test sets there should be more strict instructions on which k-mesh to use, and which type and amount of Fermi surface smearing.

Apart from the precision by which a pseudopotential calculations reproduces an all-electron result, the execution time for both calculations matters as well. If pseudopotentials do not lead to smaller execution times, there is no reason using them. Their precision will often be a trade-off with speed. Benchmarking speed is not straightforward and can probably not be done in a machine-independent way. Nevertheless, efforts in this respect would be welcome.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 3: workflow and data management

Running benchmark calculations is tedious. Having a workflow manager in which the procedure to run a benchmark set is encoded, makes this a much easier task (human time is expensive, computer time is not). This is also a way to prevent human errors as much as possible.

A similar reasoning applies to convergence testing: if it can be automated via workflows, then more people will do it, and will do it properly. Such a(n automated) procedure to determine numerical settings is to be preferred over providing default settings only, as it can take specific details of an individual case into account.

Having 'all' published DFT results (and also non-published ones) properly stored in databases, not only their results but also the input and output files, would be a rich source of information. It helps carrying over expertise to other users, it helps detecting problems, mistakes or fraud in published data, it allows reusing, recycling or re-analyzing existing data,… It just speeds up research a lot.

Disadvantages of uploading your data to a database are: Which database should I use? Will it survive for sufficiently long? Does it accept the format in which my data are? How easily can it be searched? The Optimade project (www.optimade.org) aims to make all databases interoperable, which alleviates to some extent the pain of having to make a choice.

Possible dangers are that contextless data can be dangerous for non-experts. For instance, a PBE band gap can easily be interpreted by a non-expert as a 'correct' band gap. Or a fast benchmark procedure with e.g. given a coarse k-mesh can only be compared with data obtained using the same coarse k-mesh. Information and warnings can be added, but this never prevents all incorrect use. In addition, too much information may have a deterrent effect.

Finding conflicting information in different databases (or even in the same database) can feel as disturbing. On the other hand, it's better to know there is this spread, rather than using a single value and believing out of ignorance it is the correct one.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 4: beyond total energies

A discussion develops on ‘do we need to test properties beyond total energies?’ Many properties do depend on total energy differences, even complex ones as a phonon band structure. It makes most sense to do this for properties that do not depend (entirely) on the total energy, such as magnetism.

We cannot test all properties thoroughly, hence it is important to choose wisely which one. Proxies should be used wherever possible (e.g. band structure as a proxy for magnetism). Or rather than calculating an entire phonon spectrum, the forces after a simultaneous random displacement of all atoms can be compared across codes. When prioritizing the properties for which a bench mark set will be developed, it could help to select those properties first that matter for applications.

Another discussion starts from the deliberately provocative question whether ever an ideal pseudopotential will be found. The puristic answer is: no. Eventually, there will always be a point at which the scattering properties of the nucleus matter, and a pseudopotential has no info on this. Also the PAW scheme has its limitations, due to the frozen core. When relaxing the core, the PAW scheme becomes de facto an all-electron method.

An interesting insight is that there is no need to invest too heavily into very precise pseudopotentials, because our calculations are anyway not yet sufficiently accurate (=close to experiment). What is the value to be able to reproduce – at high cost – an all-electron calculation exactly, if the predictions that are based thereupon are considerably different from the experimental values due to limitations of the XC-functional?

Are there alternatives to the use of pseudopotentials and plane waves? If we would allow to use other basis sets than plane waves, one could work with much harder pseudopotentials. Multiresolution basis sets behave differently near the nucleus and far away from it, and are becoming available for periodic calculations as well. One drawback is that it is (and always will be) more difficult to develop other features on top of this, in that respect a plane wave basis set is much easier to work with.

Work along such lines require a lot of human effort, beyond what is possible for one PhD. Concerted actions and stimulating the development of joint libraries (libxc, ESL,…) are very much needed.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 5 : accuracy assessment

We are currently in a situation where PBE is the default choice for most/all solid-state codes (in contrast to quantum chemistry, where B3LYP is the default). This is partly due to inherent properties of PBE (it works reasonably well on a reasonable number of properties for reasonably large numbers of crystals), but is also partly due to convenience (John Perdew distributed the PBE subroutines to all major code developers in the nineties, so it was easy to adopt, and that's what code developers did).

Hasn't the time come to go beyond the use of PBE as a default? It is the feeling of several experts that the answer to that question is 'yes'. This is not easy, though, as it requires a judicious assessment by a DFT user about what is the most appropriate choice of functional. By far not all users have that knowledge. Moreover, referees may object against the use of alternative functionals – they too feel often comfortable with PBE only. As all of us are referees too, we should be aware of this and try to set the right example (i.e. not objecting against papers that do not use PBE for a well-argued reason). But it might take a lot of time until the attitude of the entire community will have changed.

A bottleneck is that not all pseudopotentials are available for all exchange-correlation functionals. It is often not too complicated to make the transfer, but someone should do it. This will happen if a sufficiently large number of users asks for it, but this form a democratic demand is not the most efficient way for science to proceed.

Another issue is that many DFT codes cannot calculate all the (newest) features they provide for all functionals. LDA and PBE are usually dealt with for all features, but not always all other functionals. One could say: if users ask for it, developers will implement these features for another functional. That sounds, however, as 'scientific progress by democratic vote'. Science is not democracy.

Other fields of science might give an example about strategies our community could use. In medical science, there the phenomenon of an 'expert panel', that reviews the literature evidence on the best therapy for a given condition, and then formulates a recommendation. Medical doctors do not need to go through all literature themselves, they adopt only the recommendation. A board reviewing the practice at a hospital or by a medical doctor, checks whether the official recommendations have been followed. Translated to the electronic structure community, we could imagine an expert panel that formulates recommendations on 'what is the best functional to use for a given class of materials', or 'what is the best correction scheme for impurities in semiconductors', or 'what is the appropriate procedure to select numerical input settings', not based on their own research but based on an assessment of what has been published. Code users can follow this advice, and refer to it in publications. This will help to win over referees. On the other hand, referees can use these recommendations to check whether a manuscript uses the state of the art of choice of functional.

An effort about expert panels as described above, is difficult work to get funded and published. It can only fly if it would be a community effort. Psi-k could be the organizational level at which expert panels could be formed. The 2020 Psi-k conference could be a good occasion to brainstorm about this, and to get started.

Another way to make progress, rather than recommending the best functional, is to look at scaling relations: if the systematic deviation of two functionals for a given property/crystal class is documented, one can correct for this deviation to get closer to the experimental value. In this way, a fast and simple functional sometimes leads to results that are as good as a more expensive and more advanced functional (probably with the former having a large rescaling, and the latter a smaller rescaling). Codes could implement in their output not only the bare values, but also the rescaled values, as ‘recommended under the proper circumstances’. However, it will be a hard task to educate users to use these values in the right way, rather than cherrypicking what suits them best, perhaps for the wrong reasons. Another complication is that workflow managers may grep a value from the output (a bare or a rescaled one) without the corresponding warning. As increasingly many users do not see the code output as such any longer, but rather the parsed data by a shell or workflow manager, this may lead to delivering misleading information to the user. Alternatively, making the rescaling could be done at the level of the workflow manager, for any DFT code.

Some other aspects that were touched:

- There is often a trade-off between complexity of the functional and complexity of the physical properties. A sophisticated property can be developed on top of LDA/PBE, and that is very useful, even if LDA/PBE is not the best level for that crystal. Insisting on using a variety of better functionals (=better for easy properties) could slow down the development of more involved property modules.
- Experimentalists are used to spend most of their time to performing control experiments. But give experimentalists a DFT code, and they are happy with just the number that comes out of the box, without control calculations. We should invest more in education and tutorials.
- What do we call ‘accuracy’ in a broader context? It’s only valid to compare with experiment if you measure exactly the same thing as experiment is measuring. For instance, if you don’t include electron-phonon coupling in your band gap calculation, you cannot compare to experiment properly. The model system you examine should be sufficiently relevant before you can compare it to experiment in a meaningful way.

The bottom line: more efforts should be done to make the knowledge explicit that is implicitly known and used by some experts in the field. This should be made more visible and accessible for regular users and referees.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 6: pushing numerical convergence

When it is about the limits of numerical convergence, one will run into the problem that the same code run on different computers will lead to (slightly) different results. In this respect, the toy DFT code programmed in the Julia language (https://julialang.org/), as presented by A. Levitt in his talk, can be relevant (apart from other advantages): it can work at quadruple precision, such that differences when run on different machines/architectures should essentially vanish.

The pitfall of multiple minima was touched during the discussion. Only very simple cases with simple functionals (LDA/GGA) have one unique minimum in the self-consist solution process. When studying f-electron systems or magnetic systems, multiple minima are common. Idem when using more complex functionals (HSE, SCAN). The differences can sometimes be subtle, and are not easily spot. This can interfere with precision studies, and the more so the higher the numerical precision is that one aims for. Two calculations might look different, whereas they are perhaps just two different local minima.

Another aspect that was discussed, is: when we do not look at ultimate precision but at 'normal' precision, for publication quality, how can we ensure that all users run their calculations at the required precision? Some argue that this is not a task that can be left to the user, the software should take care of this. Others argue that this task is too complex to implement in a fool-proof way, as the required precision often depends on the physical property one tries to predict. And even if it could be done for DFT, there are more complex methods (GW) where the 'flat' regime of converged values cannot be reached within a reasonable amount of time. Everything keeps changing, whatever input parameter one touches.

In any case, a more formal procedure to obtain the proper numerical settings for a given level of precision is desirable, be it via educating users or by automating the process.

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).

Session 7 : beyond DFT

When considering codes that are even more complex than DFT implementations, the issue of benchmarking becomes ever more important. At one hand, it is for reasons of independence and neutrality not the most ideal situation of code developers run these tests (as has been the case for DFT codes). On the other hand, doing tests for GW calculations can be much more complex than doing a regular GW calculations, such that it should be mandatory done by developers. Can one do this right this from the beginning when developing more complex methods? (no clear answer emerges)

The question is raised what is the level of precision that is useful. In chemistry, there is the concept of ‘chemical accuracy’. Being much more precise than that, does not really matter. Can similar targets be formulated for solid state physics? Some arguments were given:

- It depends very much on the property you want to predict.
- The precision one aims for, should not be better than the available accuracy of the method.
- Are relative differences sometimes more relevant than absolute differences? Working with a 20 meV band gap is something different than working with a 3 eV band gap.
- Sometimes people are pragmatic: we know that some methods are intrinsically approximate, yet they agree reasonably well with experiment. Perhaps for the wrong reasons, but if it works, it works.
- When comparing codes, a higher precision is required than when comparing one code with experiment (because in the latter case the experimental uncertainty and de imperfection of the XC-functional are involved as well).

This is a digest of a dedicated discussion session held at PQ-DFT 2019. For other digests, videos of all talks and summarizing recommendations, please visit https://pqdft2019.abinit.org/. To access the videos directly on Youtube, visit http://bit.ly/2XFKUCl. Any comments, thoughts or items you want to discuss? Feel free to contact Stefaan Cottenier (stefaan.cottenier@ugent.be) or Kurt Lejaeghere (kurt.lejaeghere@ugent.be).