D3.4

Third release of MAX software: Report on the evolution actions taken in each of the codes

Luigi Genovese, Stefano Baroni, Ivan Carnimeo, Augustin Degomme, Pietro Delugas, Stefano de Gironcoli, Andrea Ferretti, Alberto Garcia, Paolo Giannozzi, Anton Kozhevnikov, Andrea Marini, Davide Sangalli, Daniele Varsano, and Daniel Wortmann

Due date of deliverable	31/01/2022 (**month 38**)
Actual submission date	21/02/2022
Final version	21/02/2022
Lead beneficiary	CEA (participant number 5)
Dissemination level	PU - Public

Document information

Project acronym	MaX
Project full title	Materials Design at the Exascale
Research Action Project type	European Centre of Excellence in materials modelling, simulations and design
EC Grant agreement no.	824143
Project starting/end date	01/12/2018 (month 1) / 31/05/2022 (month 42)
Website	http://www.max-centre.eu
Deliverable no.	D3.4
Authors	Luigi Genovese, Stefano Baroni, Ivan Carnimeo, Augustin Degomme, Pietro Delugas, Stefano de Gironcoli, Andrea Ferretti, Alberto Garcia, Paolo Giannozzi, Anton Kozhevnikov, Andrea Marini, Davide Sangalli, Daniele Varsano, and Daniel Wortmann.
To be cited as	Genovese et al. (2021): Third release of MAX software: Report on the evolution actions taken in each of the codes. Deliverable D3.4 of the H2020 CoE MaX (final version as of 21/02/2022). EC grant agreement no: 824143, CEA, France.

Disclaimer

This document's contents are not intended to replace consultation of any applicable legal sources or the necessary advice of a legal expert, where appropriate. All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user, therefore, uses the information at its sole risk and liability. For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors' view.

Contents

1 Executive Summary

The present deliverable describes the status of implementation of Work Package 3 (WP3) "*Code Evolution: exploiting algorithmic advances enabled by the exascale transition*". As described in the previous deliverables D3.1-3, WP3 deals with the implementation of algorithmic solutions and functionalities in the MAX flagship codes, in order to best exploit pre- and exascale architectures. In the previous deliverables of WP3 we have presented the identified "algorithmic advances" in the form of a software development plan (D3.1 [1]) and a first and second year release report (D3.2 [2] and D3.3 [3]). The former includes a list of the planned algorithms for each code, a classification of each algorithm according to selected criteria, and a timeline in the form of a Gantt chart.

In this document we present the final advancements of the actions ongoing in this last year of development. We do this on a per-code basis. We also highlight how the above activities are expected to provide/receive input to/from other technical Work Packages of MAX .

Each MAX flagship code has implemented several new algorithmic advances, including advanced numerical approaches as well as new physical materials properties. A first class of such developments is related to algorithms for robustness and resilience (including improved diagonalization algorithms and direct solvers for the Kohn-Sham problem, or improved charge density mixing schemes). A second class of developments is focused on the exploitation of pre- and exascale machines by enabling calculations previously considered out-of-reach or just too demanding (such as, e.g., RPA-based XC functionals and forces, Coulomb kernels in LAPW, or full frequency GW calculations using multi-pole approximations). Last, more advanced algorithms designed towards the exploitation of large scale pre- and exascale machines were developed and implemented (including e.g. exact exchange for $\mathcal{O}(N)$ calculations, fragment approaches for extended systems, or real–time propagation with atomic motion within many-body perturbation theory).

2 Introduction

WP3 addresses the progressive adaptation, in terms of newly available algorithms and calculations of physical observables, of the MAX flagship codes (and in turn of the related research community) to the advent of pre- and exascale machines. As discussed in the Software Development Plan D3.1 [1], the algorithmic advances targeted by WP3 are interconnected to the activities of other Work Packages, e.g. via requirements on libraries or code restructuring (WP1), or by the extreme scale demonstrators (WP6). Such dependencies represent pre-requisites that need to be taken into account when defining or updating the implementation schedule for each algorithm/feature (action).

In the present document, we first summarise the main classes of algorithms included in the WP3 software development plan, and classify each planned implementation task. Then, on a code basis, we present the status of algorithmic development and implementation, including validation data where relevant.

3 Algorithmic Activities

In this Section we present a detailed description of the WP3 activities focused on the development and implementation of algorithmic advances in the MAX flagship codes and libraries, as occurred in the period M24-M36 (Dec 2020 – Nov 2021). This document is therefore an update of the document D3.3 [3] released at M24.

It is important to remark that the development of each algorithm or feature identified in WP3 has a strong connection with software development activities from WP1 and WP2. For instance, libraries modularised by WP1 may represent inputs to WP3 or may be the container for the output of the algorithmic developments, while newly developed algorithms may require GPU porting and represent therefore input for WP2 work. Similarly, WP3 actions in some cases may provide new code features useful for the activities of WP5 and WP6, or may receive from these WPs a list of requirements and needs. We have presented and summarised these connections in the revised version of the deliverable D3.1 [1] Software Development Plan.

Besides a description of each planned action, for each code, we provide the relevant portion of the WP3 Gantt chart, which illustrates the status of the algorithmic actions that were organised in the software development plan. When needed, Gantt charts have been updated according to the current status of development of each action. A per-code presentation of the main achievements follows.

The WP tasks are associated are defined as follows:

T3.1 Software-related fault resilience algorithms and solutions. Within this task we intend to *prevent* a potential waste of computing time due to incorrect behaviour of the code or incorrect setup of the input parameters. Actions associated to this task are aiming at, for instance, detecting run-time misbehaviour and providing a fallback solution, or correcting code weaknesses in certain regimes; for example, by preventing the code to run in a regime for which a selected functionality has not been conceived.

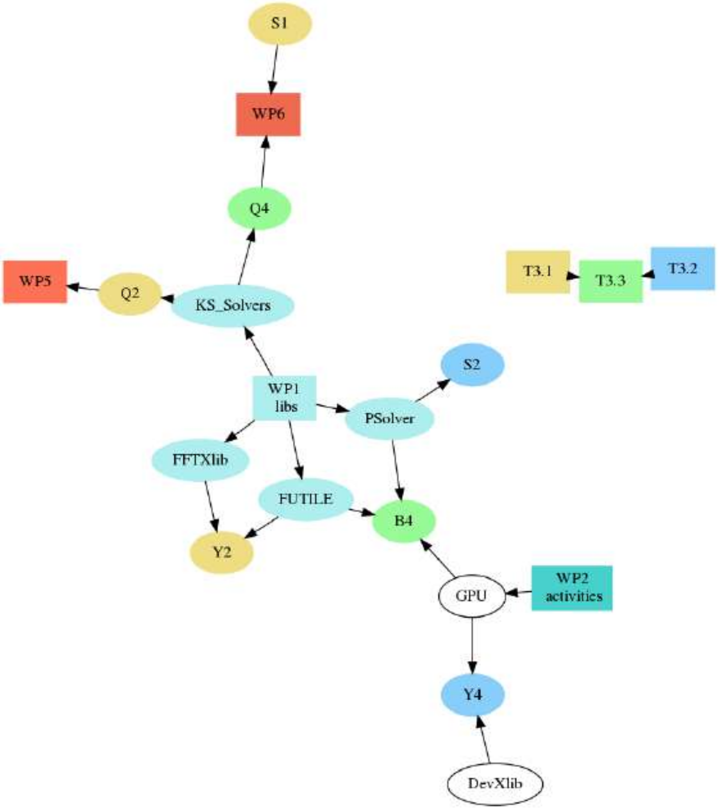
T3.2 Enabling new code functionalities with an exascale mindset. The goal is to try new solutions that are made possible by the advent of new supercomputers, that have up to now been considered unfeasible or impractical.

T3.3 Exploitation of new algorithms at the pre-exascale. The idea is to "think differently" the development actions, knowing the technical features that are nowadays available in the pre-exascale machines, and foreseen for exascale architectures. High thread concurrency and increasing computational workload will be the run-times associated to these algorithms. Actions in this tasks will pave the way towards novel investigation paradigms and solutions.

Another classification of the various activities of WP3 can be devised in terms of *Stage of advancement* of the actions. We present here such stages and their relation with the other work packages. There is clearly an interplay between the tasks and the progressive difficulty of the corresponding actions. For example, it is likely that some actions of T3.3 would be associated to Stage 3. We have provided the overview of the panorama of code activities in a table, which has been presented in D3.1 [1] as well as in the project mid-term report.

Stage1: No prerequisites, direct implementation. The algorithmic actions to be implemented do not have relevant dependencies on libraries or code restructuring, and can be directly addressed.

Objectives: understand the limitation induced by the code infrastructure we had at the beginning of the project.



The actions indicated correspond to the following:

Q2 Direct minimisation

Y2 Mixed precision algorithms

S2 Exact-exchange

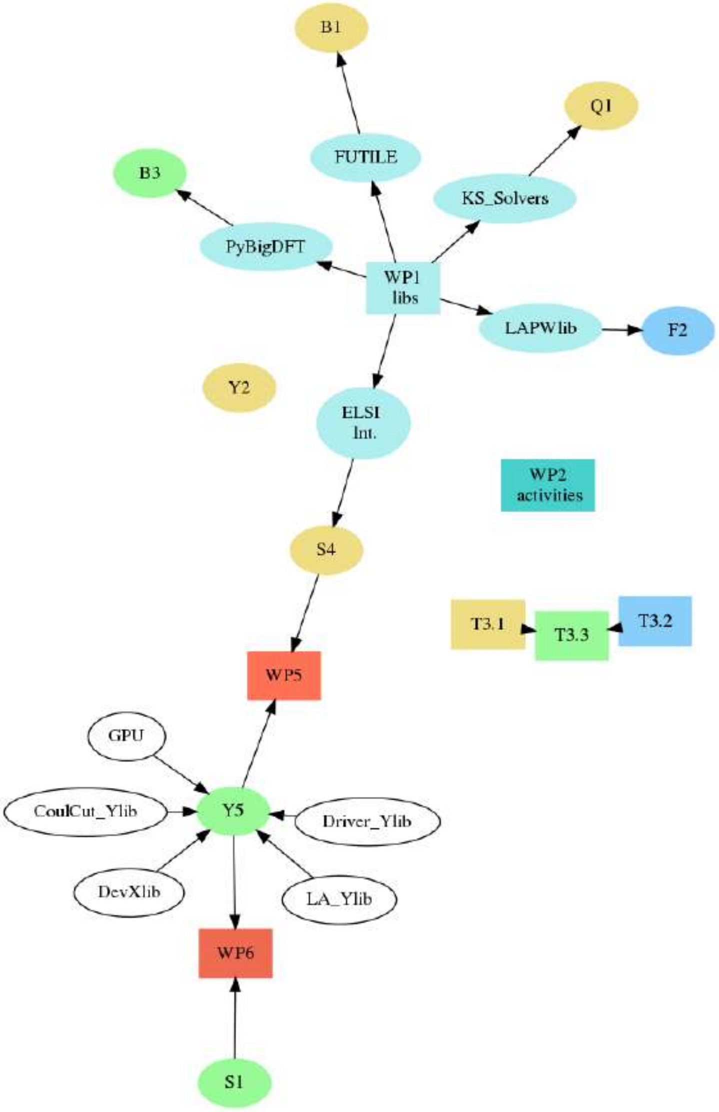
Y4 Full-frequency GW

B4 Exact exchange for $\mathcal{O}(N)$

Q4 Localised inner-projection method for EXX

Stage2: Mild prerequisites. This Stage addresses algorithms and physical features that only have limited dependencies on WP1/WP2 libraries, and may require only minor code restructuring at the global level.

Objectives: identify and promote new user experiences, that connect HPC with novel approaches for the end-user.



With:

B1 Approaches for Input-files

S4 Automatic solver dispatch

Q1 Improved diagonalisation algorithms

Y5 Advanced self-energies from MBPT

Y8 Convergence accelerators for k/q grids

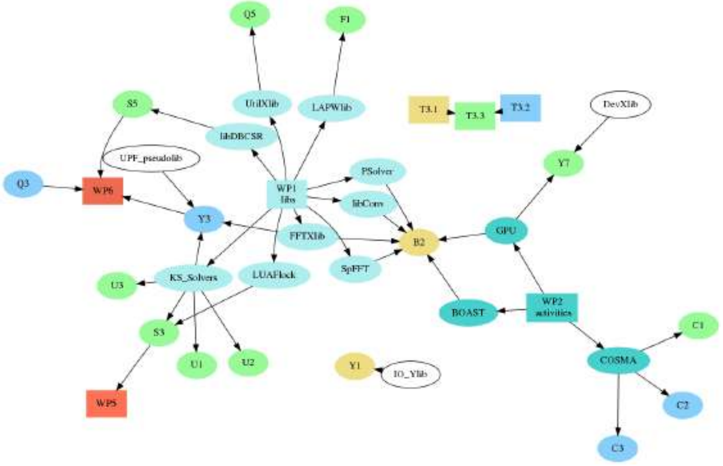
F2 Coulomb kernel in LAPW

S1 Basis-set contraction

B3 Fragment approaches for extended systems

Stage3: Strong prerequisites and code restructuring. In this stage, long term and extended actions (algorithms/features) are addressed.

Objectives: bring awareness of how to conceive scientific codes in order to simplify their usage and maintenance.



With:

B2 Mixed-precision techniques

Y1 Restart structure, parallel IO for BSE and QP

F3 Improved charge-density mixing schemes

Q3 RPA-based XC functionals

Y3 Real–time propagation with atomic motion and interface with QE

C2 Transform CP2K matrices to COSMA layout

C3 Switch to COSMA in RPA calculations

S3 Improved algorithms for SCF

S5 Re-design $\mathcal{O}(N)$ solver

Q5 Adaptive parallelisation schemes

Y6 YAMBO without empty states

Y7 Real time parallelisation

F1 Restructuring of the Hamiltonian

C1 Integrate COSMA library into CP2K

U1 Conjugate gradient method

U2 Proximal gradient method

U3 Advanced density optimisers with QUANTUM ESPRESSO and CP2K.

3.1 BigDFT

B1 Input-file manipulation and handling: wildcard approaches.

This task has been completed in due time, and its behaviour described in D3.2 [2].

B2 Mixed-precision techniques for convolutions and Poisson Solver. A working wrapper for mixed precision techniques has been designed and proven operational for both the FFT and convolution kernels. Its implementation in the stable code release has been postponed to include such wrapper technique also in the new `liborb` library, which is a new package designed in the context of WP1. The main reason is the handling of the memory allocation needed to work in mixed precision. We would like to only allocate buffers which are large enough to host double-precision results and employ the same memory space to also perform single-precision calculations, by performing a in-place conversion when needed. In order to do that, the wavefunction pointers, which explicitly appear in many portions of the code, should be handled differently, with numerous code-intrusive operations, especially at the FORTRAN level. We believe that the inclusion of the `liborb` library will simplify the code factorisation that is needed to employ such technique in production.

B3 Usage of Pseudo-Fragment approaches for extended systems in the Support Functions formalism.

The task has been completed in advance with respect to the software development plan. Its description can be found in D3.2 [2].

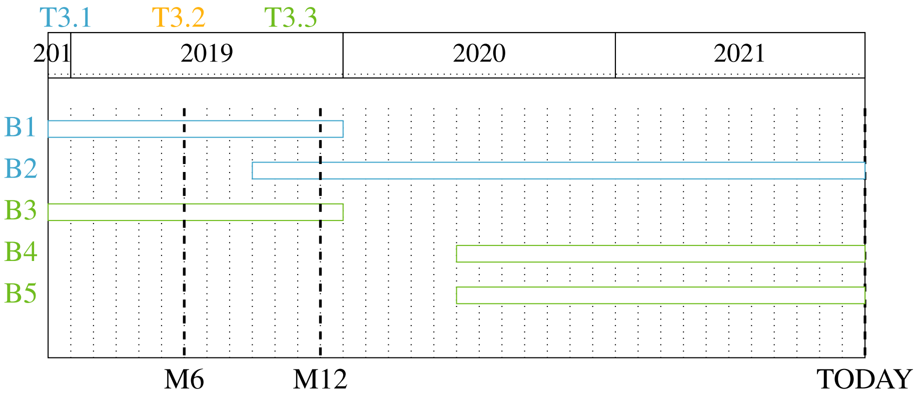
B4 Exact exchange for $\mathcal{O}(N)$ implementation. Our investigation on this action has gone further, by implementing a protocol to compare the $\mathcal{O}(N)$ description of a system across the PBE and PBE0 functionals. In particular, given the complexity reduction protocol implemented in BigDFT, we considered also an alternative implementation based on another localised basis set code (NTChem, developed at RIKEN, Japan), which can handle large PBE0 calculations. Such complexity reduction protocol has been employed in various contexts in the present project (particularly for what concerns WP6), and relies on indicators which enables to identify system fragments from coherent physico-chemical indicators.

We have in this way verified, for various proteins, that the fragmentation derived from a PBE calculation is completely consistent with that obtained using a PBE0-level calculation. This is an interesting result per se. In fact, for a large system, it appears reasonable to employ large PBE calculations to identify the fragmentation of such system, and then reprocess the energies of such fragments with PBE0 functionals. The present set of considerations constitutes the ground basis of a “multi-accuracy” protocol that will be the object of a forthcoming publication.

B5 New: Complexity Reduction Approach for Large-Scale systems. This new action has been designed and identified during the pandemics. The capability of BigDFT to treat large systems has been applied to systems of relevance in SARS-CoV-2 research. Thanks to this case-study, we have implemented an algorithm where: (i) a system of many thousand atoms is simulated and *automatically* fragmented in different units, made of group of atoms; (ii) such units are defined by first-principles indicators (as already indicated in the paper related to task B3) that enable the possibility to decompose the system observables and define the interaction among the fragments. Such data are then serialised as a post-processing of the calculation and employed to perform analysis of the main features driving the interactions of the assemblies characterising the system. Such approach have been successfully applied in the context of drug-design of peptidic inhibitors (see https://doi.org/10.1039/D1SC03628A), and we are now employing the same approach in the analysis of antibody-antigen (AA) assemblies.

In Figure 1 we summarised the details of the QM fragmentation computations for the four AA assemblies provided by the PDB ids 1BVK, 2VIS, 1IQD and 2JEL, previously processed by a polarizable-force field MD, by means of a schematic representation of the resulting 'local AA interaction patterns' allowing one to readily identify the epitope/paratope residues that contribute to globally stabilise/destabilise an AA assembly as well as to identify pairs of residues interacting at short range within the largest snapshot clusters (i.e. the set of most probable atomic conformations of epitope/paratope regions). The residues of the latter pairs can belong either to both the AA entities or only to a single one: a set of interconnected pairs correspond thus to a local interaction network of residues at the interface of an AA assembly. To complement the data provided by the local interaction patterns, we are presently looking to define a schematic representation that allows one to readily identify residues distant from the epitope/paratope regions and that also noticeable affect the assembly stability (a priori the effects of these residues on the AA stability correspond to long range electrostatic interactions).

Last, we plan also to test the numerical workflow that allows a "non expert" to perform all the simulations (MM and QM) at once by sending the main files needed by our approach from a remote server to the TGCC-IRENE ROME partition. We are presently finalising the main scripts of that workflow. This represents a pivotal step to facilitate the use and the dissemination of our HPC simulation tools within a large audience of academic and industrial potential users. Our intensive computational tests performed this year on an overall large set of different AA assemblies show that we need about 500k hours on average to quantum-ly post process 100 snapshots corresponding to a 600 residue assembly.



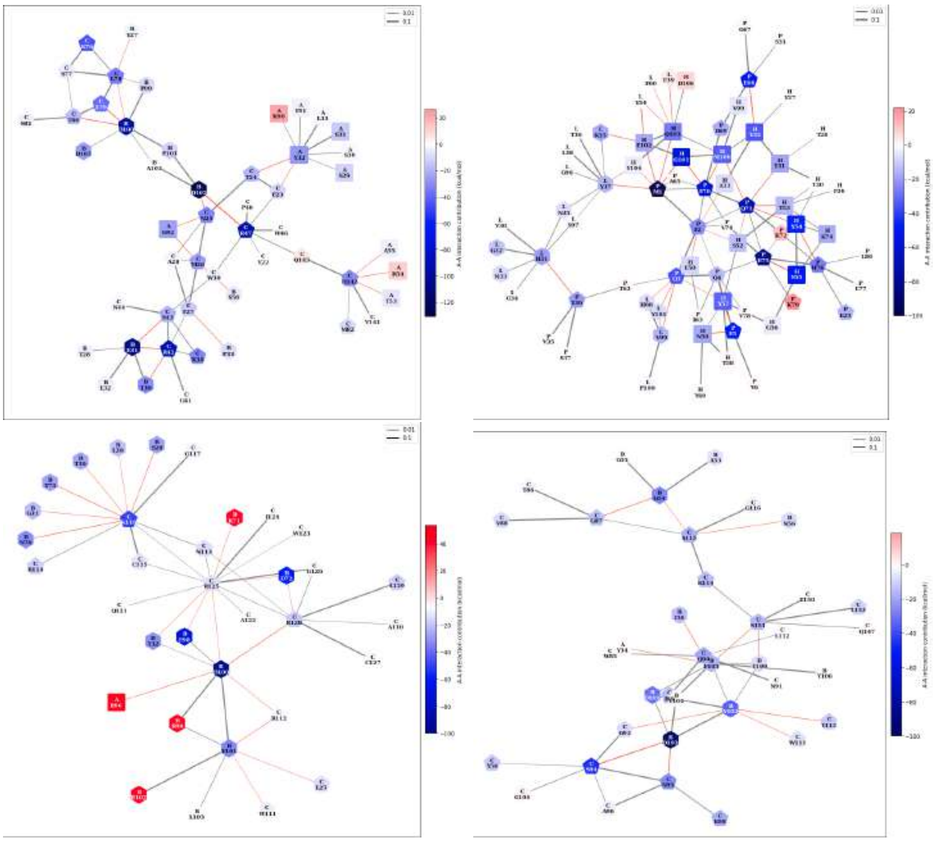


Figure 1: Local AA assembly patterns for the analysed assemblies (1IQD, 2JEL, 1BVK, and 2VIS, respectively). In blue and red: residues favouring/disfavouring the assembly stability, respectively. The darker is the colour the stronger is the stabilising/destabilising effect of the residue. Black lines interconnect residues exclusively of the antibody (or of the antigen) that interact at short range. Red lines interconnect pairs of residues belonging to both the antibody and the antigen and which interact at short range. The assembly dissociation constants Kd are expressed in nM. The antigens are denoted A (1IQD, 1BVK and 2VIS) and P (2JEL).

3.2 SIESTA

S1 Basis-set contraction.

We have completed the implementation of an algorithm to contract a given SIESTA basis set to another one of reduced cardinality. In most circumstances, one would target a *minimal* basis set. A contracted basis function $|\phi_i\rangle$ can be expressed as a linear combination of primitive functions $|\chi_q\rangle$:

$$|\phi_i\rangle = \sum_q a_{i,q}|\chi_q\rangle. \tag{1}$$

The contraction coefficients $a_{i,q}$ are optimised by an extension of the scheme proposed by Ozaki [see Ozaki, Phys. Rev. B **67**, 155108 (2003)]. In this scheme, the optimisation of a given coefficient is based on the following gradient:

$$\frac{\partial E}{\partial a_{i,q}} = 2\sum_j \langle\chi_q|H|\phi_j\rangle \cdot P_{i,j} - \langle\chi_q|\phi_j\rangle \cdot E_{i,j}, \tag{2}$$

where $P_{i,j}$ are the elements of the density matrix and $E_{i,j}$ are the elements of the energy density matrix. P, E, and H must come from a converged SCF calculation, therefore we need two main loops: an inner one for the SCF iterations, and an outer one for contraction coefficient optimisation. Convergence is achieved once the gradient above is below a certain threshold.

In the original implementation, only functions with same l and m quantum numbers are contracted. The coefficients $a_{i,q}$ are optimised for a given geometry. However, this optimisation is performed only at the first step of a molecular dynamics simulation, and thus the coefficients remain unchanged during the rest of the simulation.

In the current SIESTA implementation, instead, *all* the primitive basis orbitals belonging to a given atom can participate in a contracted orbital belonging to that atom, allowing us to drastically reduce the number of contracted orbitals. To show a practical example, one could take a carbon atom with a DZP primitive basis set. This primitive set has 13 functions: 2 s, 6 p, and 5 d for polarization. In Ozaki's original scheme, those 13 functions would be reduced to 9 (1 s, 3 p, 5 d), while SIESTA's current implementation allows us to reduce this even further to the minimal basis set: just 4 contracted orbitals. This greatly reduces the cost of Hamiltonian diagonalization, although this is tempered by the need of a larger number of SCF steps to properly optimise the contraction coefficients.

Another difference with the original implementation is that we allow for the re-optimisation of contraction coefficients every N-th molecular dynamics step, thus providing a better description overall. The initial guess for the contraction coefficients is obtained by a decomposition of the density matrix using a rank-revealing QR method.

The new scheme provides a boost in efficiency, and at the same time, through the coefficients of the "hybridised" atomic-based orbitals, offers extra chemical information that we are now beginning to explore.

S2 Exact-exchange.

The reference implementation has been completed, and it is ready for the general testing and gathering of feedback by the community. Substantial HPC resources are needed even for medium-sized systems, due to the very large number of four-center integrals involved in the computation of the Fock operator.

S3 Convergence improvements.

We have in place the basic infrastructure of Lua hooks that can enable arbitrary remediation recipes for lack of convergence or slow convergence, and an associated proof-of-concept recipe. As a production-level demonstrator is implemented in the next stage, it can impact the overall degree of successful completion of calculations, with particular relevance for the high-throughput workflows in WP5.

S4 Break-even points for various solvers and automatic dispatch based on learned heuristics.

Work has already started taking advantage of the recently implemented interface to the ELSI library of solvers. A uniform interface is available to the PEXSI, NTPoly, and OMM (cubic scaling version) solvers, as well as to diagonalization (ELPA). We are designing a new "performance model" along the lines of that developed for QUANTUM ESPRESSO.

S5 Re-design of the legacy $\mathcal{O}(N)$ solver.

This task has been completed, and we have gathered performance information in medium-to-large systems (see Fig. 2). This action re-implements the linear-scaling Orbital Minimization Method (OMM) with localized electron orbitals, using the sparse matrix-matrix multiplication library DBCSR from WP1/WP2. The interface has been abstracted through the intermediate gluing library MatrixSwitch. Actually, through MatrixSwitch it has been possible to offer a dual dense/sparse implementation, with common driver code and appropriate backends (Scalapack and DBCSR), see Fig. 3 for more details. Interfacing to DBCSR involves the use of chemically meaningful orbital blocks, which are then distributed over the available MPI ranks. All the re-distributions to-and-from the SIESTA native scheme are handled transparently by MatrixSwitch (see Fig. 4).

This work can impact the WP6 demonstrators that deal with gapped systems, and also serve as the starting point for the implementation of alternate linear-scaling solvers that reuse the sparse matrix support backend.

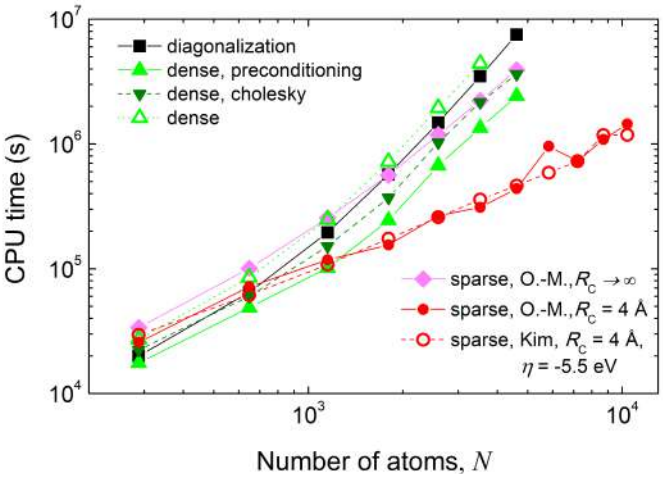


Figure 2: CPU time (in s) for 4 molecular dynamics steps for a single boron nitride layer as a function of the number N of atoms in the system computed using different approaches.

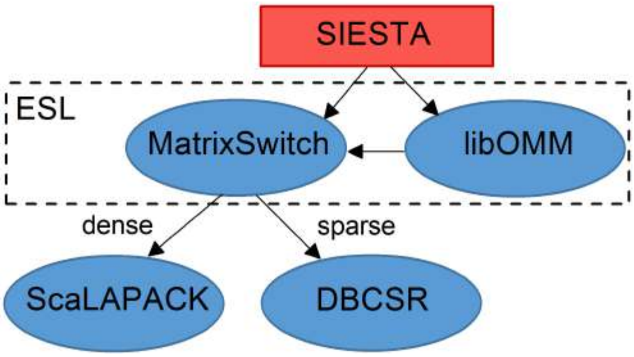
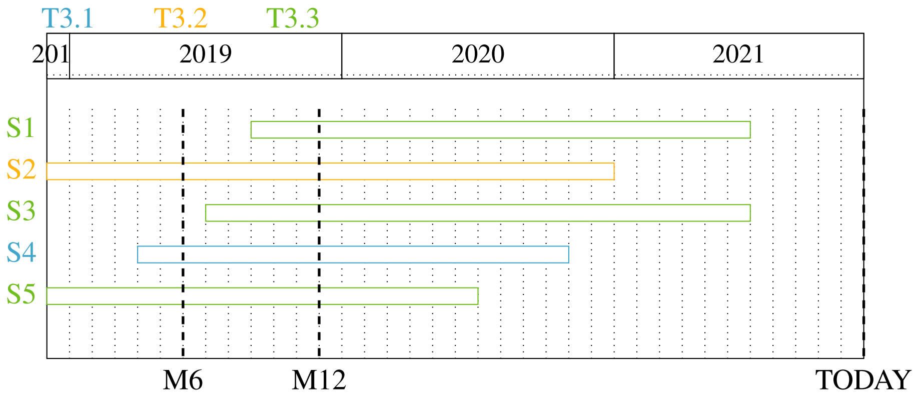


Figure 3: Use of libraries within the revised OMM solver in SIESTA. ESL stands for the Electronic Structure Library.



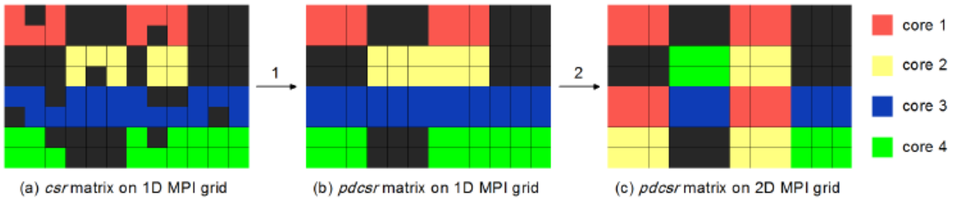

1
2
core 1
core 2
core 3
core 4
(a) *csr* matrix on 1D MPI grid
(b) *pdcsr* matrix on 1D MPI grid
(c) *pdcsr* matrix on 2D MPI grid

Figure 4: An example of the matrix format conversion from the *csr* format used in SIESTA to the *pdcsr* MatrixSwitch format handled with DBCSR: (a) a *csr* matrix distributed on a 1D MPI grid with 4 CPU cores, (b) the *pdcsr* matrix with 2 × 3 blocks distributed on the same 1D MPI grid and (c) the *pdcsr* matrix with 2 × 3 blocks distributed on a 2 × 2 2D MPI grid. The elements of the 8 × 12 matrix are shown by squares. The black squares are zero elements that are not stored. The red, yellow, blue and green blocks are stored on cores 1, 2, 3 and 4, respectively.

3.3 QUANTUM ESPRESSO

Q1 Improved diagonalization algorithms. The task was successfully completed by M24. During the project we have worked on several alternative iterative solvers:

- ParO (Parallel Orbital) update scheme;
- Projected Preconditioned Conjugate Gradient (PPCG);
- Residual minimisation-direct inversion in the iterative subspace (RMM-DIIS) scheme.

The new solvers are now all included in the production releases of the `KS_solvers` library and QUANTUM ESPRESSO (`qe-7.0`). Specifically, we have worked on the reduction of memory requirements, elimination of redundant evaluation of the Hamiltonian, minimisation of the communication time, and robustness of the algorithms. On this latter side, a mixed usage of solvers is already implemented at production level for the RMM-DIIS scheme. The alternate use of different solvers has demonstrated to be very effective in handling and recovering instabilities and will be extended to all the schemes.

The efficiency of these addictions in the production codes has been field-tested. In particular for large size benchmarks (see report D4.5 [4], Fig. 10), the RMM-DIIS algorithm mitigates significantly the diagonalization bottlenecks with an improvement of the scalability. All these algorithms allow for an improved band-parallelism, and will have significant advantages from the ongoing and planned work on the data distribution over the band-groups

Q2 Direct energy minimisation schemes. This task aims at the implementation of global minimization schemes for the energy functional. Although generally slower, the global minimization is desirable as a more robust fall-back solution for the occasional failures of the self-consistent optimization scheme. The task is still ongoing in collaboration with the CAS group in Bejing. The experimental implementation in QUANTUM ESPRESSO has been finalised and extended to USPP and PAW pseudopotentials. The new minimisation scheme has been tested on a large variety of systems, including many which have been singled out by an high-throughput screening as difficult to the SCF minimisation. Two papers reporting these results are in preparation.

Q3 RPA-based advanced exchange and correlation functionals. All the parts of the imaginary frequency integration workflow have been defined. Such calculations will require the concurrent execution of a very large number of computationally intensive tasks. The further progress in the activity strongly depends on the finalisation of the GPU porting of the `LR_Module` of QUANTUM ESPRESSO. Most part of the work allocated for this activity is thus at the moment dedicated to code-refactoring and GPU porting of the linear-response parts of the code.

Q4 Extension of the localised inner-projection method to EXX. The use of the Selected Columns of the Density Matrix [5] (SCDM) method allows to recast the density matrix in terms of localized orbitals. The exact-exchange operator (EXX) expressed in terms of these orbitals can be restricted to the overlapping orbital pairs. This may reduce the computational cost of EXX evaluation, in particular for very large systems. In this task we have developed a version of SCDM more feasible for molecular dynamics simulations. The overlap matrix at each step is computed from the previious step's overlap matrix, the most expensive operation needed by this parallel transport algorithm is a SVD on a $N \times N$ matrix, much less expensive than the QR decomposition of the selected column density matrix that has dimensions $N \times nr$ where nr is the size of the real space grid. A working implementation of the algorithm is part of `qe-7.0` .

In the last year we have also worked on the implementation of non-self-consistent calculations with previously computed EXX potentials. This methods will significantly improve the users' experience for the execution of band-structure analysis and post-processing using hybrid functionals. We are working and evaluating the performance of different interpolation schemes, either using the Shirley interpolation approach [6, 7] or the Shankland-Koelling-Wood Fourier interpolation [8, 9].

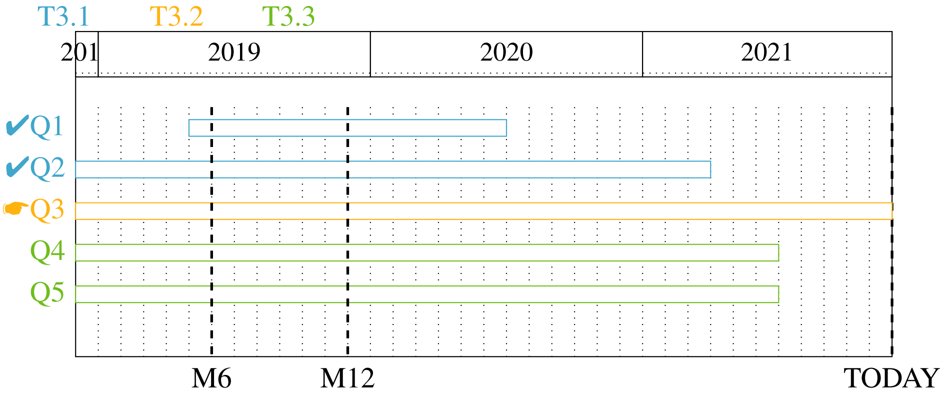
The first outcomes of this task are already available in the QUANTUM ESPRESSO development branch, and those that will be included in next release are:

- *The implementation of a non-SCF scheme for the ACE potential* that allows for the computation of an arbitrary number of empty states, reusing an SCF EXX part from a previous calculation made with the minimal number of needed empty states. Among others, this new feature will streamline MBPT calculations (exploiting sum-over-states) done on top of hybrid functionals, and will significantly reduce the computational cost for the calculation of tight-binding Hamiltonians built on top of hybrid-functional calculations using `Wannier90` .
- *The implementation of a band-structure interpolator based on the Shankland-Koelling-Wood method.* This new feature allow one to interpolate the band-structure results obtained on the irreducible wedge of an uniform mesh, to any arbitrary set of **k**-points.

Q5 Adaptive parallelisation schemes. The task in WP3 has been completed. An implementation of the selector of the parallelization scheme will be added to next release of the qeschema package[1].

Work is ongoing for combining the results of this task with the outcomes of contiguous activities planned in WPs 1,2, and 5.

In WP3 we have implemented and algorithm that takes into account the main parameters defining the calculation size and type, including e.g. the available MPI tasks, the FFT grid dimension, the number of Kleinman-Bylander projectors, the number of **k**-points, and the number of bands. In this simpler model the only machine parameter taken into account is the memory per MPI rank. A more complex machine characterisation can be obtained using the outcomes of WP5 where a model, based on machine-learning, was developed to estimate the time-to-solution of calculations based on the input parameters and on the available number of nodes. This machine-learning model must obviously be trained for each particular platform. An efficient set on inputs for the training have been collected. More details on these results will be reported in WP5 reports. Other work is ongoing in WP1 aimed at refactoring the initialization of the MPI parallelism. On top of this refactoring it will be possible to exploit a Fortran implementation executed directly inside the code.



[1] https://pypi.org/project/qeschema/

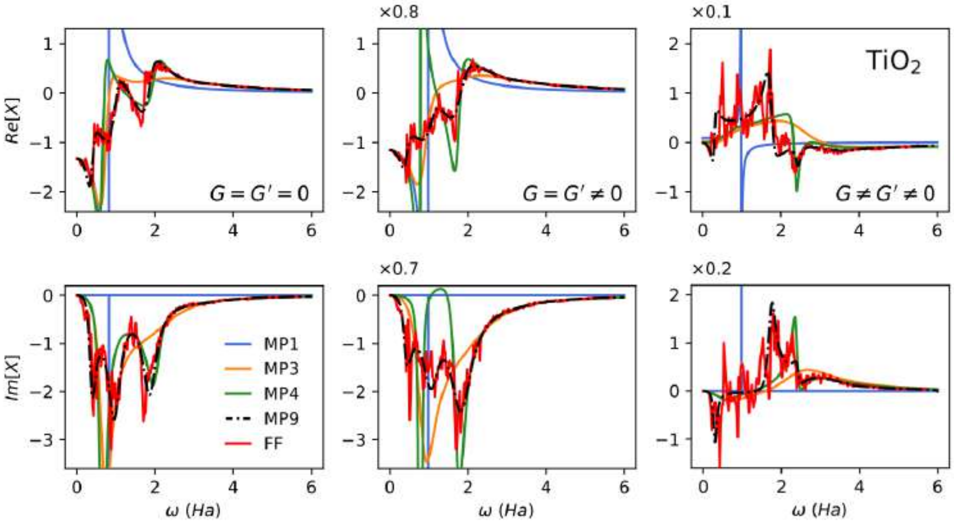


Figure 5: Numerical validation of the multi-pole approximation (MPA) used by YAMBO to represent the dynamical dependency of the screened Coulomb potential W, in order to evaluate the frequency convolution needed to compute the GW self-energy. Results for bulk rutile TiO_2 are shown for different response matrix elements.

3.4 YAMBO

Y1 Restart structure, parallel IO and database re-organization. This action was already almost completed at M12. Nevertheless, the it was taken forward with different activities.

(i) Parallel I/O is now the default mode.

(ii) The module for the I/O was split into two dependent modules, and few subroutines further split to improve the overall modularization of the code (see WP1).

(iii) The features of HDF5 parallel I/O were exploited to reduce memory use. One example is the `qindx_B` variable, which can become very large in systems with many **k**-points and can be hardly distributed in memory, since the MPI tasks often need to access values spread across the whole dataset. By using HDF5 the variable can now be stored in a single large file `ndb.kindx`, and later used as a buffer. This makes possible to distribute `qindx_B` in memory, and allows the MPI tasks to selectively load the missing values from disk when needed.

(iv) The creation of the `ndb.BS_PAR`, which contains the excitonic matrix, was already code in version 5.0. The file can easily become huge (tens or even hundreds of GB). While it can be avoided in some cases to store the data on disk, its use enables to restart calculations where the excitonic matrix is constructed, which are typically very long (thereby easing resilience). With respect to data size, a solution is data compression, which however is not compatible with parallel I/O. With YAMBO 5.1 we have developed a scheme to cope data compression and parallel I/O. The approach requires extra operations by each MPI task and synchronisations to figure out the position of the assigned matrix elements in the large file. The extra operations however are less demanding then recomputing the excitonic matrix, especially in systems with many **G**-vectors (such as 0D, 1D, and 2D systems, or bulks with large super-cells).

Y2 Exploitation of mixed precision algorithms. This task has been completed in due time. Yambo uses mixed precision by default, while double precision can be switched on at compilation time. Multiple tests have been performed to cross check double precision vs mixed precision, and possibly to improve the latter, in different parts of YAMBO as described in D3.2 [2].

Y3 Real-time propagation with atomic motion and interface with QE. This activity is ongoing. A computational scheme based on the Ehrenfest dynamics has been designed and it is currently in implementation phase. The electronic dynamics is carried out at TD-HSEX level exploiting the real-time module already present in the YAMBO code, using Kohn-Sham wavefunctions as basis set. In the Ehrenfest scheme, the electronic dynamics is coupled to a classical equation of motion (EOM) for the ions. We are presently working on the EOM for the ions. The electron-ion interaction, described by the non-local pseudo-potential term of the Hamiltonian, can be reconstructed in the YAMBO code by taking advantage of the QE pseudo library developed within MAX . Thanks to this, we can update the electron-ion term in the EOM for the electrons and realise the Ehrenfest scheme. The approach is suited to model small atomic displacements, such as the ones connected to coherent phonons generated in pump and probe experiments. For arbitrary atomic displacements, testing is needed to asses whether and when new ground state calculations (QE) are required to update the KS basis set during the dynamics.

Y4 Advanced approaches for full-frequency GW. The development of a multi-pole approximation (MPA) algorithm to represents the frequency-dependent properties of the screened interaction $W(\omega)$ has been completed and included in the new YAMBO release. The method relies on an optimal sampling in the complex frequency space and provides an accuracy comparable to full frequency methods at a much lower computational cost. The method has been extensively validated in different prototypical systems such as hBN, Silicon, rutile TiO_2 (see Fig. 5), F_2 molecule, and for all the considered systems around 10 poles (20 sampling points) are enough to reach the desired accuracy. The algorithm has been described in details in a scientific paper published in Ref. [10] (https://doi.org/10.1103/PhysRevB.104.115157). Currently, we are exploring and assessing strategies to extend the methodology to metallic systems.

Y5 Advanced self-energies from MBPT.

- **Beyond-GW methods.** We have benchmarked different beyond-GW schemes in many-body perturbation theory using spherical atoms as a test set. As shown in Fig. 6 we get the best performance from 2B@HF, GW+SOSEX@PBE and GW@PBE0. This confirms the common knowledge that hybrid functionals are good starting points for GW applied to finite systems. The justification of the SOSEX diagram as a cure for the self-screening error of GW seems to be supported by data obtained with the PBE and PBE0 starting points. Moreover, in these cases, resorting to SOSEX provides improved results with respect to GW, as long as IPs are considered. Based on these considerations we

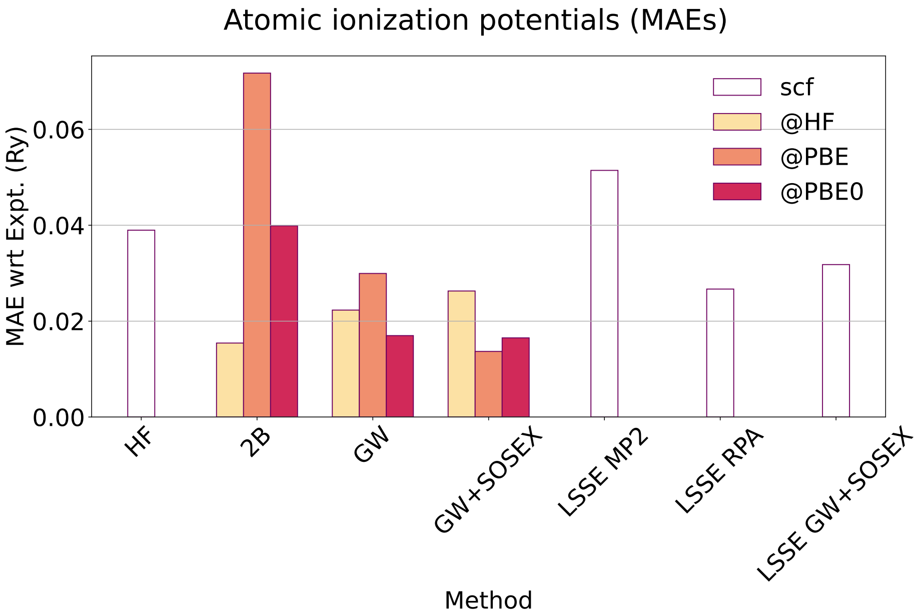


Figure 6: Mean absolute error (MAE) on the ionization potentials for different advanced self-energies methods as a function of their starting point. Transparent columns indicate the IP data obtained from independent-particle self-consistent field (scf) methods.

have decided to postpone the implementation of SOSEX for 3D systems in order to better explore alternative beyond-GW self-energies (e.g. including a full treatment of TD-HF or BSE vertex).

- **GW self-energies plus environment.** The implementation of the GW approximation in the presence of a dielectric environment, based on embedding techniques using a polarisable continuum model (PCM), was finalised. At the moment we are assessing the numerical flow of the implementation (e.g. identifying bottlenecks) and performing a validation of the method against models and realistic systems.

Y7 Real time parallelisation. As already discussed in D3.3, this activity was completed in advance due to re-prioritisation. The implementation put in place at M12 has been extensively tested and numerically validated.

Y8 Convergence accelerator for k/q grids.

- **GW for 2D materials.** Quasiparticle calculations in the GW approximation are known to be particularly challenging for 2D materials since a very dense $\mathbf{k}$,$\mathbf{q}$-sampling of the Brillouin zone is needed to obtain properly converged quantities. To address this issue, we have developed and implemented in YAMBO a new algorithm consisting in combining Monte Carlo integration techniques with interpolation schemes of the screened potential W to accelerate convergence of GW quasiparticle calculations with respect to $\mathbf{k}$-point sampling for 2D systems. The algorithm was extensively validated for a series of 2D semiconductors. In Fig. 7 we report the convergence of the gap of

MoS_2, hBN, and phosphorene monolayer with respect to the adopted $\mathbf{k}$-point grid of the new algorithm, compared to the traditional approach. As evident from the data, a speed-up of at least one-to-two orders of magnitude in terms of number of $\mathbf{k}$-points used is found. As shown, coarse grids are enough to obtain very accurate results, opening the way to studying computationally demanding 2D materials with improved accuracy. The algorithm has been included in the new release of YAMBO and currently work is ongoing to extend the algorithm to metallic systems.

- **Double k-Grid method for solving the Bethe-Salpeter equation via Lanczos approaches.** Convergence with respect to the size of the $\mathbf{k}$-point sampling of the Brillouin zone is one of the main bottlenecks in the calculation of optical spectra of periodic crystals via the Bethe-Salpeter equation (BSE). We have developed and implemented in YAMBO a double grid approach to $\mathbf{k}$-sampling compatible with the effective Lanczos-based Haydock iterative solution. This method relies on a coarse $\mathbf{k}$-grid that drives the computational cost, while a dense $\mathbf{k}$-grid is responsible for capturing excitonic effects, albeit in an approximated way. Importantly, the fine $\mathbf{k}$-grid requires minimal extra computation due to the simplicity of our approach, which also makes the latter straightforward to implement. We have validated the method for both bulk and 2D semiconductors against available experimental data (see Fig. 8). Further details of the implementation can be found in Ref. [11] (https://doi.org/10.3389/fchem.2021.763946). This scheme has the potential of enabling the calculation of optical spectra in semiconducting systems where the efficiency of the Haydock scheme alone is not enough to achieve a computationally tractable solution of the BSE, such as, for instance, large-scale systems with very stringent $\mathbf{k}$-sampling requirements for achieving convergence.

The activities Y6 (YAMBO *without empty states*) is tightly related to the interfacing of YAMBO with core libraries of QUANTUM ESPRESSO. This activity has been rescheduled and slightly delayed to better synchronise with other YAMBO and QUANTUM ESPRESSO development actions (such as those related to GPU porting, WP2, and code restructuring WP1). In particular, to avoid a lot of source code duplication among the two codes, we are currently studying a possible redesign of the way the two are interfaced. This is partly based on the experience gathered and the lessons learnt in coding the TD-HSEX + Ehrenfest dynamics, where the reconstruction of the ionic pseudopotential term from QUANTUM ESPRESSO has been included in YAMBO.

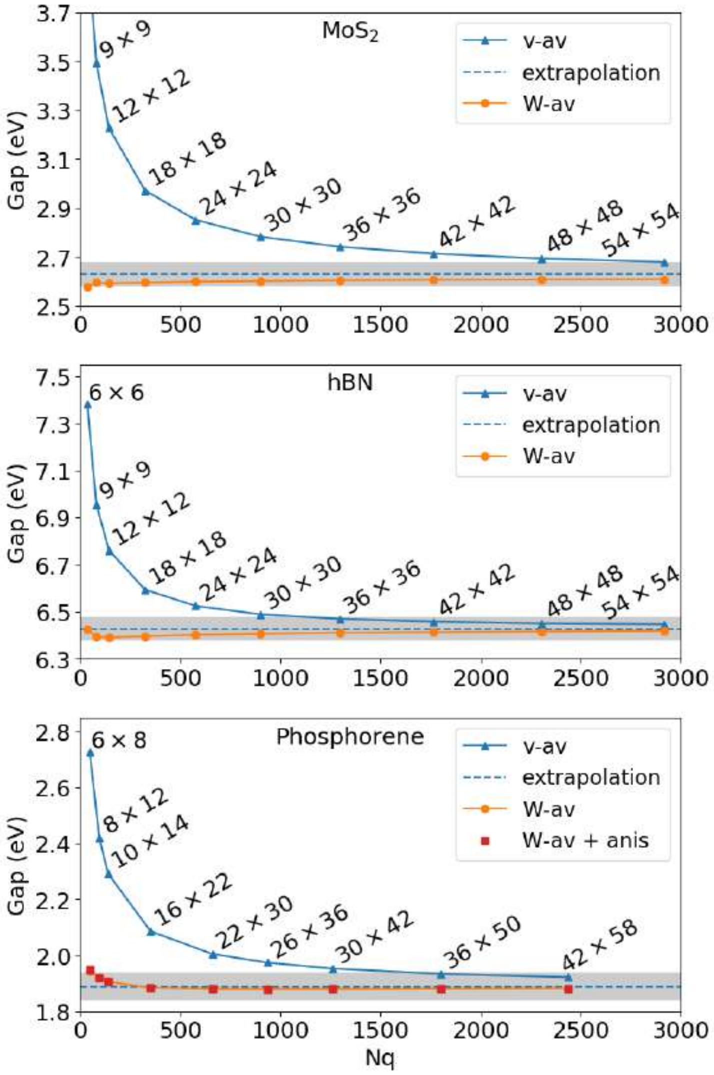


Figure 7: Convergence of the quasiparticle band gap of MoS$_2$ (top panel), hBN (middle panel) and phosphorene (bottom panel) with respect to the number of sampling points of the BZ. Blue lines indicate results obtained with standard integration methods.

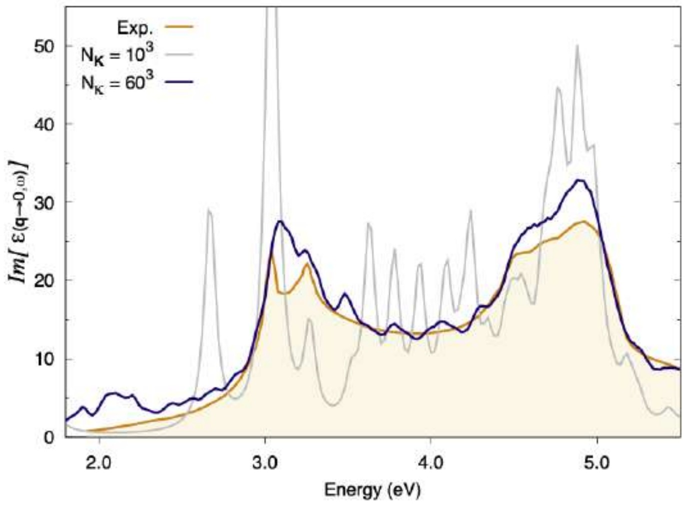
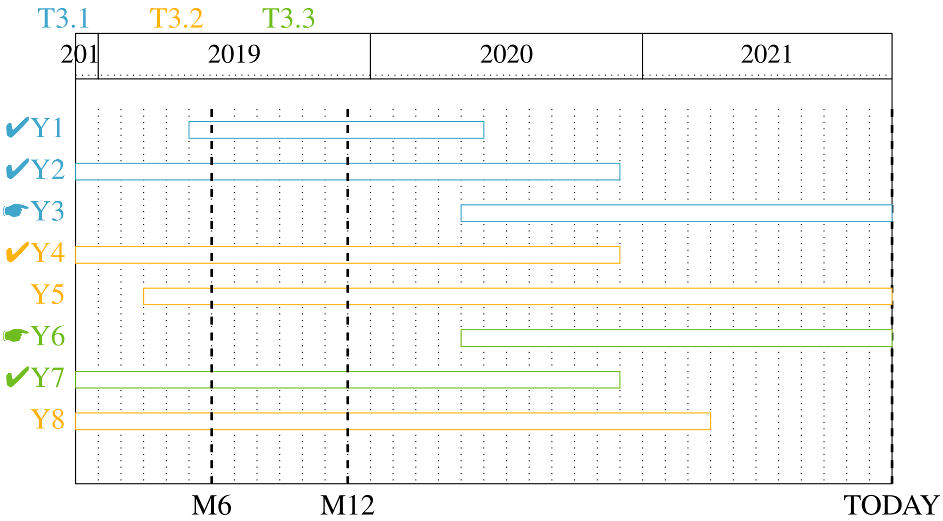


Figure 8: Optical absorption spectra of bulk GaAs. BSE spectra calculated using the double-grid approach are assessed against the experimental spectrum at 22 K. Spectra are calculated on an $10\times10\times10$ coarse $\mathbf{k}$-grid and a denser fine $\mathbf{k}$-grid, indicated by $N_{\mathbf{k}}$. $N_{\mathbf{k}} = 10^3$ corresponds to a standard calculation on a $10\times10\times10$ k-grid while $N_{\mathbf{k}} = 60^3$ corresponds to a double-grid calculation with a $60\times60\times60$ fine grid.



3.5 FLEUR

Out of the implementation tasks related to the FLEUR code, the focus of the current release lies on the finalised activities F2 and F3. In addition, two new aspects of our development have been picked up and are reported below. While these are also already available in the latest release of the code, they also indicate possible directions of future research and development. All code relevant for this work is included in the final release MAX -R6.0 of the FLEUR code available at the FLEUR web-page (https://www.flapw.de).

F1 Finishing the restructuring of the Hamiltonian setup into high-level operations.

This task has been completed in due time, and its description can be found in D3.2 [2].

F2 Evaluation of the Coulomb kernel in LAPW. The algorithmic development associated with this task has been finalised and the effort was now on modularisation (see D1.5) and performance (see D2.3).

F3 Improved charge-density mixing schemes. This work has been finalised and is in production.

F4 Precision Profiles. In HTC as well as in HPC simulations it can be very challenging to ensure proper convergence testing of the results. As the FLEUR code features many numerical parameters due to its complex basis set and all-electron treatment of the electronic structure, a naive approach can require the need to very many of these parameters in a convergence check leading to an unreasonable computational burden. On the other hand, many of these parameters are of course not independent of each other and a lot of knowledge of the convergence properties of the method is available already. Therefore, we started to implement the idea of *simulation profiles* into the input-generator of FLEUR which is used for the initial setup of the calculation. The basic idea is to provide the user with predefined sets of parameters in the form of clearly defined profiles that form a hierarchy of parameterization with increasing accuracy. These profiles on the one hand take into account reasonable constraints between different parameters adjusted to the structure at hand and on the other hand can be used to encode the experience mentioned already. The corresponding code is already available in the final release of the MAX project and first profiles have been defined. As these profiles are now specified in simple text form, further profiles can easily be added.

F5 Calculation of the local Green function. Many electronic properties can be expressed in a mathematical simple form by the Green function of the system. While the full Green function of course is a very complex and large object when expressed in the LAPW basis set used in FLEUR, its local projection on the MT spheres is both relatively easy to obtain and much more compact in its representation. Hence, we implemented this quantity and a couple of property calculators taking advantage of the Green function. In detail, we already have these properties available in the new release of FLEUR:

- Heisenberg exchange constants: these often called J_{ij} parameters are an important ingredient in our multi-scale simulations of magnetism.

- Dzyaloshinskii-Moriya interaction: by simulating the effect of the spin-orbit interaction in non-collinear magnetic setups also these antisymmetric interaction parameters can be evaluated.
- Magnetic torques: The torque acting on a magnetic moment is e.g. important for spin-dynamics.
- Magnetic Anisotropy Energy: using the magnetic torque also this quantity can be evaluated.

3.6 CP2K

The work on CP2K fits into task 3.2.1 (Implementing new physical functionalities which are possible with large scale machines) of the work package. In order to make further progress with implementation and performance optimisation of the wavefunction-based correlation methods implemented in CP2K, we have developed a tensor contraction library for sparse data sets. The library has been implemented as an extension of DBCSR, our sparse matrix multiplication library. Based on the DBCSR tensor contraction library, the low scaling implementation of RPA and GW in CP2K could be improved (see Fig. 9). The library is also used in the gradient code for double-hybrids and RI-HFX (not yet published; arXiv:1910.13555 preprint [12]). The work is also presented as a PhD thesis "Low-Scaling Electronic Structure Methods Based on Sparse Tensor Contraction" by Patrick Seewald, University of Zurich (2021).

C1 Integrate COSMA library into CP2K using an intermediate ScaLAPACK matrix layout. The work on this action point has been successfully completed at M14.

C2 Transform CP2K matrices directly to COSMA layout without a need of a ScaLAPACK storage. The work on this action point has been successfully addressed at M24. We managed to optimise the data reshuffling in the COSMA library and created a standalone communication-optimal shuffle and transpose library COSTA[2].

C3 Switch to COSMA library in RPA calculations. The work on this action point has been successfully completed at M18.

3.7 SIRIUS software development platform

The work on the domain-specific SIRIUS software development platform fits into task 3.1 (Software-related fault resilience algorithms and solutions). The implementation of the direct solvers for the wave-function optimisation is finished and encapsulated in the standalone library NLCGLIB,[3] which is interfaced with SIRIUS (see Fig. 10). The QE/SIRIUS/NLCGLIB was extensively tested on the crystal structures where the classical density mixing scheme doesn't converge. See Fig. 11 for the final results.

U1 Prototype and implement conjugate gradient method. The work on this action point has been successfully completed at M14.

U2 Prototype and implement proximal gradient method. The work on this action point has been successfully completed at M22.

U3 Interface advanced density optimizers with QUANTUM ESPRESSO and CP2K. The work on this action point has been successfully completed at M33.

[2] https://github.com/eth-cscs/COSTA

[3] https://github.com/simonpintarelli/nlcglib

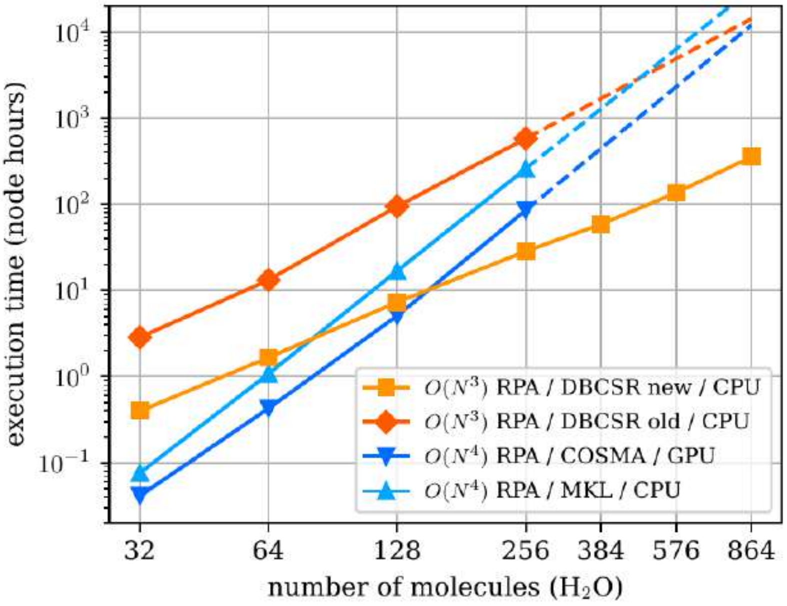


Figure 9: Weak scaling benchmark comparing different implementations of RPA for different system sizes of bulk water. The two low-scaling O(N3) calculations refer to different implementations of the same algorithm, the old implementation based on the DBCSR matrix library and the new implementation based on the tensor API of the DBCSR library. The canonical O(N4)-scaling RPA implementation is mostly PDGEMM-based and we consider an MKL-based configuration and a 3-times faster GPU-accelerated configuration using the COSMA library. The crossover between the best canonical and the best low-scaling implementation of RPA lies shortly above 128 molecules.

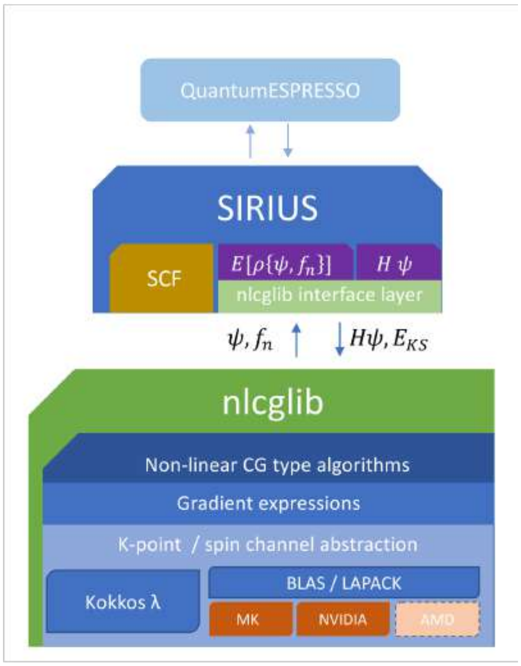


Figure 10: Current software stack implemented for the robust wave-function optimisation in QUANTUM ESPRESSO. QE is interfaced with SIRIUS and calls it as a "black box" to solve the ground state and receive back the wave-functions and band energies. SIRIUS is interfaced with the NLCGLIB that drives the wave-function optimisation. In such way we offer a SIRIUS-based solution for the standard SCF and CG types of density minimstaion.

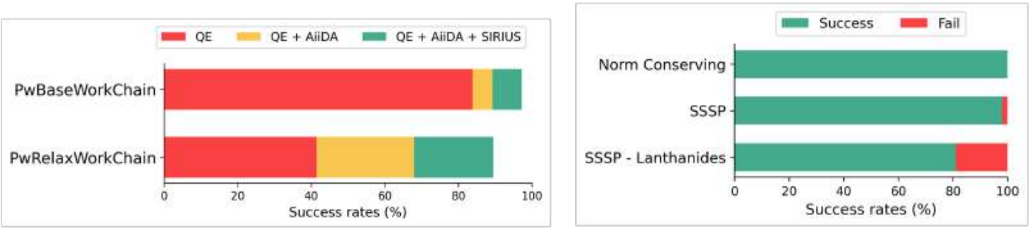
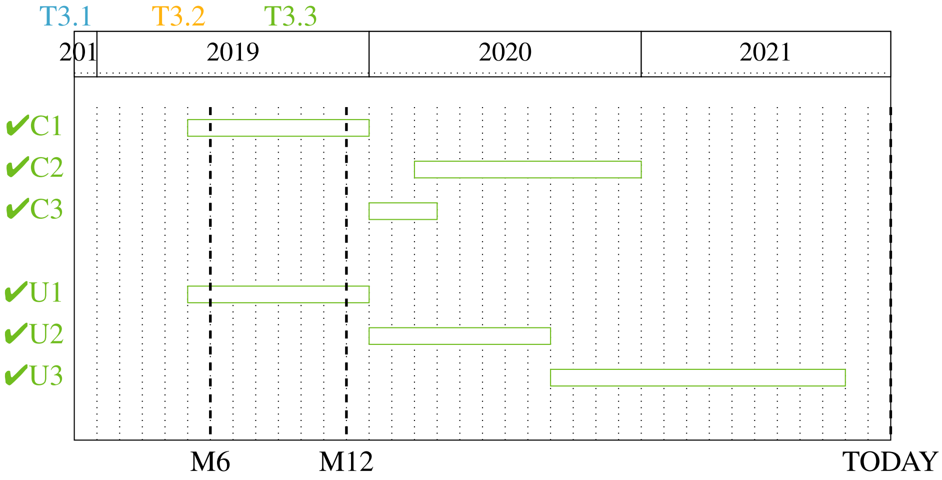


Figure 11: Left: improvement in convergence of 20’000+ materials using the QE+AiiDA+SIRIUS/NLCG approach. Right: The success rate of convergence for pure norm-conserving and SSSP pseudo-potentials (mixed NC/USPP/PAW collection). The ultrasoft and PAW pseudo-potentials for Lanthanides need further investigation as they lead to most convergence failures.


T3.1
T3.2
T3.3
201
2019
2020
2021
✓C1
✓C2
✓C3
✓U1
✓U2
✓U3
M6
M12
TODAY

4 Conclusions and overall status

In this document we have presented the last actions of algorithmic development and implementation activities of the MAX flagship codes and libraries. Such activities are deeply connected with the actions of other workpackages, in particular WP1 and WP2 for what concerns their implementation, and WP5 and WP6 for their outcome or requirement indications. After three years of project, the code actions presented here have helped in improving the performances and capabilities of the MAX flagship codes, and have contributed to pave the way towards an efficient exploitation of the novel architectures.

The foreseen objectives of the workpackage have been met (all the Key Performance Indicators reached their expected results already at the project mid-term). In particular, for each flagship code, several new algorithmic approaches have been implemented, including algorithms for robustness and resilience, algorithms aimed at exploiting exascale machines to compute properties that were previously considered out-of-reach, and algorithms addressing new physical properties or usage models.

The pandemic emergency brought some delay for a few actions, but also new ideas and new actions in the context of this WP. Rather, the code developers had the possibility to have a more focused period working on the improvement of their development approach and strategy (tools and programming models, repositories, development meetings, team communication, to name a few).

Acronyms

MBPT Many Body Perturbation Theory. 17

SCDM Selected Columns of the Density Matrix [5]. 17

SCF Self Consistent Field. 17

SVD Singular Value Decomposition. 17

References

[1] Genovese, L. *et al.* Report on identified algorithmic advances, and their software development plan. Deliverable D3.1 of the H2020 CoE MaX (final version as of 31/05/2019; revised version Dic 2020). EC grant agreement no: 824143, CEA, France. (2019).

[2] Genovese, L. *et al.* First release of MAX software: report on the identified actions, update of the software development plan, and software release. Deliverable D3.2 of the H2020 CoE MaX (final version as of 30/11/2019). EC grant agreement no: 824143, CEA, France. (2019).

[3] Genovese, L. *et al.* Second release of MaX software: Report on the evolution actions taken in each of the codes, and software release. Deliverable D3.3 of the H2020 CoE MaX (final version as of 30/11/2020). EC grant agreement no: 824143, CEA, France. (2020).

[4] Affinito, F. *et al.* Final report on code profiling and bottleneck identification. deliverable d4.5 of the h2020 project max (final version as of 30/11/2021). ec grant agreement no: 824143, cineca, casalecchio di reno (bo), italy. (2021).

[5] Damle, A., Lin, L. & Ying, L. Compressed representation of kohn–sham orbitals via selected columns of the density matrix. *Journal of Chemical Theory and Computation* **11**, 1463–1469 (2015).

[6] Prendergast, D. & Louie, S. G. Bloch-state-based interpolation: An efficient generalization of the shirley approach to interpolating electronic structure. *Phys. Rev. B* **80**, 235126 (2009).

[7] Shirley, E. L. Optimal basis sets for detailed brillouin-zone integrations. *Phys. Rev. B* **54**, 16464–16469 (1996).

[8] Pickett, W. E., Krakauer, H. & Allen, P. B. Smooth fourier interpolation of periodic functions. *Phys. Rev. B* **38**, 2721–2726 (1988).

[9] Shankland, D. G. In Marcus, P., Janak, J. & Williams, A. (eds.) *Computational Methods in Band Theory*, 362 ((Plenum, New York, 1971).

[10] Leon, D. A. *et al.* Frequency dependence in gw made simple using a multipole approximation. *Phys. Rev. B* **104** (2021).

[11] Alliati, I. M., Sangalli, D. & Grüning, M. Double k-grid method for solving the bethe-salpeter equation via lanczos approaches. *Front. Chem.* **9** (2022).

[12] Sivkov, I., Seewald, P., Lazzaro, A. & Hutter, J. Dbcsr: A blocked sparse tensor algebra library. *arXiv* (2019).