



## **Electronic structure E-CAM modules III**

E-CAM Deliverable 2.4

Deliverable Type: Report

Delivered in September, 2018



E-CAM

The European Centre of Excellence for  
Software, Training and Consultancy  
in Simulation and Modelling



Funded by the European Union under grant agreement 676531

### Project and Deliverable Information

Project Title	E-CAM: An e-infrastructure for software, training and discussion in simulation and modelling
Project Ref.	Grant Agreement 676531
Project Website	<a href="https://www.e-cam2020.eu">https://www.e-cam2020.eu</a>
EC Project Officer	Juan PELEGRIN
Deliverable ID	D2.4
Deliverable Nature	Report
Dissemination Level	Public
Contractual Date of Delivery	Project Month 34(31 <sup>st</sup> July, 2018)
Actual Date of Delivery	12 <sup>th</sup> September, 2018
Description of Deliverable	9 software modules delivered to the E-CAM repository in the area of Electronic Structure responding to requests of users, and their documentation.

### Document Control Information

Document	Title:	Electronic structure E-CAM modules III
	ID:	D2.4
	Version:	As of September, 2018
	Status:	Accepted by WP leader
	Available at:	<a href="https://www.e-cam2020.eu/deliverables">https://www.e-cam2020.eu/deliverables</a>
	Document history:	<a href="#">Internal Project Management Link</a>
Review	Review Status:	Reviewed
	Action Requested:	<b>Submit</b>
Authorship	Written by:	Valerio Vitale(University of Cambridge)
	Contributors:	Francesco Fracchia (Scuola Normale Superiore, Pisa) David López Durán (CIC nanoGUNE, San Sebastian)
	Reviewed by:	Mike Payne (University of Cambridge)
	Approved by:	Mike Payne (University of Cambridge)

### Document Keywords

Keywords:	E-CAM, Electronic Structure, Module, CECAM, Documentation, ...
-----------	--

12<sup>th</sup> September, 2018

**Disclaimer:** This deliverable has been prepared by the responsible Work Package of the Project in accordance with the Consortium Agreement and the Grant Agreement. It solely reflects the opinion of the parties to such agreements on a collective basis in the context of the Project and to the extent foreseen in such agreements.

**Copyright notices:** This deliverable was co-ordinated by Valerio Vitale<sup>1</sup> (University of Cambridge) on behalf of the E-CAM consortium with contributions from Francesco Fracchia (Scuola Normale Superiore, Pisa) David López Durán (CIC nanoGUNE, San Sebastian) . This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0>.



<sup>1</sup>[vv284@cam.ac.uk](mailto:vv284@cam.ac.uk)

# Contents

<b>Executive Summary</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Overall scope of the module set . . . . .	2
1.2 General applications and possible exploitation of the codes . . . . .	3
<b>2 Modules</b>	<b>4</b>
2.1 GRASP_Sampling . . . . .	4
2.1.1 Practical application and exploitation of the code . . . . .	4
2.2 Differential_Evolution . . . . .	4
2.2.1 Practical application and exploitation of the code . . . . .	4
2.3 WLRR . . . . .	4
2.3.1 Practical application and exploitation of the code . . . . .	5
2.4 GROMACS_Interface . . . . .	5
2.4.1 Practical application and exploitation of the code . . . . .	5
2.5 GAUSSIAN_Interface . . . . .	5
2.5.1 Practical application and exploitation of the code . . . . .	5
2.6 SCDM_WFs . . . . .	6
2.6.1 Practical application and exploitation of the code . . . . .	6
2.7 Selectively-localized-Wannier-functions . . . . .	6
2.7.1 Practical application and exploitation of the code . . . . .	6
2.8 Geomoltools . . . . .	6
2.8.1 Practical application and exploitation of the code . . . . .	7
2.9 ESL-Demo . . . . .	7
2.9.1 Practical application and exploitation of the code . . . . .	7
<b>3 Outlook</b>	<b>8</b>
<b>References</b>	<b>9</b>

## Executive Summary

In this report for Deliverable 2.4 of E-CAM, 9 software modules in the field of electronic structure—[Work Package 2 \(WP2\)](#)—are presented. They are intended to be exploited during the development process of new applications that span most of the electronic structure community, from force field fitting to High-Throughput Computing (HTC) of materials as identified in the [Deliverable 2.2](#). The modules presented here originate from E-CAM's academic user base, as well as input from E-CAM Pilot Projects' industrial partners. One module was developed by the [Electronic Structure Library \(ESL\)](#) developers team as an extension to the module set presented in [Deliverable 2.1](#) (from an Extended Software Development Workshop (ESDW) held by E-CAM in Zaragoza in June 2016).

The 9 modules are:

- GRASP\_Sampling,
- Differential Evolution,
- Wlrr,
- Gromacs\_interface,
- Gaussian\_interface,
- SCDM\_WFs,
- Selectively-Localized-Wannier-functions,
- Geomoltools,
- ESL-DEMO.

Specifically, these include five modules produced as a result in response to discussions with [Pilot Project 2 \(PP2\)](#)'s industrial partner [BiKi Technologies](#), two modules in response to [Pilot Project 3 \(PP3\)](#)'s academic user base, one module (a suite of 8 programs) from the work done in [Pilot Project 1 \(PP1\)](#) in collaboration with [Merck Solutions](#) and one module from ESL developers team.

The five modules from PP2—GRASP\_Sampling, Gromacs\_interface, Gaussian\_interface, Differential Evolution and Wlrr—are general codes that can be used for the parametrization of force fields of metal ions via a combination of machine learning techniques, as described in Ref. [5]. Notably, these codes can be applied to optimize the parameters of a general objective function with arbitrary functional form, with respect to given reference data.

The two modules from PP3—SCDM\_WFs and Selectively-Localized-Wannier-functions—contain FORTRAN codes for the automation of the generation of Wannier functions (WFs) for any given input structure, which will likely have great impact on high-throughput calculations of materials' properties. In fact, these modules are part of the Wannier90 code which has been successfully integrated in the Automated Interactive Infrastructure and Database for Computational Science (AiiDA) work-flow[15].

The module from PP1—Geomoltools—contains 8 programs for the manipulation of molecular geometries, from conversion between different formats to direct manipulations of the atomic coordinates, which hugely simplifies the work even for complex operations, i.e. translating and aligning a molecule with some predefined axes, or converting geometries between two formats used in different electronic structure codes.

The module from the ESL developers—ESL-DEMO—is a simulation software that serves as an executable documentation showing all the capabilities of the Library.

For each module a concise description is provided in [Section 2](#). Additional information concerning the codes where these modules have been implemented is also given. The source code related to each module may be found in the E-CAM GitLab repository and for the two Wannier90-related modules, e.g., SCDM\_WFs and Selectively-Localized-Wannier-Functions, in the [GitHub Wannier90 repository](#). At the end of each module links to the E-CAM documentation repository are provided.

# 1 Introduction

[Work Package 2](#) (WP2) of E-CAM focuses on selecting software functionalities that are common to many electronic structure methods, are important for the implementation and efficiency of codes, and mature enough to allow for a good definition of standards and interfaces.

In the proposed module set, 9 new modules are presented in the area of Electronic Structure. Eight modules are a direct product of the work carried out by E-CAM developers in response to the user base needs (both academic and industrial) as identified in the [Deliverable 2.2](#), and one is an important addition to the Electronic-Structure Library module set, already part of the E-CAM repository (see [Deliverable 2.1](#)).

E-CAM software development in WP2 has mainly focused on three aspects: 1) the creation of an environment to foster the transition from in-house software to high performing and reliable community packages with long-term sustainability, 2) extending the capability of well-established codes, such as [Wannier90](#), both in terms of core functionalities and of the number of interfaces to other existing codes and 3) the production of industry-driven packages. In this respect, the present module set combines these principles rigorously, presenting modules developed as a result of discussions with the user base, both academic and industrial, and from the Electronic-Structure Library (ESL) developers team, whose goal is to build a community-maintained library of software of use for electronic structure simulations.

The module developments are stored on [E-CAM GitLab repository](#) and [Wannier90's GitHub repository](#). In order to document each module, a module documentation directory is created in an [E-CAM GitLab repository dedicated to Electronic Structure](#) which automatically updates documentation for the [E-CAM Electronic Structure Module Documentation](#) on [Read the Docs](#). A [detailed contribution procedure](#) is available through the documentation repository.

## 1.1 Overall scope of the module set

Engineering materials for technological applications, or for new drugs, relies on our ability to describe and manipulate matter at the atomic scale. To this end, it is crucial to understand how atoms interact to form molecules and complex materials and how these respond to external probes. This, in turn, requires us to find accurate solutions to the fundamental equations governing systems made of many interacting particles. Several approximate models exist to address the problem, from very accurate and computationally demanding quantum chemistry methods to less accurate and very fast coarse-grain methods. Currently, Density Functional Theory (DFT) represents *de facto* the tool to investigate and predict properties of new materials, as it offers the best compromise between accuracy and computational cost.

Sometimes an in-depth understanding of processes can be achieved by looking at the electronic structure in real space. This can be facilitated by the use of localized Wannier functions (lWFs). These functions are generated directly from the outputs of a DFT calculation. Hence, lWFs play an auxiliary rôle to DFT in the understanding of many phenomena and are often employed by the electronic-structure community[11]. Moreover, lWFs provide a powerful tool to link DFT calculations to more accurate quantum mechanical calculations, such as single particle Green's function (G) and screened Coulomb potential (W) (GW), Dynamical mean field theory (DMFT), or to larger scale methods such as *ab initio* Tight Binding (TB). Being able to automatically generate Wannier functions from DFT calculations would open the way to high-throughput calculations (HT) of advanced materials properties. However, the generation of lWFs is not always a trivial task, and very often the process necessitates user intuition and chemical knowledge, which is hard to implement in computer codes. E-CAM has addressed this challenge by implementing a new algorithm by Damle, Lin and Ying[3], called "Selective columns of density matrix" (SCDM) and a heuristic to assign the value of its parameters, which allows to perform automated computation of localized Wannier functions.

On the other hand, for biological systems (with several thousands of atoms), where DFT calculations are not computationally feasible yet and fast but highly accurate classical force field (FFs) methods are necessary instead. E-CAM contribution in this very active area of research has focused on engineering FFs of metal ions in biological environments, e.g. metalloprotein catalytic sites. The major challenges here are related to the proper treatment of quantum effects, which are extremely difficult to describe within classical approaches. To effectively describe these effects, many simple models are insufficient, and new, more flexible, multi-parameter potential (e.g., including polarization) becomes necessary. Five modules within E-CAM WP2 have been developed to address this problem. The result is a general procedure to generate non-bonded FFs of metal ions using statistical learning, without affecting the FFs of the other atoms in the system, so that they could be easily integrated into consolidated Molecular Mechanics (MM) packages[5].

## 1.2 General applications and possible exploitation of the codes

### Metal ion force fields parametrization modules

The suite of 5 modules—GRASP\_Sampling, WLRR, Differential\_Evolution, GROMACS\_Interface and GAUSSIAN\_Interface—constitute the building blocks for the software tool LRR-DE, which has been successfully used to parametrize non-bonded force fields of metal ions, as explained in the recently published paper by Fracchia *et al.*[5].

### Wannier functions modules

The two modules described in this Deliverable for the automatic generation of Wannier functions and improvement of their symmetry, are not stand-alone codes, but are part of the main Wannier90 program and of the pw2wannier90 code, the latter being the interface between the popular Quantum ESPRESSO[7] code and Wannier90. They are distributed under the same GNU General Public licence used for the Wannier90 program. The code for the automation of the Wannier functions, i.e. the SCDM\_WFs module, has also been integrated in the AiiDA workflow[15], which allows the user to interact effortlessly with any number of remote HPC resources, and perform high-throughput calculations of advanced materials' properties. Moreover, the Selectively-localized-Wannier-functions module plays a complementary rôle to the Symmetry-Adapted-Wannier-functions module of Deliverable 2.3 in computing Wannier functions which respect the point-group symmetry of their centres.

Moreover, all the other electronic structure codes that are already interfaced with Wannier90, e.g. SIESTA[16], VASP[10] WIEN2k, GPAW[13], abinit[8] will greatly benefit from these developments.

### Geomoltools module

Geomoltools is intended to be used within a project devoted to the study of the energy and geometry properties of molecules and together with other codes like Avogadro[9] and SIESTA[16]. An example of application is the ongoing "Pilot Project on Calculations for Applications in Photovoltaic Devices" in the WP2 Electronic Structure of E-CAM.

### ESL-Demo module

The ESL Demonstrator is a basic atomic-scale simulation software illustrating how to use and bring together the various available components of the Electronic-Structure Library. ESL-Demo is part of ESL and is already used by newcomers to the electronic-structure field who want to learn how to perform electronic-structure calculations.

## 2 Modules

In this section, a brief description for each module is provided, followed by a link to the Merge-Request (Pull-Request for modules developed for the Wannier90 code) on [GitLab E-CAM library](#) ([GitHub repository of Wannier90](#)), which shows detailed information about code development, testing and documentation.

**The following 5 modules have been developed directly within the context of WP2 Pilot Project 2 within E-CAM.**

### 2.1 GRASP\_Sampling

The GRASP\_Sampling module performs a stratified sampling of the atomic configurations, described by vectors, of a molecular system to build a representative training set to be used in a force field fitting procedure. Given a list of candidate configurations, and selected size  $N$  of the training set required, the module executes the combinatorial optimization that maximizes a dissimilarity score among the trial elements of the training set. The dissimilarity score is based on the mutual distances between the elements, his maximization assures high coverage of the domain of the configurations.

#### 2.1.1 Practical application and exploitation of the code

The combinatorial optimization is performed using the GRASP algorithm, illustrated in the paper of Feo and Resende[4]. The application of the GRASP algorithm to perform a stratified sampling is described in the work of Fracchia *et al.* [5]. The module is applied to build the training set in the fitting procedure LRR-DE[5] of force fields of metal ions with respect to *ab initio* data.

Direct Documentation Link	<a href="#">readme.rst of GRASP_Sampling Module</a>
Merge-Request Link	<a href="#">Merge-Request for the GRASP_Sampling Module</a>

### 2.2 Differential\_Evolution

This module performs a single-objective global optimization in a continuous domain using the meta-heuristic algorithm “Success-History based Adaptive Differential Evolution” (SHADE)[18]. SHADE is a recent adaptive version of the differential evolution algorithm[17], a stochastic population-based derivative-free optimizer. The input of the module is the objective function, the upper and lower limits of the domain for each dimension of the search space, and the parameters of the algorithm. The parameters of the algorithm are the size of the vector population, the maximum number of evaluations of the objective function, and the parameters that govern the termination of the optimization. The output provides the coordinates of the identified minimum and the corresponding value of the objective function.

#### 2.2.1 Practical application and exploitation of the code

The module is a component of the software tool LRR-DE, developed to parametrize force fields of metal ions. In particular, the role of the SHADE algorithm in LRR-DE is the optimization of the hyperparameters of the model. However the module has general applicability to the black-box minimization of any cost function.

Direct Documentation Link	<a href="#">readme.rst of Differential_Evolution Module</a>
Merge-Request Link	<a href="#">Merge-Request for the Differential_Evolution Module</a>

### 2.3 WLRR

This module solves the weighted linear ridge regression problem by calculating the linear parameters of a model selected by the user that minimize the deviations of the predictions from the references of the data set. Thus, it is a supervised learning tool that optimizes the linear parameters of an analytic expression in order to fit a data set. Each element of the data set can be weighted according to the relative importance or reliability attributed by the user. The regularization provides a protection from overfitting, this issue can occur if the flexibility of the model is too high in relation to the available data. Moreover, the module calculates the *leave-one-out* cross-validation error for the employed data set.

### 2.3.1 Practical application and exploitation of the code

The WLRR module is a component of the LRR-DE software tool, developed to parametrize FFs of metal ions. In the LRR-DE software tool, the WLRR module is combined with the meta-heuristic optimization algorithm differential evolution in order to tune the hyperparameters of the model (the regularization parameter and the non-linear parameters of the model). The LRR-DE module has been developed to parametrize force fields of metal ions, however the method can be applied to optimize the parameters of a general functional form with respect to reference data.

Direct Documentation Link	<a href="#">readme.rst of WLRR Module</a>
Merge-Request Link	<a href="#">Merge-Request for the WLRR Module</a>

## 2.4 GROMACS\_Interface

### GROMACS\_Interface

The GROMACS\_Interface module reads the configurations of a molecular system generated by the molecular dynamics code GROMACS and prepares the input for the GRASP Sampling module.

The module performs the following operations:

1. It reads the configurations generated by the molecular dynamics
2. It calculates, for each configuration, the Euclidean distances of all atoms from the metal ion.
3. It identifies the permutationally equivalent atoms
4. It performs a Gaussian transformation of the distances
5. It calculates the variances of transformed distances
6. It selects the coordinates with the higher variances
7. It prepares the input for the GRASP\_Sampling module as a matrix including the transformed distances for all the configurations.

### 2.4.1 Practical application and exploitation of the code

Used in the GRASP\_Sampling module to link the results of classical simulations on  $\text{Ca}^{2+}$ ,  $\text{Zn}^{2+}$ ,  $\text{Mg}^{2+}$ ,  $\text{Ni}^{2+}$  and  $\text{Na}^+$  in water with periodic boundary conditions to the LRR-DE engine[5].

Direct Documentation Link	<a href="#">readme.rst of GROMACS_Interface Module</a>
Merge-Request Link	<a href="#">Merge-Request for the GROMACS_Interface Module</a>

## 2.5 GAUSSIAN\_Interface

The Gaussian\_Interface module generates the Gaussian input files of the clusters included in the training set to produce the reference data for the fitting procedure of the force fields. The clusters are generated by cutting a selected configuration from a larger biochemical system. The user must specify an atom of the configuration, which will be the centre of the cluster, and a cut-off distance.

### 2.5.1 Practical application and exploitation of the code

The module saturates the residues by adding Hydrogen atoms to all broken bonds, and prepares the input file for the calculation of the energy and the forces of the cluster generated in the Gaussian format. It was used in the GRASP\_Sampling module to generate the results in [5].

Direct Documentation Link	<a href="#">readme.rst of GAUSSIAN_Interface Module</a>
Merge-Request Link	<a href="#">Merge-Request for the GAUSSIAN_Interface Module</a>



**The following 2 modules have been developed directly within the context of WP2 Pilot Project 3 within E-CAM.**

## 2.6 SCDM\_WFs

This module is the implementation of the SCDM-k method[3] for obtaining a set of localised Wannier functions without the need for an initial guess for the functions. The method is also applicable to the case of a set of bands that do not form an isolated manifold (i.e., are entangled). The algorithm employs a QR factorization with column pivoting [2] (QRCP) of the density matrix, to construct the localised functions to be used by Wannier90 as initial guesses.

The SCDM\_WFs module is part of the Quantum ESPRESSO code[7], more precisely of the interface between Quantum ESPRESSO and Wannier90[14], called `pw2wannier90`.

### 2.6.1 Practical application and exploitation of the code

This module has been successfully used to generate Maximally Localized Wannier functions (MLWFs) for a variety of solid-state systems, exploiting the AiiDA[15] work-flow, which allows a user to automatically generate MLWFs for materials without having to specify initial parameters. This opens the way for high-throughput, user-intervention-free computation of advanced materials' properties with Wannier functions.

Direct Documentation Link	<a href="#">readme.rst of SCDM_WFs Module</a>
Merge-Request Link	<a href="#">Merge-Request for the SCDM_WFs Module</a>

## 2.7 Selectively-localized-Wannier-functions

This module is the implementation of the Selective-Localization algorithm by Marianetti *et al.*[19] to generate selectively localized Wannier functions. This extends the method of Marzari and Vanderbilt[12] in two important ways: 1) it allows the user to focus on localizing only a subset of orbitals of interest and 2) to fix centres of these orbitals ensuring the preservation of the point-group symmetry. The module is part of the Wannier90 code.

### 2.7.1 Practical application and exploitation of the code

This module is of particular interest when using beyond-DFT methods, to study electronic and transport properties of transition metals and metal oxides. In these systems, it is important to obtain Wannier Functions which respect the symmetries of the Hamiltonian as shown in Ref. [19].

Direct Documentation Link	<a href="#">readme.rst of Selectively-Localized-Wannier-Functions Module</a>
Merge-Request Link	<a href="#">Merge-Request for the Selectively-localized-Wannier-functions Module</a>

**The following module, containing a suite of 8 codes, has been developed directly within the context of WP2 Pilot Project 1 within E-CAM.**

## 2.8 Geomoltools

Geomoltools is a set of computer codes designed to manipulate molecules. From simple changes of coordinates (Z-matrix to XYZ coordinates and vice-versa) to more complicated operations as the generation of different stacking arrangements between molecules-are quick and easy to perform. The user is free to install the whole set or each code independently. The dependency of each single code with common libraries is managed through the Makefile file, which leads to an easy and straightforward compilation. The 8 Fortran codes that constitute the set are:

1. `mol2xyz`: converts a `.mol` file into an ordered `.xyz` file.
2. `pastemol`: joins two `.xyz` files.
3. `movemol`: translates and aligns the molecule with some predefined axes.
4. `stackmol`: generates (manually or randomly) different stacking arrangements between two molecules.
5. `geodiff`: compares the internal coordinates of two molecules.
6. `xyz2zmt_s`: converts the Cartesian coordinates contained in a `.xyz` file into Z-matrix (2 possible formats).

7. `zmt2xyz_s`: converts a Z-matrix (from 2 possible formats) into cartesian coordinates.
8. `ucubcellgen`: calculates the vectors of a unit cell given some atomic coordinates.

These codes use the libraries `libmanmol`, `libalignmol` and `libjoinmol`, which contain modules, subroutines and functions that can be easily extracted in order to be integrated into other codes.

The set Geomoltools is available under the GNU General Public License (GPL) version 2.

### 2.8.1 Practical application and exploitation of the code

Geomoltools is intended to be used within a project devoted to the study of energetic and geometric properties of molecules in combination with other codes like Avogadro[9] and Siesta[16]. An example of application is the ongoing “Pilot Project on Calculations for Applications in Photovoltaic Devices” in the WP2 Electronic Structure of E-CAM.

Direct Documentation Link	<a href="#">readme.rst of Geomoltools Module</a>
Merge-Request Link	<a href="#">Merge-Request for the Geomoltools Module</a>

## 2.9 ESL-Demo

The ESL Demonstrator is a basic atomic-scale simulation software illustrating how to use and bring together the various available components of the Electronic Structure Library. It is meant to be used as a concrete implementation example for both end-users and developers. For users, it evidences and explains the typical operations and building blocks of an electronic structure code. For developers, it shows how to bring together the different ESL components in a consistent way. It constitutes a global “executable documentation” for the ESL. It is itself documented in a standard way, using [Doxygen](#) (which generates documentation from source code), to provide relevant explanations about how to use each ESL component in the appropriate context.

### 2.9.1 Practical application and exploitation of the code

ESL-Demo is part of ESL and is already used by newcomers to the electronic-structure field who want to learn how to perform electronic-structure calculations.

Direct Documentation Link	<a href="#">readme.rst of ESL-Demo Module</a>
Merge-Request Link	<a href="#">Merge-Request for the ESL-DEMO Module</a>

### 3 Outlook

The report of Deliverable 2.4 of E-CAM describes 9 Software modules in the field of electronic structure. They include a wide range of developments from new techniques for force field parametrization, in particular for force fields of metal ions in molecular systems, to algorithms and heuristic for the automatic generation of Wannier functions, which opens the way for high-throughput (HT) user-intervention-free computation of advanced materials' properties. As outlined in Sec. 2, five of these modules are a result from the work done in the Pilot Project 2 of WP2. They deal with the parametrization of force field of metal ions in (bio)molecular systems. In particular, the `GRASP_Sampling` module, in 2.1, is used for the generation of training sets with maximal dissimilarity score<sup>2</sup> to be used by the LRR-DE method[5] for the parametrization of FFs with respect to *ab initio* data. For a general, flexible multi-parameter FF, the `WLRR` module is used for solving the weighted ridge linear regression to optimize the linear parameters of the model, whereas the `Differential_Evolution` module is used to optimize the non-linear parameters, minimizing the leave-one-out cross-validation (LOOCV) error. The last two modules, i.e., `GROMACS_Interface` and `GAUSSIAN_Interface` are two codes to interface with the GROMACS program[1] and the GAUSSIAN program[6], respectively.

Two other modules come from the Pilot Project 3 and deal with the automation of Wannier functions for performing high-throughput calculations of advanced materials' properties. The `SCDM_WFs` module in 2.6 is part of the Quantum ESPRESSO code[7] and it is integrated in the AiiDA[15] work-flow. This module solves the perennial problem of defining an initial guess for the generation of maximally localized Wannier functions (MLWFs), which used to hinder the automation of Wannier functions and consequently their in HTC of materials.

In Sec. 2.8 the `Geomoltools` module is presented, a suite of 8 codes used to perform several operations on geometry files, which greatly simplifies the production of complex molecular structures. This module is currently used in the Pilot Project 1 of WP2 of E-CAM.

Finally, a simulation software from the Electronic-Structure Library developers, namely `ESL-DEMO`, is presented in Sec. 2.9. It serves as an executable documentation of the library and its capabilities in performing general all-purpose electronic-structure calculations.

---

<sup>2</sup>based on the mutual distances between the elements of the set.

## References

### Acronyms Used

<b>CECAM</b>	Centre Européen de Calcul Atomique et Moléculaire
<b>TB</b>	Tight Binding
<b>DFT</b>	Density Functional Theory
<b>WFs</b>	Wannier functions
<b>IWFs</b>	localized Wannier functions
<b>ESDW</b>	Extended Software Development Workshop
<b>HPC</b>	High Performance Computing
<b>DMFT</b>	Dynamical mean field theory
<b>GW</b>	single particle Green's function (G) and screened Coulomb potential (W)
<b>ESL</b>	Electronic-Structure Library
<b>PP1</b>	Pilot Project 1
<b>PP2</b>	Pilot Project 2
<b>PP3</b>	Pilot Project 3
<b>WP2</b>	Work Package 2
<b>AiiDA</b>	Automated Interactive Infrastructure and Database for Computational Science
<b>FFs</b>	force field
<b>MM</b>	Molecular Mechanics
<b>HTC</b>	High-Throughput Computing
<b>HT</b>	high-throughput calculations
<b>SCDM</b>	Selected columns of density matrix

### URLs referenced

#### Page ii

<https://www.e-cam2020.eu> ... <https://www.e-cam2020.eu>  
<https://www.e-cam2020.eu/deliverables> ... <https://www.e-cam2020.eu/deliverables>  
 Internal Project Management Link ... <https://redmine.e-cam2020.eu/issues/177>  
[vv284@cam.ac.uk](mailto:vv284@cam.ac.uk) ... <mailto:vv284@cam.ac.uk>  
<http://creativecommons.org/licenses/by/4.0> ... <http://creativecommons.org/licenses/by/4.0>

#### Page 1

Work Package 2 ... <https://www.e-cam2020.eu/electronic-structure/>  
 Deliverable 2.2 ... <https://zenodo.org/record/1209090>  
 Electronic Structure Library ... [https://esl.cecama.org/Main\\_Page](https://esl.cecama.org/Main_Page)  
 Deliverable 2.1 ... <https://zenodo.org/record/1185170#.W3FbH62w3IE>  
 Pilot Project 2 ... <https://www.e-cam2020.eu/pilot-project-biki-2/>  
 BiKi Technologies ... <http://www.bikitech.com/>  
 Pilot Project 3 ... <https://www.e-cam2020.eu/pilot-project-on-electronic-structure-functionalities-for-m>  
 Pilot Project 1 ... <https://www.e-cam2020.eu/pilot-project-photovoltaic-devices/>  
 Merck Solutions ... <https://www.merckgroup.com/en/expertise/functional-solutions/solutions.html>  
 GitHub Wannier90 repository ... <https://github.com/wannier-developers/wannier90>

#### Page 2

Work Package 2 ... <https://www.e-cam2020.eu/electronic-structure/>  
 Deliverable 2.2 ... <https://zenodo.org/record/1209090>  
 Deliverable 2.1 ... <https://zenodo.org/record/1185170#.W3FbH62w3IE>

Wannier90... <http://wannier.org>  
E-CAM GitLab repository... <https://gitlab.e-cam2020.eu:10443/e-cam/Electronic-Structure-Modules>  
Wannier90's GitHub repository... <https://github.com/wannier-developers/wannier90>  
E-CAM GitLab repository dedicated to Electronic Structure... <https://gitlab.e-cam2020.eu/E-CAM/Electronic-Structure-Modules>  
E-CAM Electronic Structure Module Documentation... <https://readthedocs.org/projects/e-cam-electronic-structure-modules>  
Read the Docs... <https://readthedocs.org/>  
detailed contribution procedure... <http://e-cam-electronic-structure-modules.readthedocs.io/en/latest/contributing.html>

### Page 3

Deliverable 2.3... <https://zenodo.org/record/1207633#.W3rWsq2w3IE>  
SIESTA... <https://departments.icmab.es/leem/siesta/>  
VASP... <https://www.vasp.at/>  
WIEN2k... <http://susi.theochem.tuwien.ac.at/>  
GPAW... <https://wiki.fysik.dtu.dk/gpaw/>  
abinit... [https://www.abinit.org/sites/default/files/infos/7.6/input\\_variables/varw90.html](https://www.abinit.org/sites/default/files/infos/7.6/input_variables/varw90.html)  
Avogadro... <https://avogadro.cc/>  
SIESTA... <https://departments.icmab.es/leem/siesta/>

### Page 4

GitLab E-CAM library... <https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library>  
GitHub repository of Wannier90... <https://github.com/wannier-developers/wannier90>  
readme.rst of GRASP\_Sampling Module... [http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/GRASP\\_Sampling/readme.html](http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/GRASP_Sampling/readme.html)  
Merge-Request for the GRASP\_Sampling Module... [https://gitlab.e-cam2020.eu:10443/e-cam/Electronic-Structure-Modules/GRASP\\_Sampling/merge\\_requests/34](https://gitlab.e-cam2020.eu:10443/e-cam/Electronic-Structure-Modules/GRASP_Sampling/merge_requests/34)  
readme.rst of Differential\_Evolution Module... [http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/Differential\\_Evolution/README.html](http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/Differential_Evolution/README.html)  
Merge-Request for the Differential\_Evolution Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/20](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/20)

### Page 5

readme.rst of WLRR Module... <http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/WLRR/README.html>  
Merge-Request for the WLRR Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/32](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/32)  
readme.rst of GROMACS\_Interface Module... [http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/GROMACS\\_interface/README.html](http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/GROMACS_interface/README.html)  
Merge-Request for the GROMACS\_Interface Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/18](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/18)  
readme.rst of GAUSSIAN\_Interface Module... [http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/Gaussian\\_interface/README.html](http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/Gaussian_interface/README.html)  
Merge-Request for the GAUSSIAN\_Interface Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/19](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/19)

### Page 6

readme.rst of SCDM\_WFs Module... [https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/SCDM\\_WFs/readme.html](https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/SCDM_WFs/readme.html)  
Merge-Request for the SCDM\_WFs Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/21](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/21)  
readme.rst of Selectively-Localized-Wannier-Functions Module... <http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/Selectively-Localized-Wannier-Functions/readme.html>  
Merge-Request for the Selectively-localized-Wannier-functions Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/61](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/61)

### Page 7

readme.rst of Geomoltools Module... <http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/Geomoltools/readme.html>  
Merge-Request for the Geomoltools Module... [https://gitlab.e-cam2020.eu:10443/e-cam/Electronic-Structure-Modules/Geomoltools/merge\\_requests/29](https://gitlab.e-cam2020.eu:10443/e-cam/Electronic-Structure-Modules/Geomoltools/merge_requests/29)  
Doxygen... <http://www.stack.nl/~dimitri/doxygen/index.html>  
readme.rst of ESL-Demo Module... <http://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/modules/esl-demo/readme.html>

Merge-Request for the ESL-DEMO Module... [https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge\\_requests/47](https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/merge_requests/47)

- 
- [1] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C. Smith, Berk Hess, and Erik Lindahl. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1-2:19 – 25, 2015.
- [2] A. Damle and L. Lin. Disentanglement via entanglement: A unified method for Wannier localization. *ArXiv e-prints*, March 2017.
- [3] Anil Damle, Lin Lin, and Lexing Ying. Scdm-k: Localized orbitals for solids via selected columns of the density matrix. *Journal of Computational Physics*, 334:1 – 15, 2017.
- [4] Thomas A. Feo and Mauricio G. C. Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6(2):109–133, Mar 1995.
- [5] Francesco Fracchia, Gianluca Del Frate, Giordano Mancini, Walter Rocchia, and Vincenzo Barone. Force Field Parametrization of Metal Ions From Statistical Learning Techniques. *Journal of Chemical Theory and Computation*, November 2017.
- [6] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman, and D. J. Fox. Gaussian~16 Revision B.01, 2016. Gaussian Inc. Wallingford CT.
- [7] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L Chiarotti, Matteo Cococcioni, Ismaila Dabo, Andrea Dal Corso, Stefano de Gironcoli, Stefano Fabris, Guido Fratesi, Ralph Gebauer, Uwe Gerstmann, Christos Gougoussis, Anton Kokalj, Michele Lazzeri, Layla Martin-Samos, Nicola Marzari, Francesco Mauri, Riccardo Mazzarello, Stefano Paolini, Alfredo Pasquarello, Lorenzo Paulatto, Carlo Sbraccia, Sandro Scandolo, Gabriele Sclauszero, Ari P Seitsonen, Alexander Smogunov, Paolo Umari, and Renata M Wentzcovitch. Quantum espresso: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39):395502, 2009.
- [8] X. Gonze, B. Amadon, P.-M. Anglade, J.-M. Beuken, F. Bottin, P. Boulanger, F. Bruneval, D. Caliste, R. Caracas, M. Côté, T. Deutsch, L. Genovese, Ph. Ghosez, M. Giantomassi, S. Goedecker, D.R. Hamann, P. Hermet, F. Jollet, G. Jomard, S. Leroux, M. Mancini, S. Mazevet, M.J.T. Oliveira, G. Onida, Y. Pouillon, T. Rangel, G.-M. Rignanese, D. Sangalli, R. Shaltaf, M. Torrent, M.J. Verstraete, G. Zerah, and J.W. Zwanziger. Abinit: First-principles approach to material and nanosystem properties. *Computer Physics Communications*, 180(12):2582 – 2615, 2009.
- [9] Marcus D. Hanwell, Donald E. Curtis, David C. Lonie, Tim Vandermeersch, Eva Zurek, and Geoffrey R. Hutchison. Avogadro: an advanced semantic chemical editor, visualization, and analysis platform. *Journal of Cheminformatics*, 4(1):17, Aug 2012.
- [10] G. Kresse and J. Hafner. Ab initio. *Phys. Rev. B*, 47:558–561, Jan 1993.
- [11] Nicola Marzari, Arash A. Mostofi, Jonathan R. Yates, Ivo Souza, and David Vanderbilt. Maximally localized wannier functions: Theory and applications. *Rev. Mod. Phys.*, 84:1419–1475, Oct 2012.
- [12] Nicola Marzari and David Vanderbilt. Maximally localized generalized wannier functions for composite energy bands. *Phys. Rev. B*, 56:12847–12865, Nov 1997.
- [13] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen. Real-space grid implementation of the projector augmented wave method. *Phys. Rev. B*, 71:035109, Jan 2005.
- [14] Arash A. Mostofi, Jonathan R. Yates, Giovanni Pizzi, Young-Su Lee, Ivo Souza, David Vanderbilt, and Nicola Marzari. An updated version of wannier90: A tool for obtaining maximally-localised wannier functions. *Computer Physics Communications*, 185(8):2309 – 2310, 2014.

- 
- [15] Giovanni Pizzi, Andrea Cepellotti, Riccardo Sabatini, Nicola Marzari, and Boris Kozinsky. Aiida: automated interactive infrastructure and database for computational science. *Computational Materials Science*, 111:218 – 230, 2016.
- [16] José M Soler, Emilio Artacho, Julian D Gale, Alberto García, Javier Junquera, Pablo Ordejón, and Daniel Sánchez-Portal. The siesta method for ab initio order- n materials simulation. *Journal of Physics: Condensed Matter*, 14(11):2745, 2002.
- [17] Rainer Storn and Kenneth Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, Dec 1997.
- [18] R. Tanabe and A. Fukunaga. Success-history based parameter adaptation for differential evolution. In *2013 IEEE Congress on Evolutionary Computation*, pages 71–78, June 2013.
- [19] Runzhi Wang, Emanuel A. Lazar, Hyowon Park, Andrew J. Millis, and Chris A. Marianetti. Selectively localized wannier functions. *Phys. Rev. B*, 90:165125, Oct 2014.