



## D6.1

# Report on the deployment of the MaX Demonstrators and feedback to WP1-5

Pablo Ordejón, Uliana Alekseeva, Stefano Baroni, Riccardo Bertossa,  
Miki Bonacci, Pietro Bonfà, Claudia Cardoso, Carlo Cavazzoni,  
Vladimir Dikan, Stefano de Gironcoli, Andrea Ferretti, Alberto García,  
Luigi Genovese, Federico Grasselli, Anton Kozhevnikov, Deborah  
Prezzi, Davide Sangalli, Joost VandeVondele, Daniele Varsano, Daniel  
Wortmann

Due date of deliverable: 31/05/2020  
Actual submission date: 31/05/2020  
Final version: 31/05/2020

Lead beneficiary: ICN2 (participant number 3)  
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) / 30/11/2021 (month 36)                                 |
| Website:                      | <a href="http://www.max-centre.eu">www.max-centre.eu</a>                     |
| Deliverable No.:              | D6.1   |

**Authors:** P. Ordejón, U. Alekseeva, S. Baroni, R. Bertossa, M. Bonacci, P. Bonfà, C. Cardoso, C. Cavazzoni, V. Dikan, S. de Gironcoli, A. Ferretti, A. García, L. Genovese, F. Grasselli, A. Kozhevnikov, D. Prezzi, D. Sangalli, J. VandeVondele, D. Varsano, D. Wortmann

**To be cited as:** Ordejón, et al., (2020): Report on the deployment of the MaX Demonstrators and feedback to WP1-5. Deliverable D6.1 of the H2020 project MaX (final version as of 31/05/2020). EC grant agreement no: 824143, ICN2, Barcelona, Spain.

## 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.



## D6.1 Report on the deployment of the MaX Demonstrators and feedback to WP1-5

### Content

|   |           |
|---|-----------|
| <b>1 Executive Summary</b>  | <b>4</b>  |
| <b>2 Introduction</b>   | <b>5</b>  |
| <b>3 Deployment of the MaX Demonstrators</b>  | <b>6</b>  |
| 3.1 “On-the-fly” evaluation of complex materials properties with Molecular Dynamics   | 6         |
| 3.1.1 Ab-Initio Molecular Dynamics simulations of a COVID-19 protein with Quantum ESPRESSO and SIESTA                         | 6         |
| 3.1.2 Quasi-particle energies and optical absorption of 2D interfaces by many-body perturbation theory.                       | 11        |
| 3.1.3 High throughput calculation of the GW100 dataset using Yambo, Quantum ESPRESSO, and AiiDA on a GPU-accelerated machine. | 14        |
| 3.1.4 Computation of thermal transport properties with Quantum ESPRESSO + SIESTA + AiiDA                                      | 15        |
| 3.2 Reduction of the complexity in very large scale systems   | 20        |
| 3.3 Full DFT simulation of new particle-like objects in chiral magnets  | 24        |
| 3.3.1 Skyrmions in 2D topological non-trivial insulators  | 24        |
| 3.3.2 Bloch points in 3D materials  | 25        |
| 3.4 RPA and double hybrid based MD simulations of condensed phase systems   | 26        |
| 3.5 Coupled electrons and nuclear dynamics within NEQ–MBPT in technologically relevant materials.                             | 27        |
| <b>4 Conclusions and perspectives</b>   | <b>30</b> |



## 1 Executive Summary

One critical goal of MAX is to enable computation of materials properties on current and future (pre)-exascale machines. For that purpose, several technical work-packages (WP1 to WP4) are focused on making the MaX Flagship codes be able to run efficiently in large supercomputers, while WP5 provides tools to reach exascale via high-throughput calculations through the AiiDA platform. In turn, WP6 intends to integrate all these advances to test and demonstrate the deployment of the codes and workflows to tackle specific, realistic applications in the framework of HPC infrastructures available during MAX, and to prepare for future pre-exascale and exascale environments. Also, to show, by means of specific demonstration cases, examples of new science that will be enabled by the availability of pre-exascale and future exascale environments and by the codes and workflows able to exploit them. For that purpose, five specific demonstration projects were proposed, and this Deliverable reports on the advance of each of them:

1. “On-the-fly” evaluation of complex materials properties with Molecular Dynamics.  
We report on the advances made in integrating codes to run in a “producer-consumer” setup, in which one code generates certain data (like snapshots of a molecular-dynamics simulation) and the other digests those “on-the-fly” and computes some properties of interest. Work in this line has focused on three codes: Quantum ESPRESSO, SIESTA and Yambo. We first demonstrate the performance of these codes to tackle demanding problems in HPC environments, like the simulation of an important protein related to the SARS CoV-2 virus with Quantum ESPRESSO and SIESTA, or in obtaining the quasiparticle energies and optical excitations of 2D interfaces with Quantum ESPRESSO and Yambo. We also explore the integration of Quantum ESPRESSO, Yambo and AiiDA for high-throughput calculations on GPU-accelerated machines. Finally, we give an account of the progress made in integrating “on-the-fly” some of the codes (specifically Quantum ESPRESSO and SIESTA) to tackle the specific and important scientific problem of computing the thermal properties of materials from first-principles, discussing several possible schemes for the implementation that are being explored.
2. Reduction of the complexity in very large scale systems  
The linear scaling capabilities of the BigDFT code have been deployed to study several very-large-scale systems during this initial period. This report describes the scheme developed to decompose a given physical system (exemplified here by a protein interacting with a toxin) into coarse grained fragments. This allows to simplify the information obtained from the complex electronic structure, providing physical and chemical insight into the interaction mechanisms of the protein and the toxin. The calculations enable the possibility of high-throughput screening in large HPC infrastructures.
3. Full DFT simulation of new particle-like objects in chiral magnets.  
FLEUR is being used to perform fully ab-initio calculations of several complex magnetic structures in materials. The size (in terms of the number of atoms involved) of these structures poses a huge challenge to DFT calculations, which traditionally have been used only to obtain effective parameters for simplified and less computationally intensive



models that could describe these structures. Here, we show that FLEUR is able to produce simulations of such structures (in particular, skyrmions in 2D topological non-trivial insulators and Bloch points in 3D materials), with calculations involving hundreds to thousands of atoms in SuperMUC-NG.

4. RPA and double hybrid based MD simulations of condensed phase systems.  
Precise calculations of chemical systems in aqueous solution (mainly the alkali cation series) require very accurate methods, beyond the semilocal DFT functionals typically employed (GGA and related). The Random Phase Approximation (RPA) is a very accurate method overcoming these limitations, but at a very high computational cost. We describe the advances done using the RPA implementation available in CP2K, and show that the wrapper developed for the COSMA library (a parallel, high-performance, GPU-accelerated matrix-matrix comultiplication algorithm) allows for RPA calculations in systems with 128 water molecules reaching a very high performance, with very good perspectives of applicability in the coming pre-exascale systems.
5. Coupled electron-phonon dynamics within NEQ–MBPT.  
We have advanced in the two ingredients needed to implement the coupled electron and ion dynamics within Non-Equilibrium Many-Body Perturbation Theory. First, we have released the real-time module of the Yambo code, with a redesigned and optimized parallelization scheme and introducing parallel I/O. Second, we have improved the interoperability of Yambo with Quantum ESPRESSO, from which the Kohn-Sham orbitals are obtained. Communication between the two codes to update the KS states will be implemented “on-the-fly” through python scripts.

The main feedback provided to WP1-WP5 is: (i) the identification of the widespread need for effective solutions for large distributed linear algebra problems in accelerated systems; (ii) the growing weight of the computational power provided by accelerators in the pre-exascale HPC infrastructures and the need to accurately distribute the data structures over the most powerful computing elements; (iii) addressing and making proper use of the direct communication channels between accelerators featured by most GPU-accelerated clusters, through vendor-specific APIs or external libraries like OpenMPI; (iv) the need for further integration of codes with AiiDA not only for high-throughput but also to interface the interoperability between codes sharing data and intermediate results.

## 2 Introduction

This Deliverable presents the advances made in the deployment of these demonstrators in the HPC infrastructures currently available, with an emphasis on the prospects for the future pre-exascale and exascale infrastructures to come. Importantly, we were already able to run some demonstrators on new generation machines with heterogeneous architectures including accelerators like GPUs (such as Cineca’s Marconi100), thus providing strong evidence of the suitability of the codes to the (likely) architectures of the coming pre-exascale European supercomputers.



### 3 Deployment of the MaX Demonstrators

#### 3.1 “On-the-fly” evaluation of complex materials properties with Molecular Dynamics

A first class of MaX demonstrators is based on using two codes in a producer-consumer setup: one code would generate certain data (for example, snapshots of a molecular-dynamics simulation), and the other code would digest those “on-the-fly” and compute some property of interest.

As a preliminary step to the evaluation of complex property “on the fly”, we have first addressed the demonstration of the capabilities of the individual molecular dynamics engines involved (Quantum ESPRESSO and SIESTA) in Section 3.1.1, and the excited state simulations (Yambo) in Section 3.1.2. In Section 3.1.3, we demonstrate the combination of AiiDA + Quantum ESPRESSO + Yambo on a GPU-accelerated machine in a case where the combination of dynamics and on-the-fly calculations requires an automation tool. Finally, Section 3.1.4 addresses the “on-the-flight” calculation of thermal properties in materials combining several capabilities of Quantum ESPRESSO and SIESTA, with the aid of AiiDA.

##### *3.1.1 Ab-Initio Molecular Dynamics simulations of a COVID-19 protein with Quantum ESPRESSO and SIESTA*

Research efforts are pouring into understanding the structure and biological mechanisms of the virus causing COVID-19 (a pandemic representing a major crisis worldwide). Simulations are playing a key role in the development of drugs to fight the pandemic, and ab-initio calculations may be very relevant in this effort. However, with the current generation of supercomputers, a realistic Ab-Initio Molecular Dynamics simulation of a protein, even a relatively small one such as the SARS CoV-2 main protease M<sup>pro</sup> (an attractive target for the development of antiviral drugs against COVID-19) is an enormous challenge for DFT, due to the very large number of atoms involved (protein plus water and counterions, which add up to several tens of thousands of atoms, at least) and to the very long runs required to reach the relevant time scales. Nevertheless, it is of fundamental importance to prove that such calculations are possible, by setting up and actually running and benchmarking these simulations for a small number of steps on current HPC infrastructures. The objective is to ensure that the codes do not contain fundamental memory or computational bottlenecks, preventing the execution of these huge systems. Indeed, regardless of the size of the supercomputer available, if the codes contain data structures or operations with bad scalability properties, the code will not run on larger supercomputers either. When pushing the codes and the supercomputers to the limit of their possibilities, we discover code limitations that never showed-up (e.g.: the use of 32 bit integers for indirect vector indexing or global vector indexing, since many global vectors will have more than 2 billion elements - the limit for signed integers). We can also discover limits or bugs in the supercomputer as well, supporting and feeding the HW-SW co-design activity. Finally, proving that a given huge simulation is already possible on today’s (multi-petaflop) supercomputers is a necessary condition (for the code) to run on next generation pre-exascale and exascale machines, when these huge simulations could become the norm for, still large, DFT simulations.



Deliverable D6.1

Report on the deployment of the MaX Demonstrators and feedback to WP1-5

Here we show benchmarks for the calculation of a monomer of the main protease ( $M^{pro}$ ) of the SARS CoV-2 virus which constitutes the minimal unit from which realistic models of biological relevance can be built (the protease is formed by two monomers docked forming a dimer).  $M^{pro}$  is a protein which is key to the replication of the virus inside the infected cell, in processing the polyproteins that are translated from the viral RNA. It is therefore the target for the design of chemical compounds that can inhibit its activity, therefore becoming potential drugs to treat the disease. Several groups have published the atomic structure of the protein, obtained from x-ray diffraction studies at synchrotron facilities in crystalline samples. These models are being used by many groups worldwide as starting points for the computational design of inhibitors. We have used the coordinates provided by Fearon et al. (PDBid: 5RTY, <https://www.ebi.ac.uk/pdbe/entry/pdb/5r7y>) shown in Fig. 1. We describe here the benchmarks done for two of the MaX flagship codes: Quantum ESPRESSO and SIESTA, which are the ones used in the rest of the work on “on-the-fly” approaches for molecular dynamics simulations. Further work on the same system is being done with the BigDFT code, as described in MaX deliverable 6.2.

### Quantum ESPRESSO:

We have used the raw geometry of the protein taken from the PDB (no hydrogens, single monomer), and surrounded it by 2140 water molecules, adding up to a total of 8783 atoms (see Fig. 2). The calculations are done using Car-Parrinello molecular dynamics with the cp.x executable of Quantum ESPRESSO. The system is extraordinarily large for CP molecular dynamics simulations, as these calculations are based on DFT and plane waves, each one extending on the whole domain. Table 1 reports a few key parameters of the computation. Inputs and results of ten step simulation are available in the MaX benchmark git repository: [https://gitlab.com/max-centre/benchmarks/-/tree/master/Quantum Espresso/CP/5R7Y-COVID19](https://gitlab.com/max-centre/benchmarks/-/tree/master/Quantum_Espresso/CP/5R7Y-COVID19).

With the available resources we were able to run a few CP-MD steps, enough to demonstrate the feasibility of this kind of simulation. In fact, each step, computationally, is identical to each other, so this demo run is enough to demonstrate the possibility to run a full CP-MD simulation of many thousands of steps (picosecond). This seems a short time as compared with classical MD, and indeed it is, but when using CP simulations we are (at least in the near future) not aiming at studying thermodynamics properties of the molecule, but rather chemical properties, especially if in the protein there are metal atoms, which are notoriously difficult to be accurately described with classical MD. For example, in the specific case of SARS CoV-2  $M^{pro}$  the interaction between potential inhibitors and the active site cavity can be assessed, which is a main critical point for the screening and development of antiviral drugs.

|                               |                      |
|-------------------------------|----------------------|
| Number of Electrons           | 27910                |
| Number of Atoms               | 8783                 |
| Number of Plane waves         | 56150975             |
| FFT Grid sizes                | 360 576 540          |
| Box size (Bohr <sup>3</sup> ) | 88.8 x 139.8 x 132.3 |

Table 1: parameters for the SARS CoV-2  $M^{pro}$  Car-Parrinello calculations with Quantum ESPRESSO

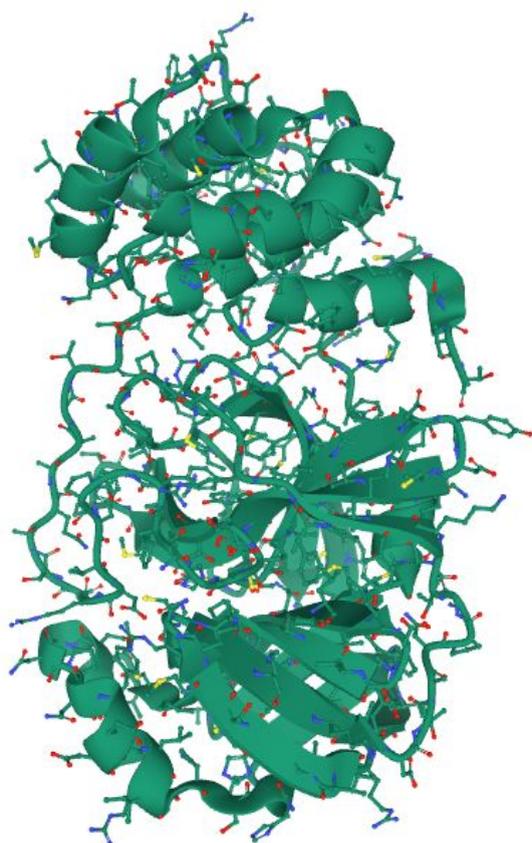
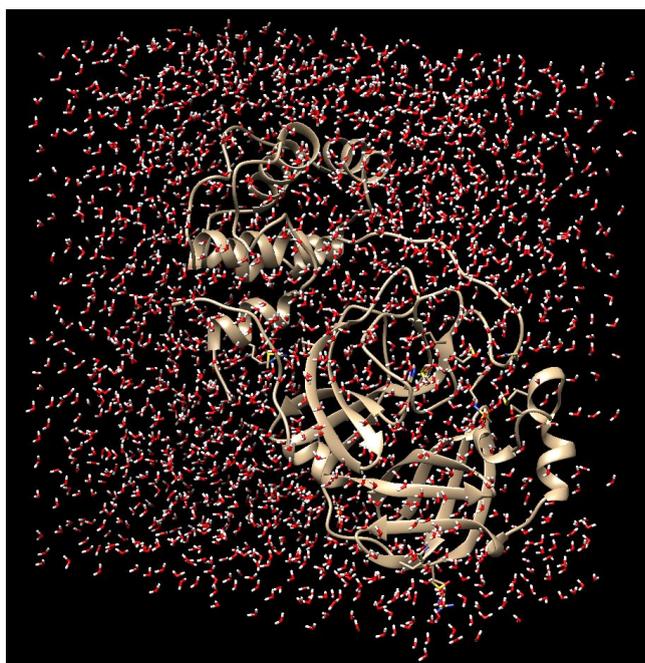


Figure 1. Experimental structure of the SARS CoV-2 M<sup>pro</sup> (Fearon et al., PDBid: 5RTY). The representation indicates the positions of the atoms, superimposed to a cartoon representation of the secondary protein structures. Note that experimental coordinates do not contain the hydrogen atoms, which can not be detected by x-ray diffraction. In biological function, two of these monomers assembly in a dimer structure, within a water and counterions environment.

Figure 2: The SARS CoV-2 M<sup>pro</sup> 5R7Y with solvation water molecules used for a Car-Parrinello molecular dynamics simulation with Quantum ESPRESSO.



## SIESTA:

The SIESTA code has the advantage of the low cardinality of the basis set compared with other approaches like plane waves: as it is based on atomic orbitals, the number of basis functions to describe accurate molecular and solid state systems is quite small. It also allows for less accurate, yet physically reasonable, calculations using very small basis sets (called “minimal bases”, where only the atomic orbitals occupied in the free atom are used in the basis set) thus minimising the computational workload to the extreme. These calculations still retain the advantages of being a fully quantum-mechanical first-principles approach, thus providing physical and chemical information, without the need to resort to parameterized force fields.

The large size of the systems needed to simulate the SARS CoV-2 M<sup>pro</sup> provides an ideal testing ground for SIESTA. We have first used a minimal basis (single-zeta -SZ-, composed of one “s” orbital for H, and one “s” and three “p” orbitals for C, N, O and S). This small basis allows us to tackle a realistic model for the SARS CoV-2 M<sup>pro</sup> monomer, now including all the H atoms missing from the experimental structures (using the utility provided in <http://molprobit.biochem.duke.edu/>). Calculations in the dry and wet forms of the protein (i.e., with no water or with the protein embedded in water) have been performed. The parameters of the calculation are shown in Table 2.

|                              |           | Dry M <sup>pro</sup> | M <sup>pro</sup> in water |
|------------------------------|-----------|----------------------|---------------------------|
| Number of Electrons          |           | 13020                | 32588                     |
| Number of Atoms              |           | 4649                 | 11987                     |
| Number of orbitals           | SZ basis  | 11693                | 26369                     |
|                              | DZP basis | 42029                | 98287                     |
| Grid sizes                   |           | 300 x 600 x 384      | 300 x 600 x 384           |
| Box size (Ang <sup>3</sup> ) |           | 38.0 x 77.0 x 48.0   | 38.0 x 77.0 x 48.0        |

Table 2: parameters for the SARS CoV-2 M<sup>pro</sup> simulations with SIESTA. SZ and DZP bases refer to “single-zeta” and “double zeta with polarization” bases, respectively.

The calculations were done in the MareNostrum 4 supercomputer at BSC in Barcelona, which has 3.456 nodes. Each node has two Intel Xeon Platinum chips, each with 24 processors, amounting to a total of 165,888 processors. For the SZ calculations, we used from ½ to 24 nodes (24 to 1152 processors). The wall time for the execution of one SCF step is shown in Fig. 3. For this system and basis set size, we observe good scaling up to a few hundred processors, and a total execution time that allows for several tens of molecular dynamics steps per day. This, for the first time, permits the realisation of structural relaxations and short (picosecond) molecular dynamics simulations in these biological systems. As an example, we were able to perform a preliminary relaxation of the initial coordinates, until reaching a maximum force tolerance of 2 eV/Ang. We note that the initial forces were very large (some of them, around 30 eV/Ang), as the protein coordinates from experimental x-ray diffraction have a relatively large error, the H atoms are placed using

heuristic rules, and the water molecules are located ad-hoc, without any optimisation. Fig. 4 shows the evolution of the conjugate gradients relaxation for the case of the dry  $M^{pro}$ , indicating the value of the largest atomic force as a function of the relaxation step. The whole relaxation (up to a tolerance of 2 eV/Å) took about 7 hours in 2 nodes (96 processors) in MareNostrum 4.

Figure 3: Execution times (per SCF step) for the SIESTA calculations of the SARS CoV-2  $M^{pro}$  in water (11987 atoms) with a minimal single-zeta basis set (26369 orbitals). Timings obtained in MareNostrum 4, using two different interfaces to ELPA from the most recent MaX release of SIESTA. The MaX-1.0-15 case (implementing the ELSI interface to the ELPA massively parallel dense eigensolver) provides superior performance. The number of MPI tasks was equal to the number of processors used.

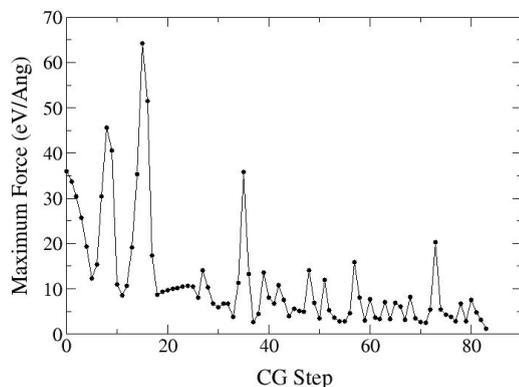
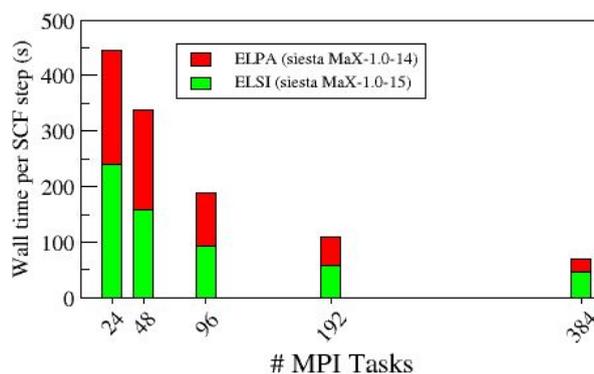


Figure 4: Evolution of the maximum atomic force in a SIESTA Conjugate Gradients (CG) relaxation of the dry SARS CoV-2  $M^{pro}$  (4649 atoms) with a minimal single-zeta basis set (11693 orbitals). The relaxation was stopped when the maximum force was smaller than 2 eV/Å.

While the SZ basis provides a first physical and chemical insight into the properties of materials, it is certainly not a sufficiently accurate basis for quantitative calculations. In SIESTA, very accurate results (comparable to standard plane wave results) are usually obtained using larger basis sets, and in particular, the so-called double-zeta plus polarization (DZP) basis. This contains approximately three times more orbitals per atom than the SZ, but provides a much higher accuracy. To test the feasibility of this DZP basis for these biological systems, we have produced benchmarks for the model of SARS CoV-2  $M^{pro}$  in water (11987 atoms). In this case, the number of basis orbitals (98287) is more than three times larger than for the SZ basis. The implications of this fact are twofold. First, the memory footprint of the code is significantly larger. We found that a minimum of 432 processors (each with a memory of 2Gb) was necessary to fit the run into memory. This is due to the quadratic scaling of the memory footprint of SIESTA (in its diagonalization mode) with respect to the number of basis orbitals, which implies a 10-fold memory increase going from the SZ to the

DZP basis. In turn, the cubic scaling of the CPU time for diagonalization implies an even larger increase in the CPU consumption. Fig. 5 shows the scaling versus number of MPI processes, showing good scaling up to some 2000 processors (significantly higher than that for the smaller SZ basis). Still, using this large number of processors, the wall time is still not prohibitive for relaxations and short molecular dynamics simulation runs in such a complex system and with an accurate basis set.

SIESTA is therefore very well suited for calculations in these complex systems, in the present CPU-based supercomputers. The advances made in the past few months in the porting of SIESTA to hybrid architectures, and in particular, adding GPU support (as described in MaX D4.3), which is already available in released versions of the code (since 4.1, using a direct interface to ELPA, and also with the ELSI interface) extends this capabilities to a larger range of HPC infrastructures, projecting towards the next generation of pre- and exascale facilities. Work in Marconi100 is already underway in this direction.

Even larger systems can certainly be tackled with SIESTA using the PEXSI solver (see the discussion on its superior scalability properties in D4.3), and the linear scaling solvers being developed and optimized in MaX, in collaboration with the BigDFT team. These will be deployed in the coming months on the largest computational infrastructures available to us.

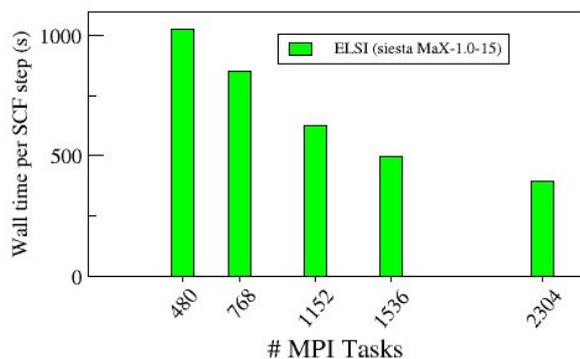


Figure 5: Execution times (per SCF step) for the SIESTA calculations of the SARS CoV-2  $M^{\text{pro}}$  in water (11987 atoms) with a DZP basis (26369 orbitals). Timings are obtained in MareNostrum 4 with the SIESTA MaX-1.0-15 release. The number of MPI tasks was equal to the number of processors used.

### 3.1.2 Quasi-particle energies and optical absorption of 2D interfaces by many-body perturbation theory.

We have started by demonstrating separately the feasibility of running large scale systems using Quantum ESPRESSO and Yambo.

For the demonstrator concerning Yambo, we have chosen to compute the quasi-particle corrections and adsorption properties of a prototypical 2D interface. This system is taken here as an example of systems of interest in the study of photocatalysis as well as for the research of new batteries, addressed in D6.2.

Our main goal is the investigation of edge- and quantum-confinement-effects for realistic graphene nanoribbons (GNR) as altered by the coupling to a substrate. In fact, there is a continuous effort in the scientific community for the characterization of GNRs adsorbed on



weakly (and less weakly) interacting surfaces. Moreover, the coupling of quantum objects with metallic substrates can lead to a wealth of interesting phenomena (renormalization of the quasi-particle energies, coupling of excitations, charge and energy transfer, to name a few).

For the case of GNR@Au(111), it was possible, by applying an image-charge model potential, to predict the strong energy-gap renormalization induced by substrate polarization effects [P. Ruffieux, et al. ACS nano, 6, 6930 (2012)], and to suggest an almost negligible influence of the substrate on the optical excitation energy [R. Denk, et al, Nature Comm. 5, 4253 (2014)]. However, in order to better understand the effects of GNR-substrate interaction and to account for polarization effects, it is necessary to perform calculations on the whole system, going beyond the application of semiempirical models. This does not only allow for benchmarking the applicability of simplified models (e.g. image-charge potential), but it also provides a definitive understanding of some so-far open issues, such as the role of the substrate in determining: (1) band-dispersion renormalization, and (2) distinction between polarization and hybridization effects, (3) the nature and features of the excitations. This direct approach was applied to a GNR@graphene, a large target system.

Using ab initio methods in the realm of many-body perturbation theory, we have computed the electronic structure, including the fundamental gap, at the GW-PPA level. These results will also be used as the starting point for the calculation of the optical absorption of GNR in the presence of a substrate, by solving the Bethe-Salpeter Equation (BSE) as implemented in Yambo.

Yambo is GPU-enabled and can exploit the newly deployed Marconi100 GPU-accelerated supercomputer (Cineca, Bologna), which consists of 980 nodes equipped with 2 IBM P9 cpus and 4 NVIDIA V100 GPUs (Nvlink 2.0, 16GB), each. We tested the scalability of the parallelization of Yambo up to 600 nodes, i.e. 2400 GPU cards (about 20 PetaFlops single run). The tests were performed on Marconi100 using 4 MPI tasks per node, 32 threads per task and a 1:1 binding between MPI tasks and GPUs. In a second set of tests, shown in Fig. 6, we improved the core/MPI/thread binding, further enhancing scalability. The new tests were performed using up to 250 nodes, i.e. 1000 GPU cards, about 8 PetaFlops single run (showing a performance  $\geq 50\%$ ).

Yambo has shown a very good performance up to a very large number of nodes. As can be seen from the Fig. 6, the scaling of the main computational kernels (Dipoles,  $\chi_0$ ,  $\Sigma_x$ ,  $\Sigma_c$ ) is quite good, being the calculation of the irreducible polarisability,  $\chi_0$ , the most expensive computational task, followed, at a large distance, by the calculation of the dipoles.

For GPU-accelerated systems the recommended usage model for our software is to use one MPI task per card, setting the number of OpenMP threads to the maximum available on the host in order to best exploit its computational capabilities. Since currently available GPUs have a quite large computational power per card, with a few MPI tasks one is already able to address and exploit a very large computational partition. Given the very effective scalability of Yambo, especially in the limit of small-medium number of tasks, the main issue hindering an efficient use of Yambo on GPU-accelerated machines is the memory footprint. Current GPUs typically have 16 to 32 GB RAM per card, while Yambo can be very memory hungry

(especially if the memory is distributed only over a few MPI tasks). In this respect, testing the GPU porting of the code against large systems is of fundamental importance. The present tests have shown the feasibility of running the Yambo GPU-enabled version to address the study of the electronic and optical properties of large 2D interfaces.

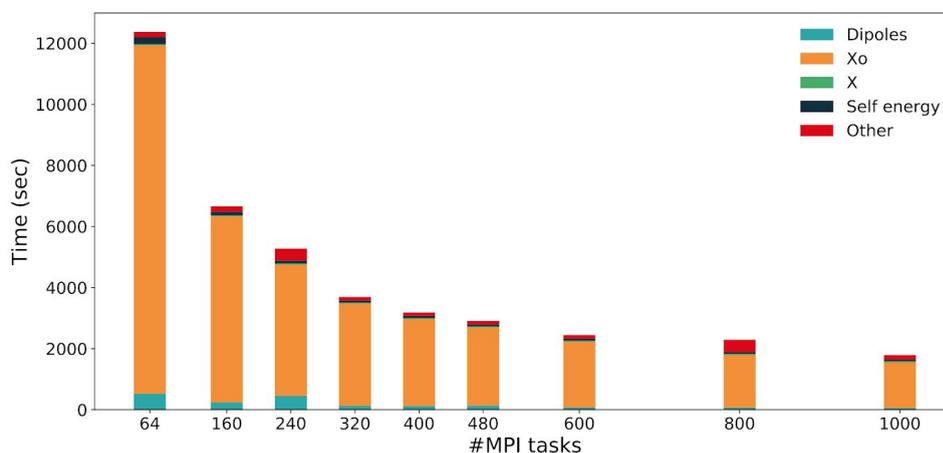


Figure 6: Execution times for the tests performed with the Yambo code to verify the efficiency of the MPI parallelisation. One MPI task per V100 GPU was used; 1000 MPIs corresponding to about a 8 PFlop partition. The time for each of the main tasks of the code is given separately. The total time taken to perform other tasks is labeled as "Other". Timing data is explicitly reported in Table 3.

| #MPI | #threads | Dipoles | $\chi^0$ | $\chi$ | $\Sigma x$ | $\Sigma c$ | WALL_TIME |
|------|----------|---------|----------|--------|------------|------------|-----------|
| 64   | 32       | 519     | 11419    | 32     | 6          | 225        | 12370     |
| 80   | 32       | 433     | 9685     | 31     | 4          | 184        | 10506     |
| 160  | 32       | 240     | 6092     | 33     | 3          | 109        | 6660      |
| 240  | 32       | 453     | 4303     | 34     | 1          | 89         | 5276      |
| 320  | 32       | 129     | 3346     | 32     | 1          | 69         | 3694      |
| 400  | 32       | 118     | 2848     | 35     | 1          | 69         | 3182      |
| 480  | 32       | 132     | 2562     | 37     | 1          | 59         | 2906      |
| 600  | 32       | 73      | 2154     | 35     | 1          | 60         | 2447      |
| 800  | 32       | 67      | 1722     | 39     | 0          | 61         | 2292      |
| 1000 | 32       | 58      | 1494     | 35     | 0          | 56         | 1792      |

Table 3: Execution times for the tests performed with the Yambo code to verify the efficiency of the MPI parallelisation. The tests were performed on Marconi100 using 4 MPI tasks per node, 32 threads per task and a 1:1 binding between MPI tasks and GPUs. The time for each of the main tasks of the code is given separately, in seconds.



### *3.1.3 High throughput calculation of the GW100 dataset using Yambo, Quantum ESPRESSO, and AiiDA on a GPU-accelerated machine.*

The study of ionization potential and electronic affinity for 100 molecules, by means of the GW approximation (in the  $G_0W_0$ [PBE] flavour), represents a large scale study that requires the combined use of high-throughput techniques and high performance computing. Both the ground state properties and the quasiparticle steps of the calculations are efficiently managed within the AiiDA framework through automated workflows, as implemented in the `aiida-yambo` plugin (YamboConvergence is the main workflow). This approach allows one to perform hundreds of calculations in a high-throughput fashion: a first step towards turn-key solutions for many-body-perturbation-theory studies.

Both codes, i.e. Quantum ESPRESSO (for the ground state - DFT part) and Yambo (for the quasiparticles - GW part), are GPU-enabled and were able to exploit the Marconi100 (M100) GPU-accelerated supercomputer (Cineca), which consists of 980 nodes equipped with 2 IBM P9 CPUs and 4 NVIDIA Volta V100 cards (Nvlink 2.0, 16GB) each. After porting of the code and their deployment on the M100 machine, all the DFT+GW calculations needed for a single molecule could run in less than one hour. Indeed, the evaluation of quasiparticle corrections at the GW level is performed by extrapolating on 10 to 16 results obtained for different values of the parameters involved in calculations, where a single GW run requires from 2 to 8 computing nodes (8 to 32 V100 GPUs) and lasts less than 5 minutes.

Considering the large number of calculations to be performed (about 2000), the high-throughput, fully automated implementation was fundamental, and allowed us to cover the full dataset in a week. Moreover, this extensive study allowed us to highlight the main bottlenecks in the calculations and to plan for the next implementation actions. In fact, the major computational effort is represented by the calculation of the screening dielectric matrix, with the matrix size determining the resources to be used. To obtain very accurate results, it is necessary to compute large screening matrices (energy cutoff  $> 8$  Ry, matrix size  $> 25000 \times 25000$ ) for rather large unit cells ( $40 \times 40 \times 40$  Bohr<sup>3</sup> box). Given the size of the matrices and the large number of wavefunctions to deal with, the main bottleneck of the calculations turned out to be the memory usage on the GPUs. In turn, this can be traced back to the lack of parallel distributed linear algebra support on GPUs. It is important to note that this is a general issue with more or less all MaX codes (and also beyond the MaX community itself). Therefore, the development of a parallel distributed linear algebra library for dense matrices on GPUs (pretty much what Scalapack is for CPUs) is one of the crucial priorities in the field.

As soon as any library with these features is available, we plan to adopt it in Yambo (as well as in other MaX codes) as the next major step for improving the performance and code usability, paving the way to the investigation of even larger/more realistic systems with many-body perturbation theory level of accuracy.



### 3.1.4 Computation of thermal transport properties with Quantum ESPRESSO + SIESTA + AiiDA

Transport phenomena, arising from the stochastic nature of atomic motion, are irreversible processes which govern the transfer or storage of heat, charge and mass. Investigating the transport properties of materials is thus of paramount importance in many branches of science and technology, ranging from the design of energy-related devices (batteries, fuel cells, thermoelectric materials, thermal insulators, heat sinks), to the modelling of planetary evolution, where the experimental access to materials properties is, to a large extent, impossible. Computer simulations may thus be the only tool to investigate transport properties (heat and electrical conductivity, viscosity, etc.) of materials at experimentally inaccessible conditions (like extreme pressures and temperatures).

The specific scheme for the calculation of the thermal transport coefficient from DFT is based on a recent breakthrough by the SISSA group [Marcolongo, Umari & Baroni, *Nature Phys* 12, 80–84 (2016)]. It was assumed until recently that the lack of a univocal definition of the electronic energy density in DFT would preclude the use of the Green-Kubo formalism for the calculation of thermal fluxes and their correlation functions, needed for thermal transport. It has been demonstrated, however, that, subject to very general conditions of extensivity of the energy partition, the actual transport coefficients are independent of the detailed form used for the energy density. This opens the door to the calculation of thermal transport from first-principles without the need to set up ad-hoc molecular-dynamics simulations with imposed temperature profiles, or to account explicitly for complicated phonon-phonon interactions. The practical scheme for computing the thermal fluxes from DFT was implemented in the Quantum ESPRESSO (QE) code. Some of the terms needed can reuse parts of the code originally implemented to perform Density-Functional Perturbation Theory (DFPT) calculations, of which QE was a pioneer.

In practice, the computation of the transport properties involves two distinct tasks. The first one is to generate a long Ab-initio Molecular Dynamics (AIMD trajectory). This can be done with a variety of DFT codes (Quantum ESPRESSO, SIESTA, BigDFT, CP2K, ...), and using different approaches such as Car-Parrinello or Born-Oppenheimer. Then, the time-series of the DFT heat flux must be computed, according to the procedure described in the reference quoted above. The heat flux associated with a specific configuration depends on the electronic structure at that particular configuration and its time derivatives, the latter computed in practice from the knowledge of the atomic velocities and the perturbation induced by the associated virtual movement of the atoms. Hence, fluxes can be sampled at specific snapshots of a molecular-dynamics calculation, and the time series stored for further statistical analysis. The ab-initio heat conductivity is finally evaluated from the statistical properties of the heat flux according to the Green-Kubo theory of linear response.

The computational load involved in the procedure drafted above is very large. For instance, the calculation of the heat conductivity for a single thermodynamic state of a few hundreds of atoms currently requires approximately a month of wall time on many nodes of a modern supercomputing facility. The reason is that very long NVE trajectories are usually required (spanning at least  $\sim 100$  ps), while the computational work needed to obtain the forces from



DFT at each time step to generate these trajectories is very large. Additionally, the cost of computing the DFT heat flux for each snapshot of the AIMD run roughly doubles that of the DFT force calculation

The generation of the AIMD trajectories can, in principle, be done using any DFT code, while the calculation of the heat flux needed for the thermal properties can be done independently, using a different code. In this MaX demonstrator, we have advanced in the use of two of the MaX flagship codes (Quantum ESPRESSO and SIESTA) for the calculation of the thermal properties as described above. Here we summarize the progress made in several aspects of the project.

### Generating the AIMD trajectories

In practice, the computational procedure to generate the AIMD runs is costly and complicated, but quite repetitive, being characterised by the following procedure: (i) prepare the initial state, (ii) benchmark the integration time (and electronic fictitious mass for Car-Parrinello dynamics), (iii) equilibrate the system towards the desired thermodynamic conditions, (iv) optimize the parallelization options, and (v) finally run the NVE simulation. Thanks to the technology developed by the AiiDA team, we encoded all these operations into an AiiDA workflow for the QE code using Car-Parrinello molecular dynamics [<https://github.com/rikigigi/aiida-QEcPWorkChain>].

Alternatively, SIESTA, by virtue of the use of strictly localized basis sets, offers very competitive times-to-solution and memory footprints, even for large systems, so it is a good choice to perform the molecular-dynamics calculations needed. It should be noted that, beyond its intrinsic advantages (small cardinality of the basis set, use of sparse matrices leading to opportunities to employ reduced-scaling algorithms, etc), SIESTA's performance has been significantly enhanced by the use of accelerator-enabled solver libraries (see report in D4.3), further increasing its appeal as a simulation engine. It is, therefore, being now explored as the generator of the AIMD trajectories.

### Computing the time series of the DFT heat flux

The DFT computation of the heat flux for selected snapshots of the AIMD trajectory can already be done in QE. During this period, we revamped the heat-flux code by completely rewriting the drivers and the documentation, making it performant and user-friendly, and aligning it to the latest released distribution of QE. In this way we implemented and made available to a wider public of practitioners a state-of-the-art technique whose feasibility had been only demonstrated at an experimental level and which was not portable, as such, onto hybrid-architecture supercomputing machines: a devoted driver now calls the core subroutines (two SCF and a density functional perturbation theory calculations), and then transfers the data using the available RAM alone, rather than relying on ad-hoc bash scripts or on the filesystem for data communication.

Additionally, work is underway to compute the heat flux from SIESTA, which would have the added advantage of speed (as the formulation would be based on SIESTA's localized basis sets and provides an advantageous scaling with system size). The theoretical formulation has



already been established [*Dikan, Ordejón, García, unpublished*], and is currently being implemented in the SIESTA code. Similar implementations in other codes would allow to extend the offer of heat flux solvers for the integration in the workflows developed here.

### Integration of AIMD trajectory generation and DFT heat flux calculation

The decoupling of the generation of the AIMD trajectories and the calculation of the heat flux for the snapshots of the simulation allows for an independent calculation of both components for the determination of the thermal properties of a material. This implies that each of them can be performed in different (independent) runs, but also that different codes and approximations can be used on each of these steps. One may think of using a faster AIMD trajectory generator like SIESTA, while using the machinery developed within QE to obtain the heat flux information. As the coupling is "loose", based only on concepts common to both codes (coordinates and optionally charge density), this presents no problem, and the scheme has already been explored successfully. It should be noted, however, that SIESTA and QE can be set up to use the same (norm-conserving) pseudopotentials, and the SIESTA basis set can be tuned to match the accuracy of the QE's one, so the degree of actual physical interoperability is high.

However, this scheme also allows for several levels of integration and information exchange between the two calculations. This would greatly increase the efficiency of the calculation (in terms of reducing its computational cost), but also and most importantly, boost the potential to adapt and make full use of current pre- and future exascale infrastructures (by scheduling simultaneous calculations in a coordinated manner).

As described above, the calculation of the DFT heat flux is currently a post-processing operation where several required DFT quantities (Kohn-Sham orbitals and energies, electron charge density, etc) must be re-calculated, as their storage on disk along the AIMD is unfeasible. This represents an evident waste of computational resources and a severe efficiency bottle-neck. In a more general context, this will also be the case of other physical properties that may be computed from instantaneous wave functions or density, which require calculations not done by the standard quantum engines. Therefore, in the light of future simulations of transport (or other) properties for large-size systems, we aim to abandon the current post-processing approach in favour of on-the-fly calculations of the instantaneous properties, directly performed during the AIMD simulation.

The actual implementation of the "on-the-fly" scheme in an HPC framework offers several possibilities of increasing technical complexity and integration level. These are some of the options we are already exploring:

1. Use an MPI-based dispatch in which some processes are assigned to the AIMD trajectory generation code (SIESTA or QE), and some other pool of tasks (slaves) are assigned to the calculation of the heat flux (now QE, eventually also SIESTA) as post-processing. This is an integrated solution, using many tasks at the same time, but raises quite a number of problems of synchronization, load-balancing, and resilience in the face of node failures.



2. Have a running (and potentially restartable at will) AIMD trajectory generator (SIESTA or QE) master job spawn periodically a sub-process that will simply send a job to the queuing system to compute the heat flux (now QE, eventually also SIESTA) for a given snapshot with the appropriate data. This solves all the problems of load-balancing and synchronization, and is conceptually simple to implement. The use of individual jobs for the heat flux part, while not guaranteeing complete resilience, at least isolates and keeps any potential failures recorded for future action.
3. Use the AiiDA framework to orchestrate the calculations. This can be done in a fully integrated way, including the AIMD trajectory generation run, or partially, using AiiDA for the downstream execution of the snapshot processing only. The fail-safe procedures in the AiiDA workflows would guarantee the completion of the computation of the fluxes. In a fully integrated approach, the AiiDA workflow would also restart the MD calculation as needed (to meet wall-time limits in the supercomputer, or to fix issues of lack of convergence or outright hardware or algorithm failures).

Technical, yet important hurdles must be overcome to achieve this goal: as already remarked, to compute a wide range of instantaneous properties we need to access the wavefunction at many timesteps of the dynamics. Communicating these data via a traditional file-system approach is inefficient, due to the size of the data, and the modest speed of the reading/writing tasks. In the current route along the development of powerful HPC systems, the hard disk is getting more and more expensive with respect to the performance and availability of RAM. Hence the appeal of the RAM-based master-slave architecture, where the master process interfaces with the molecular dynamics calculation, collects the wavefunctions at the current timestep and passes them to a particular slave for the necessary calculations (see Fig. 7). This sets some challenges, like the availability of enough memory and the quest for a fast safety-buffer larger than the RAM, which has no need to be a distributed file-system, but only a local one. New technologies like Intel's Optane *persistent memory* are moving in this direction.

To overcome these challenges the collaboration with the technological Work Packages of the MaX project will be crucial. Important contributions for more integration, flexibility, improved organization of the data hierarchy, and access to emerging HPC technologies are coming from WP 1, 2 and 3 with which we are in continuous feedback. WP4 is experimenting with advanced programming models for the efficient usage of persistent non-volatile memory technologies. The access to these technologies would be a significant breakthrough that would simplify and streamline the communication and the data exchange along the execution. The AiiDA package developed in WP5 has also been crucial for the efficient organization of the workflow, further refinement of the workchain should allow to unify the workflow for Quantum ESPRESSO and SIESTA and other compatible codes.

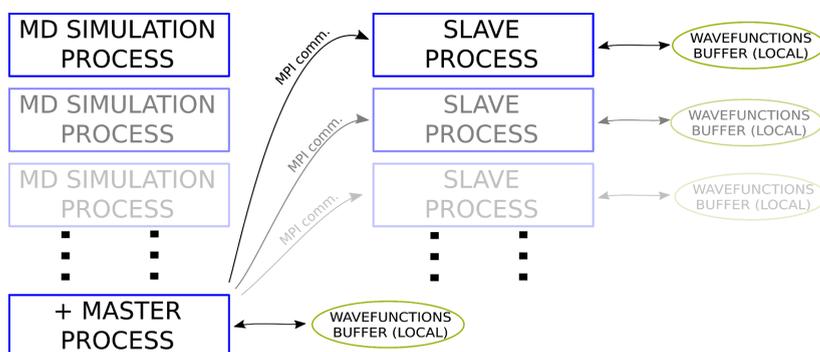


Figure 7: diagram of a possible master-slave implementation is reported in the Figure above. The master-slave paradigm, when implemented, will optimise the deployment of both computational and human time, and boost the performance and scalability of the code to compute on-the-fly DFT-accurate instantaneous properties for large-size systems ( $\sim 10,000$  electrons). The use of non-volatile memories will be constrained by their actual availability on “pre-exascale” machines of our supercomputing partner centres. Our first goal will be to realize a proof-of-concept of our master/slave paradigm with the hardware at our disposal.

### 3.2 Reduction of the complexity in very large scale systems

State-of-the-art characterization of enzymatic-substrate systems often revolves around QM/MM methods that rely on preparatory chemical observations and intuition. QM/MM has shown success in many cases, but its failure goes unnoticed and leaves us with no alternatives for more in depth investigations whenever the relevant dimensional scale exceeds a few hundred atoms. Performing a full QM characterization of enzyme-substrate interaction would allow an agnostic approach to enzymatic characterization, independent of the knowledge necessary to hypothesize proper QM regions in the model. Moreover, a full QM approach would allow validation of previous QM/MM models and enable high detail characterizations of less studied/tractable enzyme-substrate systems.

During the activities of this project, we have so far shown how BigDFT's linear scaling capabilities enable calculations on large systems. Nonetheless, these calculations remain computationally intense, and it is unrealistic to expect DFT calculations to replace commonly used forcefield methods, as a full statistical sampling of a system's configuration space (free energy calculations, molecular dynamics, etc.) remains expensive. It is thus crucial for us to develop analysis tools which use the results of large scale DFT calculations to gain new kinds of insight into the emergent properties of systems.

One such tool we have developed in this spirit is a complexity reduction framework, which takes large scale calculations, and uses them to decompose systems into coarse grained fragments. (see J. Chem. Phys. 152, 194110 (2020) for a recent publication on the topic which also includes references to our recent work).

This fragment generation procedure is based on two metrics:

- the purity indicator which measures the quality of a fragment;



- the fragment bond order which quantifies inter-fragment interactions.

Both measures can be cheaply computed directly from the single particle density matrix, and can be used to automatically partition a system into fragments, design embedding environments for QM/MM type approaches, and produce graph-like views of a system.

We have employed these methods in various production calculations in recent months. As an example, we consider the problem of describing the active site of a protein, in interaction with a substrate. Here we consider a Laccase enzyme, in contact with aflatoxin, which is known to have an active site involving four copper atoms. One copper sits alone, and is used to oxidize substrates that dock with the protein. This electron is then transferred to a three copper ring, where it is finally used to reduce oxygen to water. A coarse grained view of this electron transfer, and knowledge of the participating amino acids, might shed insight into this mechanism, and how this enzyme might be tuned for application.

Among mycotoxins, aflatoxins (AFs) produced by *Aspergillus* species are the most dangerous natural pollutants, being associated with several diseases, from liver damage and liver cancer to childhood stunting. Aflatoxin contamination is thus a major food safety concern. To protect consumers, contaminated foodstuff is disposed of every year; decontamination is costly and unsafe in industrialized countries, and often unfeasible in developing ones.

One bioremediation strategy is to take advantage of natural occurring enzymes. Among available options, laccase is a versatile and environment-friendly candidate with the potential to detoxify aflatoxin-contaminated food commodities. Laccase itself has been studied widely before, and is regarded as the “ideal green catalyst” and a pivotal enzyme in biotechnological research. The enzyme is a monomeric multicopper oxidase of relatively small molecular weight, first identified in *Rhus vernicifera* at the end of 19th century in Japan. The enzymatic function consists of a single-electron oxidation of a substrate, coupled with a four-electron reduction of molecular oxygen to water.

By performing QM modeling of the full system, and post-processing using the Fragment Bond Order tool described in our earlier publication, we can generate a more detailed, graph-like view of the electronic interactions with the laccase site (Fig. 10). These interactions can be used to define a substrate specific binding site. In this framework, we observe differences in the interaction strength of the toxin and copper depending on the particular configuration. The substrate binding site definitions presented here can guide such future studies.

The present work provides its contribution by addressing the issue of Aflatoxin decontamination from a novel angle, which is mechanistic by vocation and interdisciplinary by necessity: a bottom-up understanding of the detoxification process. QM modeling is by definition a mechanistic approach and can thus be used to validate any alternative empirical approach. Reservations about its employment are only limited to its feasibility.

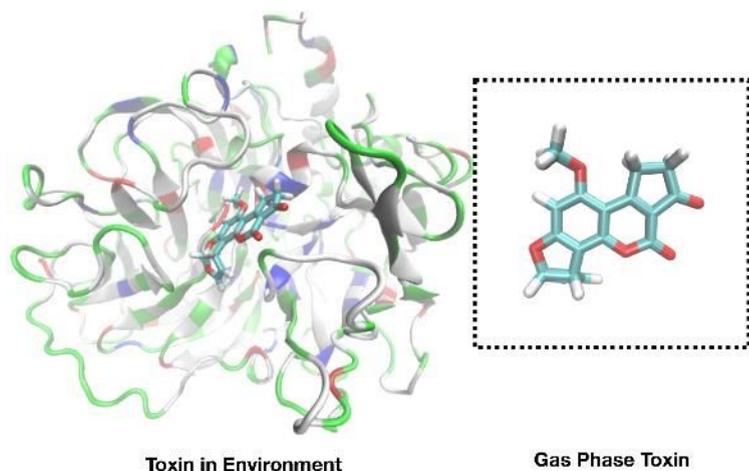


Figure 8: Toxin molecule in the gas phase (right) and in the environment of the protein (left)

Figure 9: Docking of Aflatoxin poses. The protein is represented by a wired isosurface and the coppers are represented as orange balls. The HIS and ASN amino acids of the catalytic triad are highlighted as solid pink and purple surfaces respectively.

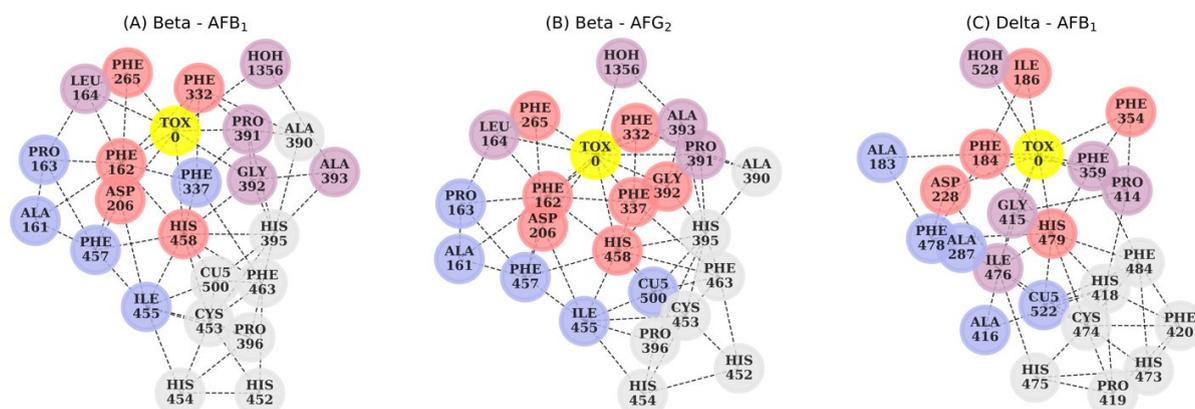
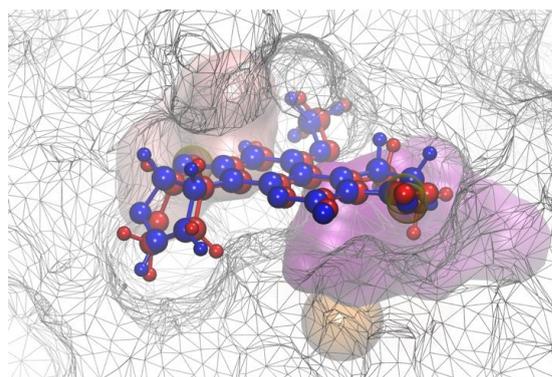


Figure 10: Graph representation of the substrate specific binding sites. The toxin is labeled “TOX 0” in yellow, and the lone copper “CU5 522”. Nodes are labeled according to the protein's residues. The strength of the interaction with the toxin is colored with red as the strongest, followed by purple, blue, and finally gray (weakest).



It is thus appealing to consider a full QM approach which would shed light on various relevant questions, for instance: (i) Is the oxidation rate of the toxin dependent on how the latter approaches the active site? (ii) Is the enzyme active site altered by the toxin presence? (iii) How accurate are the QM/MM approximations with respect to an unbiased, full QM calculations? In order to provide answers to such questions, it is greatly beneficial to employ a QM approach that is, potentially, able to treat systems up to many thousands of atoms at the QM level of theory. The BigDFT code, employed in this context for QM modeling, has been proven to be able to tackle KS-DFT calculations of systems up to few tens of thousands atoms, and can provide reliable information on the identification of the systems' fragments and associated physical observables.

On the IRENE-AMD Rome partition, of the French TGCC center, a single energy point calculation of such a system takes 80 minutes of walltime on 16 Nodes (with 16 MPI tasks per node and 8 OMP threads per task), which amounts to about 3000 CPU hours per calculation. It is interesting to notice that such calculations can be performed with a very little number of compute nodes per calculation, thereby enabling the possibility of high-throughput calculations with large systems. Moreover these calculations enable also the possibility of performing fragmentation of the system, to understand which amino acids have to be merged together in order to have a QM-reliable fragmentation that can be then used as the definition of building blocks for an in depth system analysis.

Work is ongoing in the direction of the full QM characterization of prototypical enzyme-substrate complex conformations. A successful, rational approach to laccase engineering is, nowadays, not beyond feasibility.

### 3.3 Full DFT simulation of new particle-like objects in chiral magnets

Such particle-like objects are currently under intense scrutiny as entities enabling new and exciting concepts in information storage and processing within the framework of spintronics. Currently their description relies on a multiscale approach in which DFT is used to extract model parameters for atomic and micromagnetic simulations. While such simulations are very efficient in the description of these large magnetic objects, the question of accuracy of the underlying model and the task to include all relevant interactions remain challenging.

We are currently investigating two different classes of structures with slightly different focus as outlined in the two sub-projects below. To facilitate these calculations we applied for a NIC large-scale computational project to obtain CPU quota on the SuperMUC-NG. 35 MioCore-h were granted for the simulations. In addition we use smaller available quota on the Stuttgart supercomputers and the JARA-allocations in JSC and the RWTH-Aachen university.

#### 3.3.1 Skyrmions in 2D topological non-trivial insulators

It was shown that hydrogenated Bi (111) monolayers can exhibit large topological bandgaps and manifest a quantum spin Hall state. In the case of partial hydrogen decoration, p-state



magnetism is found leading to a quantum anomalous Hall (QAH) state. As polarized edge states are formed in these non-trivial phases, these materials with their large band gaps are interesting candidates for exploitation in spintronic transport concepts. As the partially hydrogenated Bi (111) is magnetic, the effect of magnetic order on the electronic structure can be significant. In particular the complex interplay between the topological protection on the QAH state and the magnetic order introduces a rich playground in which details of the electronic structure can be controlled and manipulated by the arrangement of magnetic moments of the system. While such effects have been simulated by means of simple tight-binding models, these models can only indicate the basic phenomenon but fail to grasp the details. For example, the introduction of magnetic skyrmion will lead to the formation of metallic states in the material and thus modify the intrinsic electronic structure of the material significantly. We therefore plan to study large unit-cells with non-trivial magnetic order imposed.

Our first system for simulating a skyrmion is a  $8 \times 8$  supercell film HBi with 192 atoms. Due to the size of the system we only use a single k-point and hence have to rely on the eigenvalue parallelism in FLEUR alone. In the non-collinear calculations, the sizes of the Hamiltonian and Overlap matrices are about  $200000 \times 200000$  double precision complex numbers. One iteration of the self-consistency cycle takes 47 minutes on 256 nodes of the SuperMUC-NG supercomputer (Intel Skylake 48 cores/node).

Being a film, this case differs considerably from systems used in our benchmark suite. Hence, the code performance needs some tuning for these special code paths and corresponding use cases will be profiled.

### *3.3.2 Bloch points in 3D materials*

A Bloch point manifests itself as a singularity at which the magnetisation vanishes. This indicates that drastic changes in the electronic structure can be expected as the intra-atomic exchange interaction will compete with effects originating from the long-range magnetic order. Thus, we expect the electronic state of the Bloch point to demonstrate peculiar features unknown so far and not accessible in simpler model approaches.

Our initial test system for simulating a Bloch point is a MnGe supercell. From micromagnetic simulations based on the Landau-Lifshitz-Gilbert equation we obtained six test cases ranging from  $5 \times 5 \times 8$  supercell with 1600 atoms to  $8 \times 8 \times 10$  supercell with 5120 atoms. We are currently using these magnetic configurations as the input for the FLEUR calculations. To understand the behaviour and scalability of the code, we started to perform non-collinear simulations with a  $2 \times 2 \times 4$  MnGe supercell containing a total of 128 atoms and using 18 k-points. The Hamiltonian and overlap matrices of this system are of the size about  $20000 \times 20000$  double precision complex numbers, and a single self-consistency iteration takes 17 minutes to complete on 6 nodes of the supercomputer CLAIX 2018 (Intel Skylake, 48 cores/node). As a first step towards the study of a more realistic system, we performed collinear simulations with one of the production systems, a  $5 \times 5 \times 8$  MnGe supercell with 1600 atoms. These are again single k-point calculations. The Hamiltonian and overlap matrices of this system are of the size about  $120000 \times 120000$  double precision complex elements. For this system a single self-consistency iteration takes 75 minutes to complete on 64 nodes of



the SuperMUC-NG supercomputer (Intel Skylake 48 cores/node) using 1 k-point. It might be necessary to increase the number of k-points to improve the convergence, which will increase the computation effort accordingly. As the final simulations will have to take the non-collinear spin configuration into account, a further doubling of the matrix sizes will have to be considered.

### 3.4 RPA and double hybrid based MD simulations of condensed phase systems

RPA type of calculations is sometimes called the 5th rung in Jacob's ladder in DFT for the reason that its accuracy outperforms the accuracy of semi-local, meta-GGA and hybrid functions. This comes at a huge computational cost as it involves dealing with the subspace of the unoccupied Kohn-Sham states (infinite in theory). Numerical implementation of RPA boils down to a matrix-matrix multiplication problem with dimensions  $(\mathbf{n}, \mathbf{k}) \times (\mathbf{k}, \mathbf{n})$  for each frequency point  $\omega$ , where  $\mathbf{n}$  is the auxiliary basis size used to expand electron-hole pairs and  $\mathbf{k}$  is the number of such electron-hole pairs. In the benchmark of 128 water molecules the values of  $\mathbf{n}$  and  $\mathbf{k}$  are 17408 and 3473408 correspondingly. The cost of such matrix-matrix multiplication is  $2.1 \times 10^{15}$  Flop for each  $\omega$  point. In the CP2K code this operation is performed by the PBLAS function `pdgemm` which until now was implemented in free Netlib version (CPU only) and in proprietary vendor libraries (MKL, Cray LibSci, IBM ESSL, etc.)

In the past year, the work has been done to create an optimized `pdgemm` wrapper for COSMA library and test its performance in the RPA calculation of the 128-water molecule system (feasibility up to 512 molecules demonstrated). COSMA is a parallel, high-performance, GPU-accelerated, matrix-matrix multiplication algorithm that is communication-optimal for all combinations of matrix dimensions, number of processors and memory sizes, without the need for any parameter tuning. The results of the benchmarking are reported in detail in D4.3 ("Second report on code profiling and bottleneck identification"). We managed to scale the CP2K RPA calculation of 128 water molecules to 1024 nodes reaching 1.256 TFlop/s/node performance, significantly exceeding the performance of the best vendor library on GPUs (Cray LibSci) (**2x on the scientific testcase**), and further gain for larger systems. The efficiency of the calculation, allowing for energy calculations in <3min on 1024 Piz Daint GPUs, will allow Monte Carlo sampling (requiring 1000s of energy evaluations) as GPU-accelerated pre-exascale architectures become available (tentative 2021). Quite importantly, the COSMA library is vendor agnostic (i.e. not Cray proprietary), and already targets NVIDIA and AMD GPUs, with others easy to add, ensuring that **calculations will be possible on all of the most likely pre-Exascale systems**. The current code is thus a suitable prototype to perform the challenged simulations planned (solvation studies of ions), but is undergoing further development to increase efficiency and the breadth of applications possible. This includes the GPU acceleration of the calculations of the basic integrals that enter the RPA matrix elements, as well as the development and implementation of the necessary response theory that will give access to ionic forces in the context of RPA. The latter ambitious development will be a milestone for practical applications of RPA, and would open a completely new field of application.





1. Electron dynamics within NEQ-MBPT is implemented in the real-time module of the Yambo code (`yambo_rt`). The module has been recently released under GPL (yambo 4.4). The adopted parallelization scheme was focused on the distribution of the workload for the dissipative processes, i.e. the terms coming out from the highest level approximations to the many-body self-energy. This is the most time-consuming part of the real time module and has been optimized for short time simulations (up to few hundreds of fs). For the MaX demonstrator we will focus instead, at least at first, on the lowest level approximations for the many-body self energy which can more easily allow for real time propagation up to few tens of ps (i.e. 10.000 fs). To this end, in the past months we have re-designed the parallelization scheme, allowing for a distribution in memory of the density matrix on different MPI tasks based on the k-point index. In doing so we obtained both a reduction of the memory consumption in large scale simulations and a good scaling of the simpler real time propagation, allowing us to reach, for the electronic part only, simulations of the desired length. In such longer simulations the I/O of data also became a relevant bottleneck. This is why, together with the new parallelization scheme we re-designed also the output of binary data introducing parallel I/O, unlimited size variables and caching. Thanks to all these ingredients the I/O time became negligible.
2. The Electron dynamics implementation is based on the KS basis set and the equilibrium Hamiltonian is provided by Quantum ESPRESSO. In particular the (non-local) ionic pseudo potentials,  $V_I(R; r, r')$  are directly constructed from the KS eigenvalues and not from the atomic positions. However if the atomic positions  $R$  change during the dynamics,  $V_I(R; r, r')$  need to be updated. This is why we have interfaced the Yambo code with the `upf_pseudo` library. Starting from that library we are planning to build the subroutine which defines  $V_I(R; r, r')$  dynamically during the real time simulation. A second issue to handle is that the KS basis set itself depends on the atomic positions  $R$  and, once the positions change on a scale which is significant compared to the typical spread of the electrons' wave-functions, the basis set needs to be updated. This would mean that Yambo needs to call back Quantum ESPRESSO on the fly during the dynamics, get the new basis set, and go on with the dynamics. Such approach has been already used in recently published papers and could be handled by a python layer which calls back and forth Quantum ESPRESSO and `yambo_rt`. We started a re-styling of the *whypy* python code with the precise goal of handling such procedures. Up to now *whypy* was used mostly for standard GW and/or BSE simulations. The first step has been to implement new functionalities to handle `yambo_rt` simulations. Once this step will be over, we will move to the handling of `yambo_rt` together with Quantum ESPRESSO.

#### 4 Conclusions and perspectives

Pre-exascale demonstrators have highlighted a number of challenging technological issues to be addressed by WP1-5 of the MaX project. A list providing the most relevant feedback obtained from the deployment of pre-exascale demonstrators follows.



1. One of the most compelling problems shared by many flagship codes is the identification and the adoption of effective solutions for large distributed linear algebra problems on accelerated systems. This issue has already been encountered in different fields of numerical modeling and a number of innovative solutions have been proposed recently. Indeed candidate options are represented by open source libraries like COSMA, the new ELPA2 and SLATE. These potential solutions should be tested and validated using miniapps to analyze their performance by technical work packages, before proceeding with their adoption in flagship codes. To this aim it is also of utmost importance to exploit the well established HW-SW co-design framework, where optimal solutions for vendor specific architectures can be identified.
2. A second issue is represented by the continuous drop of the ratio between the amount of memory and memory bandwidth with respect to the computational power available on each compute entity. On top of this, a constantly growing fraction of the computational power of each node in pre-exascale HPC clusters is provided by accelerators, where memory hierarchies must be carefully controlled by the developers. The direct consequence of these technological trends is the need to accurately distribute data structures over the most powerful compute elements of the cluster. This has been already observed in pre-exascale systems, for example in the case of Car-Parrinello simulation of a COVID-19 protein discussed above, where the memory footprint of Ultrasoft projectors, currently replicated on all computational devices, reached about half of the total memory available on each GPU card.
3. A point that deserves attention is the presence of new communication channels inside each node. For example, most GPU accelerated clusters feature direct communication channels among accelerators in order to avoid the bottleneck introduced by the memory bus. These hardware specific features can be directly exploited into applications with vendor specific APIs or throughout external libraries, like, for example, OpenMPI. Understanding and defining effective strategies for taking advantage of these new data exchange protocols is of great importance, since shared options can drastically reduce the burden of maintaining additional communication layers inside each flagship application.
4. The integration of the codes with AiiDA is considered a key factor in the deployment of many of the demonstrators, ensuring the handling of complex workflows, not only involving many simultaneous jobs in high-throughput exploration schema, but also allowing the interfacing and interoperability between different codes and applications which need to share data and intermediate results. Work is underway in this direction in several “on-the-fly” demonstrators, but will require a deeper level of integration between the codes, AiiDA and the specific capabilities of each computational environment concerning memory and storage structure and availability.