Electronic structure E-CAM modules IV

E-CAM Deliverable 2.5
Deliverable Type: Other
Delivered in January, 2020

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	https://www.e-cam2020.eu
EC Project Officer	Juan PELEGRIN
Deliverable ID	D2.5
Deliverable Nature	Other
Dissemination Level	Public
Contractual Date of Delivery	Project Month 47(31st August, 2019)
Actual Date of Delivery	10th January, 2020
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 IV
	ID:	D2.5
	Version:	As of January, 2020
	Status:	Draft
	Available at:	https://www.e-cam2020.eu/deliverables
	Document history:	Internal Project Management Link
Review	Review Status:	Accepted by WP leader
Authorship	Written by:	Michele Ruggeri(CEA)
	Reviewed by:	Mike Payne (University of Cambridge), Emine Kucukbenli (Harvard University)
	Approved by:	Mike Payne (University of Cambridge)

Document Keywords

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

10th January, 2020

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.*

[1] michele.ruggeri@cea.fr

Contents

Executive Summary

In this report for Deliverable 2.5 of E-CAM, 9 software modules for electronic structure computations —Work Package 2 (WP2)—are presented.

The modules in this report are part of larger software packages and are intended to extend their features and functionalities in the study of condensed matter physics, material science and quantum chemistry.

In particular five of the presented modules are part of the PANNA library [6]. Recently great interest has developed in the theoretical condensed matter and material science communities for the application of artificial intelligence and more specifically neural networks to the study of atomistic systems; PANNA (Properties from Artificial Neural Network Architectures) is a Python package for the design, implementation and deployment of deep neural networks for the study of solid state, and especially for crystal systems. The tools provided in this package allow the user to input their own data to train and validate neural networks to model condensed matter systems; an interface with the TensorFlow package allows efficient and user friendly implementation and monitoring of the network. Computation of energies and forces are available, to perform structural optimazion, and an interface with the molecular dynamics software LAMMPS allows the network model to be used to build an effective interatomic potential.

We present in this document five modules related to the generation and storing of data for the descriptor vectors used in the neural networks in PANNA , for the training of the networks themselves and for the computation of physical observables

- PANNA-GVECT: this module efficiently generates Behler-Parinello and modified Behler-Parinello descriptor vectors, used in the training of deep neural networks.
- PANNA-TFR: this module is used to convert and store data for Behler-Parinello and modified Behler-Parinello descriptor vectors in the TensorFlow data format.
- PANNA-TRAIN: a module that implements the training of a neural network for extracting atomistic data, to predict e.g. the total energy and the interatomic forces for a solid state system given its crystal structure.
- PANNA-EVAL: this module evaluates the physical quantities of the system described by a fully connected neural network, for the validation of the network itself or to export information so that it can be used by other softwares for atomistic simulations.
- PANNA-Charges: starting from the description via neural network of a condansed matter system, this module allows the computation of the local atomic charges.

Four more modules presented here, were written as part of the Electronic-Structure Library (ESL) project; Electronic structure simulations are a fundamental tool in the study of a large array of problems in condensed matter and quantum chemistry, which have a large range of applications in several field of research, from condensed matter physics and material science to quantum chemistry to biology and nanomedicine. Theoretical studies in electronic structure are mostly performed independently by separate research groups, employing a wide array of different methods (such as but not limited to Density Functional Theory classical and quantum Monte Carlo and various quantum chemistry methods), each implemented in different softwares and employing different data standards.

While undoubtedly many remarkable results were achieved the sheer variety of methods and tools used in the electronics structure community may make the communication, data exchange and collaboration between different research group challenging. The goal of the Electronic Structure Library initiative is to provide a library of standard, high quality software for electronic structure computations, to facilitate sharing of data, results and ultimately knowledge between different research group, and to allow scientists to focus more on their research topics than to coding and software engineering.

During the *ESDW: Scaling Electronic Structure Applications* held in Dublin in January 2019, four modules were developed as part of the ESL:

- LibGridXC: this module provides routines for the computation of exchange and correlation energy and potential for Density Functional Theory (DFT) simulations, both in atomic and periodic systems, with a large selection of LDA, GGA and van der Waals DFT functionals.
- ELSI: contributed during the E-CAM ESDW in Dublin in 2019, the ELectronic Structure Infrastructure (ELSI) is a library of numerical methods to deal with linear algebra problems, in particular to solve eigenvalue equations in electronic structure simulations. Scalability and portability were the main design pillars of this module: this library can efficiently run on a wide array of machines, going from laptops to HPC supercomputers, both on CPUs and GPUs.

- ESL Easyconfigs, ELPA-EasyBlock: state of the art computations in electronic structure require the use of large HPC machines, with software that must be written, installed and built in an optimal way, depending on the specifics of each supercomputer. Often this process of optimisation can require considerable effort; these two modules provides the tool to easily and efficiently install, configure and build scientific software from the ESL on HPC supercomputers, considerably reducing the detailed fine tuning and time effort needed to properly set up ab initio simulations. These two modules were written during an E-CAM ESDW held in Dublin in 2019 and an ESDW held in Lausanne in 2018, respectively.

A short description is written for each module, followed by a link to the respective Merge-Request on the GitLab service of E-CAM. These merge requests contain detailed information about the code development, testing and documentation of the modules.

1 Introduction

Work Package 2(WP2) of E-CAM focuses on software for the study of electronic structure; understanding the behaviour of electron systems is fundamental to evaluate and predict the properties of physical systems of interest in condensed matter physics and quantum chemistry, which have a great range of applications e.g. in material science, biology and nano-medicine.

There are several methods for the theoretical study of electronic structure systems, such as the Density Functional Theory, Quantum Monte Carlo, mean field and quantum chemical methods, and more recently techniques based on artificial intelligence and neural networks. All these techniques rely on computer simulations, and the aim of the WP2 of E-CAM is to provide high quality, standard codes and interfaces that can be used in these studies.

1.1 Overall scope of the module set

Five of the modules presented in this report are part of a larger software package, PANNA, and four of the modules are part of the Electronic-Structure Library project.

Recently great interest has developed in the theoretical condensed matter and material science communities for the application of artificial intelligence and more specifically neural networks to the study of atomistic systems; PANNA (Properties from Artificial Neural Network Architectures) is a Python package for the design, implementation and deployment of deep neural networks for the study of solid state, and especially for crystal systems. The tools provided in this package allow the user to input their own data to train and validate neural networks to model condensed matter systems; an interface with the TensorFlow package allows efficient and user friendly implementation and monitoring of the network. Computation of energies and forces are available, to perform structural optimisation, and an interface with the molecular dynamics software LAMMPS allows the network model to be used as an effective inter-atomic potential. Five modules related to the PANNA package are presented in this report.

Four more modules were written as part of the ESL project. Theoretical studies in electronic structure are mostly performed independently by separated research groups, employing a wide array of different methods (such as but not limited to DFT, classical and quantum Monte Carlo, and various quantum chemistry methods), each implemented in different software and employing different data standards. While undoubtedly many remarkable results have been achieved, the sheer variety of methods and tools used in the electronics structure community may make the communication, data exchange and collaboration between different research groups challenging. The goal of the Electronic Structure Library initiative is to provide a library of standard, high quality software for electronic structure computations, to facilitate sharing of data, results and ultimately knowledge between different research groups, and to allow scientists to focus more on their research topics rather than coding and software engineering.

1.1.1 Neural network models of condensed matter systems with PANNA

We present in this document five modules related to the generation and storing of data for descriptor vectors in a neural networks in PANNA, for the training of the networks themselves and for the computation of physical observables:

- PANNA-GVECT: this module efficiently generates Behler-Parinello and modified Behler-Parinello descriptor vectors, used in the training of deep neural networks.
- PANNA-TFR: this module is used to convert and store data for Behler-Parinello and modified Behler-Parinello descriptor vectors in the TensorFlow data format.
- PANNA-TRAIN: a module that implements the training of a neural network for extracting atomistic data, to predict e.g. the total energy and the interatomic forces for a solid state system given its crystal structure.
- PANNA-EVAL: this module evaluates the physical quantities of the system described by a fully connected neural network, for the validation of the network itself or to export information so that it can be used by other software for atomistic simulations.
- PANNA-Charges: starting from the description via neural network of a condensed matter system, this module allows the computation of the local atomic charges.

1.1.2 Extending the functionality of the Electronic-Structure Library

Four modules were developed as part of the ESL:

- LibGridXC: this module provides routines for the computation of exchange and correlation energy and potential for DFT simulations, both in atomic and periodic systems, with a large selection of LDA, GGA and van der Waals DFT functionals.
- ELSI: contributed during the E-CAM ESDW in Dublin in 2019, ELectronic Structure Infrastructure is a library of numerical methods to deal with linear algebra problems, in particular to solve eigenvalue equations in electronic structure simulations. Scalability and portability were the main design pillars of this module: this library can efficiently run on a wide array of machines, going from laptops to HPC supercomputers, both on CPUs and GPUs.
- ESL Easyconfigs, ELPA_EasyBlock: state of art computations in electronic structure require the use of large HPC machines, with software that must be written, installed and built in an optimal way, depending on the specifics of each supercomputer. Often this process of optimisation can require considerable effort; these two modules provide tools to easily and efficiently install, configure and build scientific software from the ESL on HPC supercomputers, considerably reducing the detailed fine tuning and time effort needed to properly set up ab initio simulations. These modules were written during an E-CAM ESDW held in Dublin in 2019 and an ESDW held in Lausanne in 2018, respectively.

1.2 General applications and possible exploitation of the codes

Recently great interest has arisen for applications of artificial intelligence and neural networks in a large range of fields, one of these being material science. An efficient package for training neural networks and use them to successfully model physical systems of interest such as PANNA can therefore be of great interest both for fundamental research in academia and for material science research in industry. Being able to accurately compute potential energy surfaces, charged distributions and effective interaction potentials and forces can allow, for example, the characterisation of new materials or the modelling of chemical reactions.

The goal of the ESL is to provide a repository of tested, common software for electronic structure computations that can work as a tool box of utilities used by many electronic structure codes, that is simple to use by non-experts in a wide range of fields and applications (material science, chemistry, biology, ...). This allows researchers from both industry and academia to focus more on research in their specialised field rather than in software design or matters relating to the installation and compilation of their codes. The aim of the ESL modules set present in this deliverable is to expand the ESL repository, providing efficient, tested code to be included in any software for DFT simulations, or enabling an easier, more straightforward installation, compilation and usage of pre-existing ESL code on High Performance Computing (HPC) machines.

1.3 How to read this report

For each module, we give a short overview, followed by links to the `Merge-Request` and the `Documentation` on the GitLab service of E-CAM, which shows detailed information about code development, testing and documentation. The documentation shows how to use the modules in practice, while possible practical exploitation and industrial applications have been outlined above (partly, with some additional remarks added below).

2 Modules based on PANNA

The base code for the following five modules is PANNA [6]. PANNA (Properties from Artificial Neural Network Architectures) is a package for the implementation, training and validation of neural networks to represent atomic energy potentials. PANNA is written in Python and is based on the open source TensorFlow package.

2.1 PANNA-Charges

2.1.1 Module description

The PANNA-Charges module demonstrates how to train a neural network to predict local atomic charges. This network can later be used to calculate the electrostatic energy density of a crystal. See Ref. [2] for the theoretical model behind this approach. PANNA-Charge supports periodic and aperiodic structures, multiple species, and a different all-to-all connected network architecture for each species. It further supports control of the training dynamics, e.g. freeze/unfreeze layers, weight transfer, decaying learning rates, etc.

Direct Documentation Link	readme.rst of the PANNA-Charges Module
Merge-Request Link	Merge-Request for the `PANNA-Charges` Module

2.2 PANNA-EVAL

2.2.1 Module description

PANNA-EVAL module evaluates an all to all connected neural network to predict atomistic quantities, e.g. total energy and forces of a given crystal structure. PANNA-EVAL can be used with other modules of the PANNA project for neural network validation, but it can also serve to carry the information of the trianed network to other platforms such as molecular dynamics code LAMMPS.

Direct Documentation Link	readme.rst of the PANNA-EVAL Module
Merge-Request Link	Merge-Request for the `PANNA-EVAL` Module

2.3 PANNA-GVECT

2.3.1 Module description

The PANNA-GVECT is used to efficiently generate Behler-Parinello [3] and modified Behler-Parinello descriptors [8]. These descriptors can then be used in machine learning algorithms. Even though these descriptors were originally designed for neural network models, they are equally suitable for other supervised learning schemes such as kernel methods, or unsupervised ones such as clustering techniques.

PANNA-GVECT, unlike other modules within the PANNA project, does not use the TensorFlow framework.

Direct Documentation Link	readme.rst of the PANNA-GVECT Module
Merge-Request Link	Merge-Request for the `PANNA-GVECT` Module

2.4 PANNA-TFR

2.4.1 Module description

PANNA-TFR module demonstrates how to pack the Behler-Parinello and modified Behler-Parinello descriptor vectors (See References [1], [2], [3]) written in binary format, into TensorFlow data format for efficient reading during training. These descriptors can then be used within TensorFlow efficiently, reducing the overhead during batch creation. PANNA-TFR is built on TensorFlow.

Direct Documentation Link	readme.rst of the PANNA-TFR Module
Merge-Request Link	Merge-Request for the `PANNA-TFR` Module

2.5 PANNA-TRAIN

2.5.1 Module description

PANNA-TRAIN is a neural network training module for atomistic data, eg. prediction of total energy and forces given a crystal structure. It implements a separate atomic network for each species, following the seminal work of Behler and Parinello (see References [3, 8]) which can later be used as interatomic potential in molecular dynamics simulations. PANNA-TRAIN uses TensorFlow framework as the underlying neural network training and data i/o engine.

Direct Documentation Link	readme.rst of the PANNA-TRAIN Module
Merge-Request Link	Merge-Request for the `PANNA-TRAIN` Module

2.6 Motivation and exploitation of the PANNA modules data set

Despite the large amount of studies concerning application of machine learning and neural networks to condensed matter physics and material science that have been published recently, most of these works are more interested in providing proofs of concept, or very specific examples. PANNA on the other hand has the goal of providing a software for the design of neural networks to be used to accurately model physical systems on large production scale projects, with special attention to performance and the possibility of interfacing with existing codes for e.g. codes for molecular dynamics simulations. PANNA comes with a toolbox that interfaces it with other widely used codes such as Quantum Espresso (QE), VASP, LAMMPS; so that users can generate their test or train data on any of these widely used codes, and then convert it to the code-agnostic PANNA format for machine learning purposes. These tools and code agnostic format can also be used to go between codes QE-> PANNA -> LAMMPS or other way around; allowing the user to mix the training data from different sources, or gradually expand the training set as the training goes on etc.

As an open source software PANNA aims to provide to users both in academia and in the industry a platform for realistic simulations in material science, for example for the prediction of the equilibrium geometry of a solid or a molecule, or the computation of optical spectra and electronic properties of crystals (band gaps, Debye temperatures or the density of states).

3 Modules developed in collaboration with the ESL

The following four modules are developed in collaboration with the ESL initiative.The aim of these modules is to expand the ESL repository, providing efficient, tested code to be included in software for DFT simulations, or enabling an easier, more straightforward installation, compilation and usage of pre-existing ESL code on HPC machines.

3.1 LibGridXC

3.1.1 Module description

LibGridXC is a module extracted from the SIESTA software [9], and now part of the ESL; this module provides a set of routines for the computation of the exchange and correlation energy in DFT computations; these routines can be used to study isolated, spherically symmetrical systems such as atoms or systems in periodic boundary conditions, such as a crystal lattice. Several energy functionals are available, LDA, GGA as well as van der Waals, implemented as in [7].

3.1.2 Motivation and exploitation

DFT is one of the most important methods in electronic structure computations, which has a great number of application both in fundamental research in academia and in applied research and development in industry. One of the fundamental part of a DFT computation is the evaluation of the exchange and correlation contributions to the total energy of the system under study. To understand the behaviour of a physical system is essential to be able to accurately compute the energy, and from it other properties such as the pressure or forces. This module allows a general, tested, accurate and efficient computation of correlation and exchange energies for a wide variety of physical systems and exchange and correlation functional, that can be used to study with DFT a large array of systems, allowing the study for example of chemical reactions, interface physics or electrical and optical properties of crystalline solids.

Direct Documentation Link	readme.rst of the LibGridXC Module
Merge-Request Link	Merge-Request for the `LibGridXC` Module

3.2 ELSI

3.2.1 Module description

ELectronic Structure Infrastructure (ELSI) provides enhanced and scalable solvers for linear algebra problems with applications in electronic structure theory. In particular ELSI deals with the Kohn–Sham eigenvalue problem, which is the core of the DFT. The solution of this problem is often the most expensive part of a DFT computation on a large parallel machine. ELSI provides several different algorithms for the solution of the Kohn–Sham eigenvalue problem, with the corresponding implementations.

3.2.2 Motivation and exploitation

DFT computations are at the base of many computational studies in many fields, such as condensed matter physics, material science, quantum chemistry or molecular biochemistry. ELSI therefore provides tools that can be useful to deal with a large number of problems in modern science, as well as the infrastructure to solve other eigenvalue problems. During the design of this module, portability had the maximum importance: ELSI is a versatile code that can support computing environment ranging from a laptop to large HPC supercomputers. ELSI is used in several electronic structure codes, such as DFTB+ [1], DGDFT [5], FHI-aims [4], and SIESTA [9].

Direct Documentation Link	readme.rst of the ELSI Module
Merge-Request Link	Merge-Request for the `ELSI` Module

3.3 ESL Easyconfigs

3.3.1 Module description

ESL Easyconfigs is a collection of EasyBuild easyconfig files to allow easy installation of software from the ESL in all HPC supercomputers that use the EasyBuild software installation infrastructure. This module includes both a set of recipes to build the libraries in the ESL with all their dependencies.

3.3.2 Motivation and exploitation

Supercomputers are essential in modern computational research, but often installing, configuring and building high performance scientific software in a way that is optimal for the HPC machine in use is a task that can be complicated and time consuming. This module provides a simple, fast way to build code from the ESL to help users (scientist, developers, computer engineers) to install custom complex software combinations without tracking all the dependencies by hand. The recipes in this module will be kept up to date for the Intel and FOSS toolchains, and once the recipes are considered mature enough they will be upstreamed to the EasyBuild official catalogue.

Direct Documentation Link	readme.rst of the ESL Easyconfigs Module
Merge-Request Link	Merge-Request for the `ESL Easyconfigs` Module

3.4 ELPA_EasyBlock

3.4.1 Module description

ELPA_EasyBlock is a module for the automation of the selection of appropriate configuration flags for ELPA within the EasyBuild software, depending on the type of CPU and available features, and allowing the inclusion of additional options if appropriate. It enables single and double precision building of ELPA and ensures the correct linking to the expected version of the linear algebra libraries.

3.4.2 Motivation and exploitation

EasyBuild is a software build and installation framework that allows to manage (scientific) software on HPC systems in an efficient way. Full details on can be found in the EasyBuild documentation. EasyBuild already had limited support for ELPA, this module allows for automated hardware specific configuration and optimisations.

Direct Documentation Link	readme.rst of the ELPA_EasyBlock Module
Merge-Request Link	Merge-Request for the `ELPA_EasyBlock` Module

4 Outlook

The present deliverable contains nine software modules that are part of the software package for electronic structure computations PANNA, and that are part of the ESL project. All the modules have been accepted in the E–CAM software library, and both PANNA and ESL will continue to be updated and developed, to expand their functionality and efficiency.

E-CAM will continue to support such feature-development and adoption by the community, and to support the libraries in their usage by the community.

References

Acronyms Used

CECAM	Centre Européen de Calcul Atomique et Moléculaire
DFT	Density Functional Theory
ESDW	Extended Software Development Workshop
HPC	High Performance Computing
ESL	Electronic-Structure Library
WP2	Work Package 2
LDA	Local-density approximation
GGA	generalized gradient approximations
ELSI	ELectronic Structure Infrastructure

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/178
michele.ruggeri@cea.fr ... mailto:michele.ruggeri@cea.fr
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/
ESDW: Scaling Electronic Structure Applications ... https://www.e-cam2020.eu/event/extended-software-developme
?instance_id=46

Page 3

Work Package 2 ... https://www.e-cam2020.eu/electronic-structure/
PANNA ... https://gitlab.com/PANNAdevs/panna
Electronic-Structure Library project ... https://esl.cecam.org/

Page 4

E-CAM ESDW in Dublin in 2019 ... https://www.e-cam2020.eu/event/extended-software-development-workshop-s
?instance_id=46
ELectronic Structure Infrastructure ... https://wordpress.elsi-interchange.org/
GitLab service of E-CAM ... https://gitlab.e-cam2020.eu/

Page 5

readme.rst of the PANNA-Charges Module ... https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Mo
modules/PANNA-Charges/readme.html
Merge-Request for the PANNA-Charges Module ... https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/219
readme.rst of the PANNA-EVAL Module ... https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modul
modules/PANNA-EVAL/readme.html
Merge-Request for the PANNA-EVAL Module ... https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/169
readme.rst of the PANNA-GVECT Module ... https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Mod
modules/PANNA-GVECT/readme.html
Merge-Request for the PANNA-GVECT Module ... https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/169
readme.rst of the PANNA-TFR Module ... https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Module
modules/PANNA-TFR/readme.html
Merge-Request for the PANNA-TFR Module ... https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/169

Page 6

readme.rst of the PANNA-TRAIN Module . . . https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modu
modules/PANNA-TRAIN/readme.html

Merge-Request for the PANNA-TRAIN Module . . . https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/169

Page 7

readme.rst of the LibGridXC Module . . . https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules
modules/libgridxc/readme.html

Merge-Request for the LibGridXC Module . . . https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/176

readme.rst of the ELSI Module . . . https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Modules/
modules/ELSI/readme.html

Merge-Request for the ELSI Module . . . https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/175

EasyBuild . . . https://easybuild.readthedocs.io

readme.rst of the ESL Easyconfigs Module . . . https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Mod
modules/esl-easyconfigs/readme.html

Merge-Request for the ESL Easyconfigs Module . . . https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/172

Page 8

EasyBuild documentation . . . https://easybuild.readthedocs.io

readme.rst of the ELPA_EasyBlock Module . . . https://e-cam.readthedocs.io/en/latest/Electronic-Structure-Moc
modules/ELPA_easyblock/readme.html

Merge-Request for the ELPA_EasyBlock Module . . . https://gitlab.e-cam2020.eu:10443/e-cam/E-CAM-Library/
merge_requests/174

[1] B. Aradi, B. Hourahine, and Th. Frauenheim. Dftb+, a sparse matrix-based implementation of the dftb method. *The Journal of Physical Chemistry A*, 111(26):5678–5684, 2007. PMID: 17567110.

[2] N. Artrith, T. Morawietz, and J. Behler. High-dimensional neural-network potentials for multicomponent systems: Applications to zinc oxide. *Phys. Rev. B*, 86:079914, 2012.

[3] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, Apr 2007.

[4] Volker Blum, Ralf Gehrke, Felix Hanke, Paula Havu, Ville Havu, Xinguo Ren, Karsten Reuter, and Matthias Scheffler. Ab initio molecular simulations with numeric atom-centered orbitals. *Computer Physics Communications*, 180(11):2175 – 2196, 2009.

[5] Wei Hu, Lin Lin, and Chao Yang. Dgdft: A massively parallel method for large scale density functional theory calculations. *The Journal of Chemical Physics*, 143(12):124110, 2015.

[6] Ruggero Lot, Franco Pellegrini, Yusuf Shaidu, and Emine Kucukbenli. PANNA: Properties from Artificial Neural Network Architectures. *arXiv e-prints*, page arXiv:1907.03055, Jul 2019.

[7] Guillermo Román-Pérez and José M. Soler. Efficient implementation of a van der waals density functional: Application to double-wall carbon nanotubes. *Phys. Rev. Lett.*, 103:096102, Aug 2009.

[8] J. S. Smith, O. Isayev, and A. E. Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.*, 8:3192–3203, 2017.

[9] José M Soler, Emilio Artacho, Julian D Gale, Alberto García, Javier Junquera, Pablo Ordejón, and Daniel Sánchez-Portal. The SIESTA method forab initioorder-nmaterials simulation. *Journal of Physics: Condensed Matter*, 14(11):2745–2779, mar 2002.