

AFLOW π : A minimalist approach to high-throughput *ab initio* calculations including the generation of tight-binding hamiltonians

Andrew R. Supka^{1,2}, Troy E. Lyons¹, Laalitha Liyanage³, Pino D'Amico^{4,5}, Rabih Al Rahal Al Orabi¹, Sharad Mahatara¹, Priya Gopal¹, Cormac Toher^{7,8}, Davide Ceresoli⁶, Arrigo Calzolari^{3,4,5,8}, Stefano Curtarolo^{3,8}, Marco Buongiorno Nardelli^{2,8}, and Marco Fornari^{1,2,8*}

¹ *Department of Physics, Central Michigan University, Mount Pleasant MI, USA*

² *Science of Advanced Materials Program, Central Michigan University, Mount Pleasant MI, USA*

³ *Department of Physics and Department of Chemistry, University of North Texas, Denton TX, USA*

⁴ *Dipartimento di Fisica, Informatica e Matematica, Università di Modena and Reggio Emilia, Via Campi 213/a, 41125 Modena, Italy*

⁵ *CNR-NANO Research Center S3, Via Campi 213/a, 41125 Modena, Italy*

⁶ *CNR-ISTM, Istituto di Scienze e Tecnologie Molecolari, I-20133 Milano, Italy*

⁷ *Materials Science, Electrical Engineering, Physics and Chemistry, Duke University, Durham NC, 27708*

⁸ *Center for Materials Genomics, Duke University, Durham, NC 27708, USA*

***corresponding:** marco.fornari@cmich.edu

Abstract

Tight-binding models provide a conceptually transparent and computationally efficient method to represent the electronic properties of materials. With AFLOW π we introduce a framework for high-throughput first principles calculations that automatically generates tight-binding hamiltonians without any additional input. Several additional features are included in AFLOW π with the intent to simplify the self-consistent calculation of Hubbard U corrections, the calculations of phonon dispersions, elastic properties, complex dielectric constants, and electronic transport coefficients. As examples we show how to compute the optical properties of layered nitrides in the AMN_2 family, and the elastic and vibrational properties of binary halides with CsCl and NaCl structure.

Keywords: High-throughput Calculations, Computer Simulations, Materials Databases

1. Introduction

Accelerated materials discovery is currently the focus of much infrastructure development both for experimental (see for instance Refs. [1–4]) and theoretical research [5–7]. The key components always involve robust data generation or collection, real time feedback and error control, curation and archival of the data, and post-processing tools for analysis and visualization. The ultimate goal is to discover relationships that may be hidden in the large data sets and possibly predict materials with improved specific functionalities.

Flavors of density functional theory (DFT) usually form the basis for the generation of theoretical data since they provide reasonable transferability across different chemical compositions and structural themes at acceptable computational cost. Standard electronic structure packages such as Quantum Espresso [8] or VASP [9] are used as the engine of the calculations but an additional layer of software tools is needed to efficiently manage the high-throughput (HT) workflows. In addition, to streamline data generation, those software tools usually curate the data and prepare them for distribution in online electronic structure properties databases [10–13] that can be mined *a posteriori*.

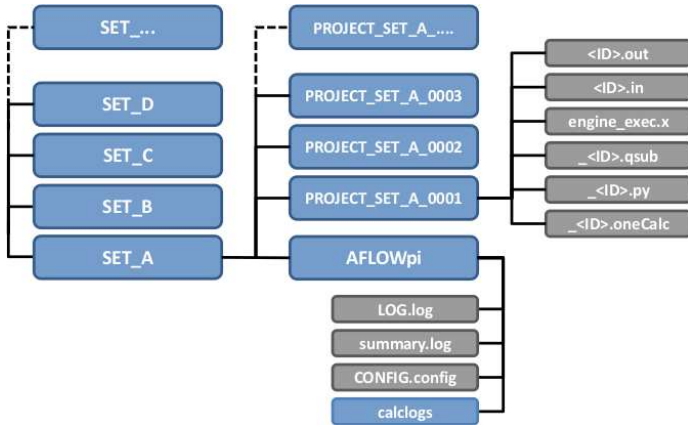


Figure 1: The directory tree created by AFLOW π during the preparation of the calculation with in a PROJECT directory. Each SET contains the AFLOWpi directory with several auxiliary files discussed in the text. PROJECT and SET are substituted with specific values.

Several HT frameworks for electronic structure calculations were discussed in the literature. PYMATGEN [14] is the data generation software infrastructure behind the Materials Project database [10]. It is written in Python 2.7 using libraries such as Numpy and Scipy and exploits VASP and AbInit [15] as standard electronic structure engines. Many tools are available to study phase stability and phase diagrams. AiiDA [16] is an integrated software based on a paradigm involving automation, data, environment, and storage. It is written in Python 2.7 and revolves around relational databases for the overall design in addition to the storage component. For the automation component AiiDA involves workflows that use Quantum Espresso although other electronic structure codes are supported. In addition, scripting interfaces such ASE [17] can be used by AiiDA to control the workflow. ASE is paired with the Computational Materials Repository [18]. Additional HT software tools for density functional calculations are qmpy (associated with the Open Quantum Materials Database) [19] and the HT toolkit (HTTK associated with the Open Materials Database, <http://httk.opendatabase.se>); these software packages are written in Python and streamline electronic structure data generation. The framework AFLOW [20–27] is an efficient tool for high-throughput calculations with VASP. It is written in C++ and includes tools to create input files for Quantum Espresso, to deal automatically with supercells and fractional occupations, to perform symmetry analysis, and to evaluate several thermodynamical and thermal transport properties at different levels of approximation [28, 29]. Data generated with AFLOW

are available online on the AFLOW.org repositories [11, 30, 31] (see <http://www.aflow.org>).

With AFLOW π we are complementing AFLOW with a minimalist software infrastructure that is easily portable, simple to use, and integrated with the AFLOW.org repositories. AFLOW π has been initially developed for verification and testing purposes but has evolved into a modular software infrastructure that provides automatic workflows for the *ab initio* generation of tight-binding hamiltonians within the projected atomic orbital (PAO) scheme [32–34] and the self-consistent calculation of Hubbard U corrections within the ACBN0 approach [35]. In addition, workflows for the calculation of phonon dispersions with Hubbard U correction, the calculation of elastic constants, diffusive transport coefficients, and optical spectra are available. When possible, the code exploits the tight-binding hamiltonians as in Ref. [36].

2. Software design and data organization

AFLOW π 's design provides a simple, task oriented Python package to construct large sets of calculations from one or few input files for a specific *ab initio* engine and then to perform a workflow of tasks for each calculation in the set. The infrastructure is modeled on a simple and common *modus operandi* that involves prototypical input files (reference files) that are modified for desired tasks. The chosen workflow instructs AFLOW π how to modify the reference files and guides the process of submitting, running, and monitoring all calculations.

The software is implemented in Python 2.7 (using the Python standard library and Numpy, Scipy, CIFfile, and Matplotlib). The systematic use of regular expression (`re` module) to parse and modify the input and output files according the specific workflow makes the software very portable and expandable to a variety of electronic structure engines and tools. Currently, AFLOW π encapsulates `pw.x` and several post-processing software from the Quantum Espresso package [8] as well as `ElaStic` [37]. In addition it uses features of `findsym` [38]. The generation and manipulation of the TB Hamiltonians is done with the engine PAO π [39].

The calculations and the resulting data are organized according to the layer structure adopted by AFLOW [30]. The first layer in the directory hierarchy is the

Project layer that can contain multiple instances of the next layer, the **Set** layer. The directories for the individual calculation pipelines in the set are named `PROJECT_SET_XXXX` with `PROJECT` being the project name, `SET` being the name of the set and `XXXX` being the index of the calculation (See Figure 1). Each calculation pipeline receives a unique identifier to keep track of each pipe of calculations during setup and runtime as well as for storage and retrieval of the results. The identifier is constructed as a CRC64 hash [40], which is a unique 16 character string representing the initial *ab initio* engine input file for each calculation pipeline. The input file is first cleaned of commented out content and unnecessary whitespace, then lines of text are sorted alphabetically. Cleaning and sorting is necessary to ensure that extra whitespace or the ordering of the parameters will not change the value of the hash. Files for each task, or step, in each calculation pipeline are designated by `ID_XX` where `XX` is the step number and `ID` is the identifier for that calculation pipeline. An input for the first step in a pipeline with the identifier `20b8c6f83a2ca9b` would read `c20b8c6f83a2ca9b.01.in`. Files and directories within each calculation’s directory that are considered unneeded after runtime, often referred to as `scratch`, such as Python scripts generated by `AFLOW π` , cluster submission files, and temporary files of the *ab initio* engine have a prepended underscore. For example, the filename of the Python executable for the second step in that pipeline would be `_c20b8c6f83a2ca9b_02.py`. `AFLOW π` modularizes the calculation pipelines by separating each task into its own executable. When a task is completed, the next task is initiated and so on until the pipeline completes. The location of the directory for each calculation pipeline is preserved for all tasks the user issues for the set. All files needed to perform the tasks as well as all files generated by the tasks are stored in the folder pertaining to the individual calculation pipeline in order to ensure control and transferability.

Within each **Set**, `AFLOW π` creates a directory named `AFLOWpi` that contains a master runtime log of the session, a copy of the configuration file used in the session, a summary log with relevant information from the output of every calculation within the set, as well as the loadable calculation files (LCFs). In order to ensure data provenance `AFLOWpi` includes mandatory information to be provided at the beginning of the data generation process. Each calculation pipeline has its own LCF that are pickled and used to control and mon-

itor the workflow especially for resuming or expanding the calculations. `AFLOW π` will run as a cluster job submission (currently PBS, <http://pbsworks.com>) or as a local process on systems without a cluster queue. The user has the freedom to choose how the steps in the workflows of the calculations are grouped in terms of cluster submissions.

Specific mechanisms for auto-restart and both manual and automatic error control have been implemented to manage the full set of calculations at once. The restart mechanism keeps track of the time a cluster job has been running and prepares calculations so that the electronic structure engine exits cleanly before the wall-time runs out. A wall-time buffer with a default of 10% of the total wall-time request allows time to save data to disk and, if local scratch option is used, for the temporary files in the local scratch disk space to be copied to the global scratch. After cleaning up temporary files, the command to resubmit the cluster job is given and then the job exits. There are several locking mechanisms in place to ensure that there is no wasted calculation time. In addition to auto restarting, if there are major failures of the electronic structure engine, the user can load a calculation set and filter only the jobs where there were errors causing the execution pipeline to stop at a certain point and check what kind of errors occurred, if any. Calculations that failed can be filtered into a subset. Certain actions can be taken to help the subset to complete.

`AFLOW π` ’s source code includes documentation

```
import AFLOWpi
session = AFLOWpi.prep.init('mHT_TB',
    ↪ 'III-V',config='./mHT_TB.config')
allvars={}
allvars.update(
    _AFLOWPLA_ = ('Al','Ga','In'),
    _AFLOWPLB_ = ('P','As','Sb'),)
calcs = session.scfs(allvars,'mHT_TB.ref')
calcs.vcrelax()
tb_ham = calcs.tight.binding()
tb_ham.dos()
tb_ham.bands()
tb_ham.plot.bands(DOSPlot='APDOS')
calcs.acbn0()
tb_ham = calcs.tight.binding()
tb_ham.dos()
tb_ham.bands()
tb_ham.plot.bands(DOSPlot='APDOS')
calcs.submit()
```

Figure 2: Workflow to compute total energies, tight-binding hamiltonians, band structures, densities of states (DOS), and atom projected DOS with and without ACBN0 for AlP, GaP, InP, AlAs, GaAs, AlSb, GaSb, and InSb. The construction of this workflow is discussed in the text.

strings written in the Google Python docstring format for readability. These docstrings are located at the beginning of each function or class. `AFLOW π` documentation is generated via the `Sphinx` Python package. `Sphinx` can be used to parse source code written in Python, C, C++ as well as several other programming languages to automatically generate user documentation from docstrings. We use `Sphinx` to generate \LaTeX documentation for `AFLOW π` as well as a searchable html documentation.

3. Generation of tight-binding hamiltonians

Theory and applications of the methodology used by `AFLOW π` to generate tight-binding (TB) hamiltonians for first principles plane-wave pseudopotential calculations have been discussed in Refs. [32–34]. Our implementation does not require any additional input with respect to the electronic structure calculations and provides the real space representation of the TB matrix in self-contained XML format. The sparse TB matrix is represented on a grid as $H_{\alpha,\beta}(\mathbf{r})$ and can be easily Fourier transformed into

$$H_{\alpha,\beta}(\mathbf{k}) = \sum_{\mathbf{r}} \exp(i\mathbf{k} \cdot \mathbf{r}) H_{\alpha,\beta}(\mathbf{r}),$$

and diagonalized to determine the full energy dispersions, $E_n(\mathbf{k})$, with the desired level of resolution in reciprocal space or to compute additional properties associated with derivatives in reciprocal space (linear momenta, and effective masses). The real space representation can also be used directly. We have recently applied these methodologies to study diffusive and ballistic transport as well as the optical properties in the independent particles approximation [36].

4. ACBN0 calculations

The TB representation of the electronic structure can be exploited to efficiently compute two-electron integrals for the development of local exchange functionals. `AFLOW π` implements the direct and self-consistent evaluation of the on-site Hubbard U correction that greatly improves the accuracy of standard DFT calculations [35]. Due to the accurate TB representation, the evaluation of the U parameters for atoms in different chemical environments or close to topological defects becomes trivial and `AFLOW π` facilitates the

investigation of such systems. ACBN0 only demands computational resources comparable to regular LDA or PBE calculations. We have extensively investigated the improvements and the limits of ACBN0 calculation in Refs. [41, 42]. Overall, ACBN0 delivers better accuracy for lattice parameters and bulk moduli, improves the energy band gap on semiconductors and insulators in transition metal oxides, and optimizes the phonon dispersions and associated thermal transport properties.

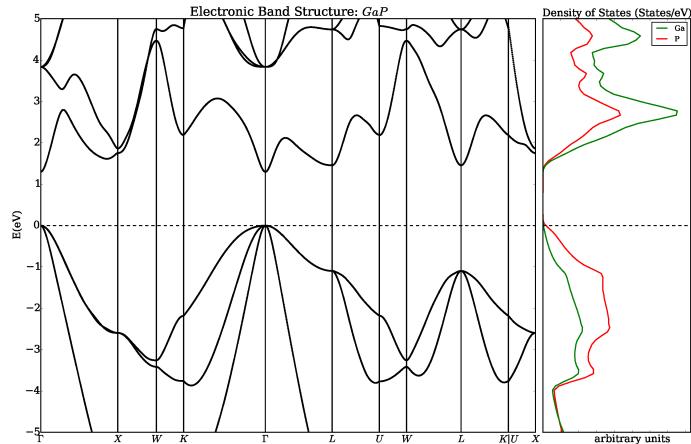


Figure 3: Band structure and DOS plot generated by `AFLOW π` using the tight-binding (TB) hamiltonian for GaP without ACBN0. The workflow in Figure 2 has been used to obtain this result. TB hamiltonians can be post-processed to perform investigations of the energy dispersions away from symmetry lines or interpolations in reciprocal space at minimal computational cost.

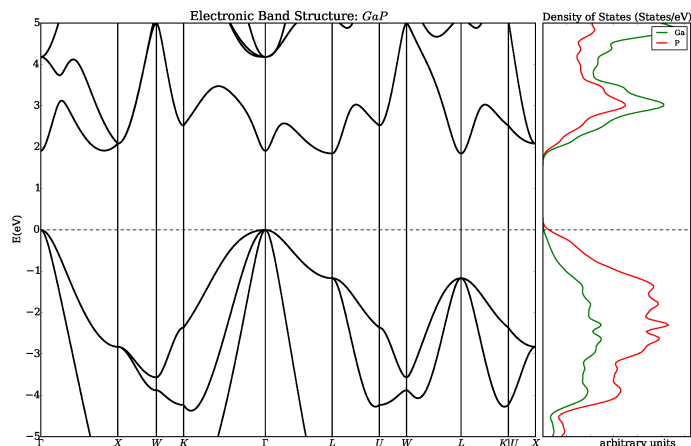


Figure 4: As in Figure 3 but computed with ACBN0. The main improvements for this specific example are associated with a more realistic energy gap, and changes in the dispersion curves near X and L in the conduction band. The top valence bands are also slightly more dispersed.

Table 1: Hubbard U corrections (in eV) computed self-consistently for III-V semiconductors with the workflow discussed in the text. The theoretical lattice parameters a_L^0 and a_L^U (in Å) are computed with and without ACBN0 respectively. Energy gaps (in eV) for $\Gamma - \Gamma$, $\Gamma - X$ and $\Gamma - L$ are calculated with and without ACBN0. Experimental values are shown in parentheses.

	AIP	AlAs	AlSb	GaP	GaAs	InP
U_{III}	0.01	0.08	0.17	20.00	20.01	14.97
U_V	2.31	2.12	1.11	1.87	1.75	0.00
a_L^0	5.49 (5.46)	5.74 (5.66)	6.23 (6.13)	5.53 (5.45)	5.79 (6.05)	5.97 (5.86)
a_L^U	5.44 (5.46)	5.67(5.66)	6.17(6.13)	5.48(5.45)	5.73(6.05)	5.91 (5.86)
E_0^Γ	3.1 (3.6)	1.8 (3.0)	1.2 (2.3)	1.3 (2.8)	0.0 (0.8)	0.4 (1.4)
E_U^Γ	3.9 (3.6)	2.5 (3.0)	1.7 (2.3)	1.9 (2.8)	0.5 (0.8)	1.0 (1.4)
E_0^X	1.3 (2.5)	1.5 (2.2)	1.3 (1.7)	2.1 (2.3)	1.6 (1.1)	1.8 (2.4)
E_U^X	2.1 (2.5)	1.9 (2.2)	1.5 (1.7)	2.1 (2.3)	1.9 (1.1)	2.3 (2.4)
E_0^L	2.7 (2.5)	1.7 (2.4)	1.3 (2.3)	1.4 (2.7)	0.7 (0.9)	1.1 (2.0)
E_U^L	3.3 (2.5)	2.6 (2.4)	1.5 (2.3)	1.8 (2.7)	1.1 (0.9)	1.8 (2.0)

Table 2: Value of the energy gaps and U corrections (in eV) computed with ACBN0 for AMN_2 compounds.

	SrTiN ₂	BaTiN ₂	SrZrN ₂	BaZrN ₂	SrHfN ₂	BaHfN ₂
U(A)	0.04	3.69	0.04	4.32	0.05	4.40
U(M)	0.28	0.29	0.28	0.20	0.13	0.15
U(N)	3.09	2.93	3.09	2.80	3.06	2.80
E_g	1.25	1.02	1.85	1.58	2.21	1.87

Table 3: Elastic constants computed for a set of alkali halides. We determined the U corrections with ACBN0 and compared with the experimental data from Ref. [43] and data from Ref. [44]. Structural parameters and U values are in Table 4.

	C_{ij} (GPa)	ACBN0	PBE	EXP	Model
LiF	C_{11}	113.6	111.5	113.9	104.6
	C_{12}	50.6	46.3	47.6	42.4
	C_{44}	66.2	60.4	63.6	70.8
LiCl	C_{11}	51.5	52.1	60.7	66.6
	C_{12}	23.9	21.8	22.7	20.9
	C_{44}	27.1	24.9	26.9	24.6
NaF	C_{11}	101.1	95.9	108.9	97.6
	C_{12}	24.8	22.2	22.9	22.9
	C_{44}	29.4	26.4	28.9	29.9
NaCl	C_{11}	51.2	49.2	57.3	54.5
	C_{12}	13.4	12.0	11.2	11.2
	C_{44}	13.9	12.3	13.3	11.6
KF	C_{11}	60.9	60.1	75.7	85.0
	C_{12}	14.9	14.1	13.5	14.7
	C_{44}	14.5	13.6	13.3	17.1
KCl	C_{11}	37.6	36.2	40.9	53.8
	C_{12}	6.8	6.3	7.0	5.4
	C_{44}	7.2	6.6	6.2	8.3

Table 4: Structural parameters (in Å) and U corrections (in eV) computed with ACBN0 for selected AX alkali halides using the NaCl structure

	LiF	LiCl	LiBr	NaF	NaCl	NaBr	KF	KCl	KBr
U(A)	0.14	0.04	0.01	0.19	0.02	0.02	0.17	0.03	0.02
U(X)	11.41	6.31	5.49	11.95	6.95	5.97	12.77	7.29	6.27
a_L	3.99	5.06	5.41	4.60	5.57	5.89	5.32	6.24	6.55

5. Installation and data generation

The minimalist style of `AFLOW π` makes it easy to install the software. One command to download the source code from its repository online, and two commands to enter the directory and to install `AFLOW π` as an importable Python package. The `setup.py` in the base directory of the `AFLOW π` source code tree is executed to build and make it importable in Python scripts. All the commands can be run interactively through any Python shell. Currently `AFLOW π` includes three basic submodules: `prep`, `run`, `scfuj` to perform the basic calculations including ACBN0 and the generation of tight-binding hamiltonians. The regular expressions for interacting with a specific electronic structure engine (`pw.x` for instance) are stored separately. In addition, the `plot`, `aflowlib`, `db`, `retr` modules can be used to extract, post-process, and export the computed quantities. The distribution also includes the `pseudo` module for pseudo-potential testing. A detailed description of all the methods is included in the distribution as well as practical examples. A simple interface class allows for easy customization of calculation workflows using methods to instruct `AFLOW π` to prepare the specific tasks.

The construction of a workflow starts with the command `AFLOWpi.prep.init()` which initiates the `AFLOW π` session and requires a `projectname` and the name of the configuration file as input. Specifying sets within a project is optional but highly recommended. `AFLOWpi.prep.init()` returns a session object which has methods to form the calculation sets. When one of the methods of the session object is called, it returns a “`calculation_set`” object. The `calculation_set` object contains methods to construct workflows which add tasks to be done for each calculation in the set. Let assume we plan to generate the PAO-TB hamiltonians for the III-V semiconductors (AlP, GaP, InP, AlAs, GaAs, AlSb, GaSb, and InSb) from the electronic structure generated without and with the ACBN0 approach. In a prototypical reference input file (*e.g.* `mHT_TB.ref`) the labels

for the atomic species are substituted with the variables `_AFLOWPI_A` and `_AFLOWPI_B`. That input combined with the list of chemical substitutions is processed by `AFLOW π` in order to prepare a pipeline of calculations starting with the determination of the ground state potential (`session.scfs`), the structural relaxations (`calcs.vcrelax`), and the generation of the TB hamiltonian (`calcs.tight_binding`) including plots for the densities of states and the band structures. Then the workflow continues with the self-consistent calculation of the Hubbard corrections (`calcs.acbn0`) before recalculating a new TB hamiltonian and new plots. At the end of the user’s `AFLOW π` script, the command `calcs.submit` must be included to run the calculation workflow (see Figure 2). In Figs. 3 and 4, we show plots generated by `AFLOW π` before and after the ACBN0 cycle. It is interesting to pinpoint the corrections of the band gap and the modification of the dispersion relationships (the Hubbard corrections determined running this workflow are in Table 1). All the tight-binding hamiltonians are stored as XML files and can be distributed independently. We stress the fact that the energy dispersion $E_n(\mathbf{k})$ can be computed with *ab initio* accuracy.

Several styles may be used in `AFLOW π` to define the calculations in a set. The first style is based on the cartesian product of two lists (as in Figure 2). The second style, labelled “zip mode”, generates all the entries by taking, in order, the corresponding entries in the keywords lists (see Section 7.2). In addition, calculation sets can be formed from a list of one or more input filenames for the electronic structure engine (these can be completed appropriately with the missing variables).

6. Data retrieval and exporting

`AFLOW π` extracts and monitors data such as energy, structure, atomic positions, total force, total stress, and automatically records it in the `summary.log` file in the calculation set’s `AFLOWpi` directory. Quanti-

ties such as those listed above can be extracted for a given step of the calculation pipeline using the `AFLOWpi.retr` module.

In addition, results can be packaged for the online repository www.aflow.org by adding

```
AFLOWpi.afflowlib.export('export_example',
    set_name='III-V',
    config='./mHT_TB.config')
```

to the workflow.

`AFLOWπ` also includes post-processing routines to compute and plot the radial distribution function, compare the energetics of different distortions, generate CIF files, and perform symmetry analysis.

7. Examples

7.1. Optical (and transport) properties

Layered nitrides with chemical formula AMN_2 were studied earlier from the point of view of thermoelectric potentials and superconductivity [45–50]. As an example we analyze the same chemical space (A =Sr, Ba and M =Ti, Zr, Hf) from the point of view of the electronic transport in the constant relaxation time approximation and the optical properties within the independent electronic approximation. Using well converged basis sets and k-space sampling (see Ref. [45] for computational details), we have determined the electrical conductivity, the Seebeck coefficient, and the the complex dielectric functions with and without ACBN0. The workflow is in Figure 5.

```
import AFLOWpi
session = AFLOWpi.prep.init('Transport', 'Si',
    ↪ config='./transport.config')
calcs = session.from_file('transport.in')
calcs.vcrelax()
# get PAO-TB hamiltonian
calcs_TB = calcs.tight.binding()
# optical and transport properties
calcs_TB.epsilon()
calcs_TB.transport(temperature=[300,400])
# plot optical and transport properties
calcs_TB.plot.transport()
calcs_TB.plot.epsilon()
calcs.submit()
```

Figure 5: Workflow to compute optical (and transport) properties for layered nitrides with chemical formula AMN_2 . The complex dielectric constant is computed within the independent particle approximation with and without the ACBN0 correction. The comments show how to compute the electronic transport as discussed in Ref. [36].

We report data for AMN_2 compounds in Table 2 and in Figure 6. Among the AMN_2 compositions, the compounds containing Ti have absorption edges below the Shockley-Queisser limit of 1.34 eV; the compounds containing Hf and Zr exceed such a limit but seem potentially interesting as end points of $A(\text{Ti},M)N_2$ alloys with M =Zr and Hf where tuning of the band gap can lead to improved photovoltaics potentials.

7.2. Phonons and elastic constants

In order to show how the calculations of phonon dispersions and of the elastic constants can be performed in `AFLOWπ` we analyzed a small chemical space of alkali halides AX with (A =Li, Na, K and X =F, Cl, Br) with the NaCl and the CsCl structures. The workflow is in Figure 7: notice the use of the zip mode to specify the chemical space and the use of two reference files for the NaCl and CsCl structure, respectively. Elastic constants are computed solely on the NaCl structure.

Under normal conditions, alkali halides have NaCl or B1-type crystal structure, but they transform under moderate pressure to the CsCl or B2-type structure. There has been much work done on thermal conductivity during the phase transition of alkali halides: Averkin [51] found that the phase transition completely eliminates the influence of the elastic anisotropy on the thermal conductivity, as well as its dependence on pressure. In that paper an empirical relation was also established between elastic anisotropy and thermal conductivity. However, Slack [52] concluded that there is no need to invoke the elastic anisotropy to explain thermal conductivity differences. He noticed that NaCl structure compounds have a higher thermal conductivity ratio than CsCl ones by about a factor of two and conjectured a large increase in the Grüneisen parameter as the co-ordination number changes from six to eight. Our results indicate, however, that a major contribution to the change in the thermal conductivity is related to the average sound velocity reduction in CsCl structure with respect to the NaCl structure. In Figure 8 we report, as an example, phonon dispersions automatically generated by `AFLOWπ` using a finite difference method. We did not observe significant changes using the ACBN0 approach with respect to the PBE calculations, and we choose to report selected ACBN0 data as an example.

The elastic constants of alkali halides are important as they are closely related to the ionic binding force in the crystal [44]. There are several phenomenological

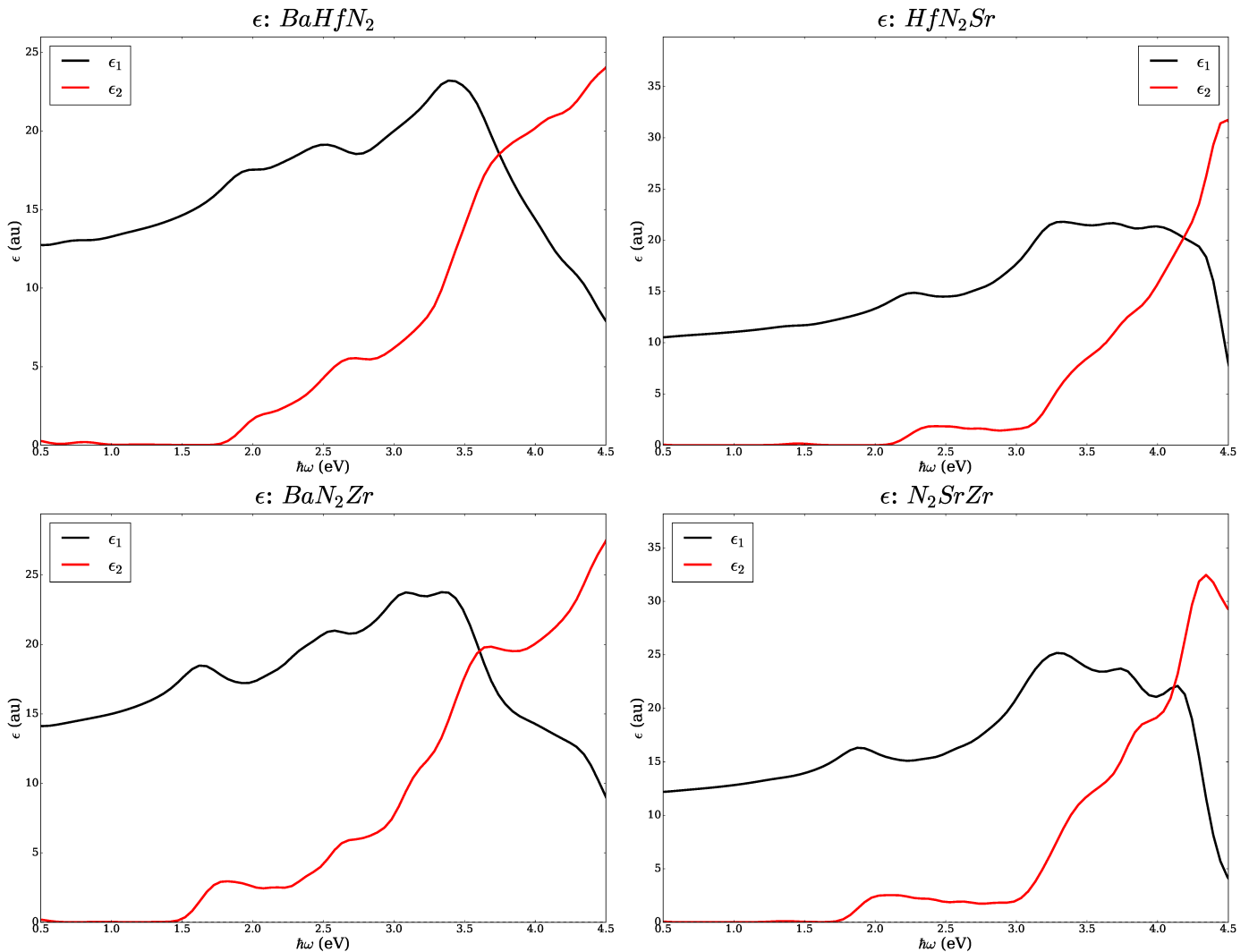


Figure 6: Complex dielectric constant generated by AFLOW π using the tight-binding hamiltonian as in Ref. [36] for AMN_2 compounds with $A=\text{Ba, Sr}$ and $M=\text{Hf, Zr}$. The titanates exhibit absorption edges just above 1eV. The workflow in Figure 5 has been used to obtain this result. Additional data including the correction to the band gap provided by ACBN0 are in Table 2.

models to determine the elastic constants: in the central force model [53], the elastic constants are given by the sum of the terms due to long range and short range central forces. This model leads to a Cauchy relation $C_{12} = C_{44}$, which is not true experimentally. The degree to which this Cauchy relation is not satisfied is a measure of the importance of many-body forces in lattice dynamics of the alkali halide which Lowdin [54] has found to exist. The breathing shell model [55] includes many-body forces and makes a prediction $C_{12} < C_{44}$ in alkali halides. Mahan [44] studied the effect of induced polarization on phonon modes at long wavelength and introduced a polarization term in the central force model predicting $C_{12} > C_{44}$. Experimentally the deviation is found in both directions. We report in Table 3 the comparison between first prin-

ciples calculations with and without ACBN0, experimental data, and results from Ref. [44]. Our results improve the agreement with experimental data with respect to the model in Ref. [44] and indicate the sensitivity of the elastic constants to the choice of the computational method.

8. Conclusions

We have presented here a minimalist high-throughput software infrastructure that, in addition to organizing and controlling the computational workflow, facilitates the generation of tight-binding Hamiltonians to represent accurately the electronic properties with *ab initio* accuracy. AFLOW π exploits the tight-binding formal-

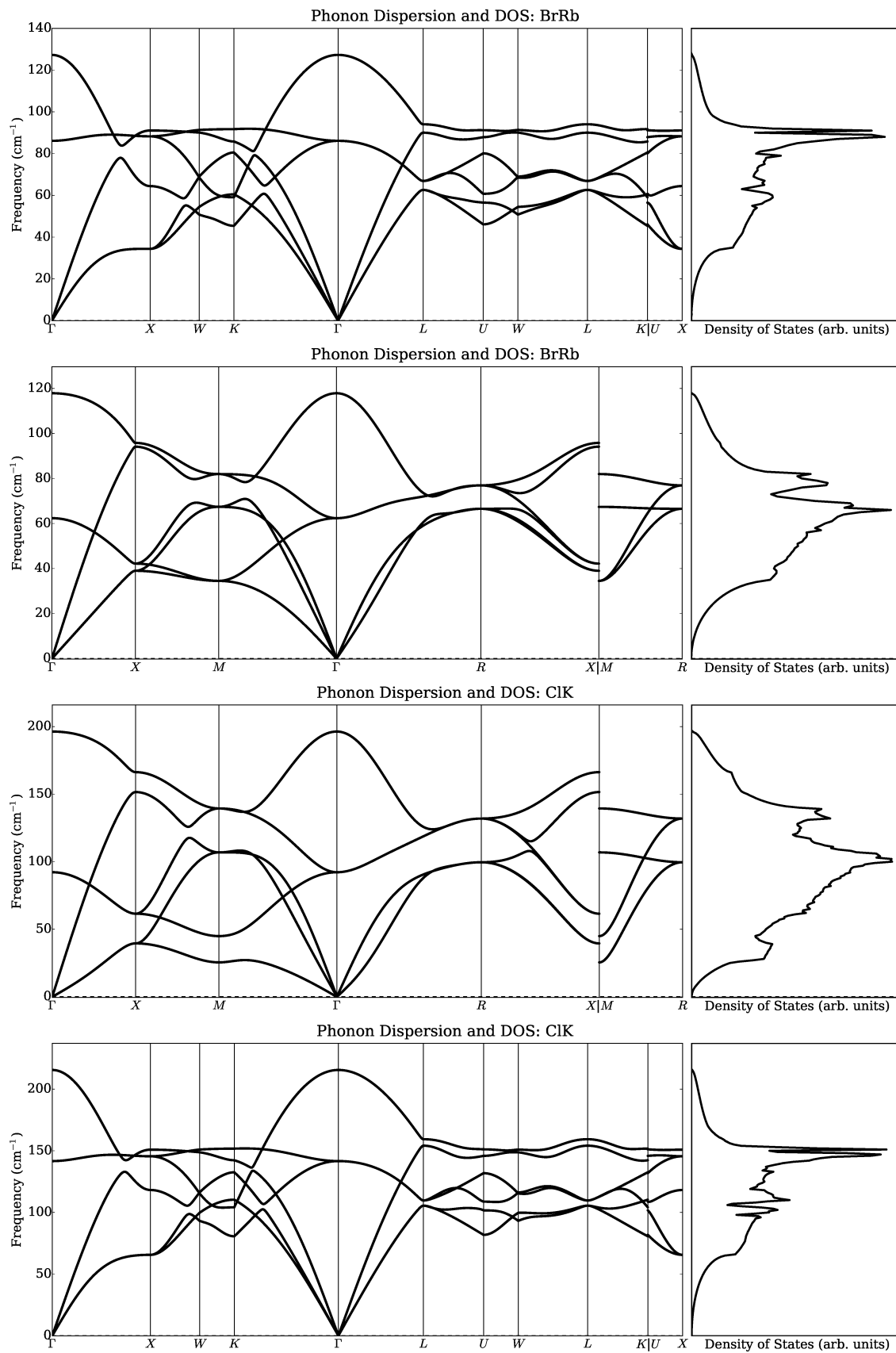


Figure 8: Phonon branches and vibrational density of states for CsCl and NaCl structure of RbBr and KCl at equilibrium lattice parameters. We used the workflow in Figure 7.

```

import AFLOWpi
# Create AFLOWpi session
ses_CsCl = AFLOWpi.prep.init('Halide', 'CsCl',
                             config='./phonon.config')
ses_NaCl = AFLOWpi.prep.init('Halide', 'NaCl',
                             config='./phonon.config')
# Generate a calculation set from a reference input file
allvars={"_AFLOWPI_A.":("Li","Na","K"),
         "_AFLOWPI_B.":("F","Cl","Br","I"),}
CsCl = ses_CsCl.scfs(allvars, 'CsCl.ref',)
NaCl = ses_NaCl.scfs(allvars, 'NaCl.ref',)
#do a vc-relax
CsCl.vcrelax()
NaCl.vcrelax()
# change the thresholds in the input files
# changing input for a step are called after
# the call for the step. the change.calcs
# below affect the vc-relax above
CsCl.change_input("&control","etot_conv_thr","1.0D-5")
CsCl.change_input("&control","forc_conv_thr","1.0D-4")
CsCl.change_input("&electrons","conv_thr","1.0D-16")
NaCl.change_input("&control","etot_conv_thr","1.0D-5")
NaCl.change_input("&control","forc_conv_thr","1.0D-4")
NaCl.change_input("&electrons","conv_thr","1.0D-16")
#do another relax just to be safe
CsCl.vcrelax()
NaCl.vcrelax()
#do phonon calculations with 2x2x2 supercell
CsCl.phonon(mult_jobs=True,nrx1=3,nrx2=3,nrx3=3,innx=2,
            ↪ de=0.003,LOTO=True,field_strength=0.003,
            ↪ disp_sym=True,atom_sym=False,)
NaCl.phonon(mult_jobs=True,nrx1=3,nrx2=3,nrx3=3,innx=2,
            ↪ de=0.003,LOTO=True,field_strength=0.003,
            ↪ disp_sym=True,atom_sym=False,)
#plot the phonons
NaCl.plot.phonon(postfix='NaCl')
CsCl.plot.phonon(postfix='CsCl')
# do the elastic constants with the ElaStic Package
# Install: http://exciting-code.org/elastic
NaCl.elastic(mult_jobs=False,num_dist=10,eta_max=0.001)
#do acbn0 to do phonon with self-consistently
#determined Hubbard U. Changing the cell while
#converging the U should be done with caution.
NaCl.scfuj(relax='vc-relax')
CsCl.scfuj(relax='vc-relax')
NaCl.elastic(mult_jobs=False,num_dist=10,eta_max=0.001)
NaCl.phonon(mult_jobs=True,nrx1=3,nrx2=3,nrx3=3,innx=2,
            ↪ de=0.003,LOTO=True,field_strength=0.003,
            ↪ disp_sym=True,atom_sym=False,)
CsCl.phonon(mult_jobs=True,nrx1=3,nrx2=3,nrx3=3,innx=2,
            ↪ de=0.003,LOTO=True,field_strength=0.003,
            ↪ disp_sym=True,atom_sym=False,)
CsCl.plot.phonon(postfix='CsCl.acbn0')
NaCl.plot.phonon(postfix='NaCl.acbn0')
# submit the calcs to run
calcs.submit()

```

Figure 7: Workflow to compute phonon dispersion with and without ACBN0 for few alkali halides in NaCl and CsCl structure. For the NaCl set of calculations also the elastic constants are computed and compared with the experimental data in Table 3.

ism to compute electronic transport and optical properties within the independent particle approximation. Three examples were used to illustrate the features of AFLOW π (1) the calculation of the band structure of III-V semiconductors with and without the ACBN0 correction, (2) the study of the complex dielectric constant of AMN_2 (A =Sr, Ba and M =Ti, Zr, Hf), and (3) the phonon dispersions and the elastic constants of binary halides (AX with A =Li, Na, K and X =F, Cl, Br). We found that $A(\text{Ti},M)N_2$ alloys with M =Zr and Hf may have potential for photovoltaic applications. In addition, we justified the dramatic changes in thermal conductivity occurring at the transition between NaCl and CsCl structure in terms of sound velocity. The code, including the examples presented in this paper, is available at www.aflow.org/aflowpi.

9. Acknowledgments

We are grateful to the High Performance Computing Center at Michigan State University and the Texas Advanced Computing Center at the University of Texas Austin. The members of the AFLOW Consortium (<http://www.aflow.org>) acknowledge support by DOD-ONR (N00014-13-1-0635, N00014-11-1-0136, N00014-15-1-2863). The authors also acknowledge Duke University — Center for Materials Genomics — and the CRAY corporation for computational support.

Bibliography

- [1] R. Potyrailo, K. Rajan, K. Stoewe, I. Takeuchi, B. Chisholm, and H. Lam, *Combinatorial and High-Throughput Screening of Materials Libraries: Review of State of the Art*, ACS Combinatorial Science **13**, 579–633 (2011).
- [2] K. Rajan, *Combinatorial Materials Sciences: Experimental Strategies for Accelerated Knowledge Discovery*, Annual Review of Materials Research **38**, 299–322 (2008).
- [3] I. Takeuchi, C. J. Long, O. O. Famodu, M. Murakami, J. Hattrick-Simpers, G. W. Rubloff, M. Stukowski, and K. Rajan, *Data management and visualization of x-ray diffraction spectra from thin film ternary composition spreads*, Review of Scientific Instruments **76**, 062223 (2005).
- [4] A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M. J. Kramer, C. Long, and I. Takeuchi, *On-the-fly machine-learning*

- for high-throughput experiments: search for rare-earth-free permanent magnets, *Scientific Reports* **4**, 6367 EP– (2014).
- [5] S. Curtarolo, G. L. W. Hart, M. Buongiorno Nardelli, N. Mingo, S. Sanvito, and O. Levy, *The high-throughput highway to computational materials design*, *Nat. Mater.* **12**, 191–201 (2013).
- [6] D. Morgan, G. Ceder, and S. Curtarolo, *High-throughput and data mining with ab initio methods*, *Measurement Science and Technology* **16**, 296 (2005).
- [7] S. Curtarolo, D. Morgan, and G. Ceder, *Accuracy of ab initio methods in predicting the crystal structures of metals: A review of 80 binary alloys*, *Calphad* **29**, 163–211 (2005).
- [8] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G. L. Chiarotti, M. Cococcioni, I. Dabo, A. Dal Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A. P. Seitsonen, A. Smogunov, P. Umari, and R. M. Wentzcovitch, *QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials*, *J. Phys.: Condens. Matter* **21**, 395502 (2009).
- [9] G. Kresse and J. Furthmüller, *Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set*, *Comput. Mater. Sci.* **6**, 15–50 (1996).
- [10] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. a. Persson, *The Materials Project: A materials genome approach to accelerating materials innovation*, *APL Mater.* **1**, 011002 (2013).
- [11] S. Curtarolo, W. Setyawan, S. Wang, J. Xue, K. Yang, R. H. Taylor, L. J. Nelson, G. L. W. Hart, S. Sanvito, M. Buongiorno Nardelli, N. Mingo, and O. Levy, *AFLOWLIB.ORG: A distributed materials properties repository from high-throughput ab initio calculations*, *Comput. Mater. Sci.* **58**, 227–235 (2012).
- [12] M. Scheffler, C. Draxl, and Computer Center of the Max-Planck Society, Garching, *The NoMaD Repository*, <http://nomad-repository.eu> (2014).
- [13] P. Scherpelz, M. Govoni, I. Hamada, and G. Galli, *Implementation and Validation of Fully Relativistic GW Calculations: Spin-Orbit Coupling in Molecules, Nanocrystals, and Solids*, *Journal of Chemical Theory and Computation* **12**, 3523–3544 (2016).
- [14] S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder, *Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis*, *Comput. Mater. Sci.* **68**, 314–319 (2013).
- [15] X. Gonze, J.-M. Beuken, R. Caracas, F. Detraux, M. Fuchs, G.-M. Rignanese, L. Sindic, M. Verstraete, G. Zerah, F. Jollet, M. Torrent, A. Roy, M. Mikami, P. Ghosez, J.-Y. Raty, and D. C. Allan, *First-principles computation of material properties: the ABINIT software project*, *Comput. Mater. Sci.* **25**, 478–492 (2002).
- [16] G. Pizzi, A. Cepellotti, R. Sabatini, N. Marzari, and B. Kozinsky, *AiiDA: automated interactive infrastructure and database for computational science*, *Comput. Mater. Sci.* **111**, 218–230 (2016).
- [17] S. R. Bahn and K. W. Jacobsen, *An object-oriented scripting interface to a legacy electronic structure code*, *Comput. Sci. Eng.* **4**, 56–66 (2002).
- [18] D. D. Landis, J. S. Hummelshøj, S. Nestorov, J. Greeley, M. Dulak, T. Bligaard, J. K. Nørskov, and K. W. Jacobsen, *The Computational Materials Repository*, *Comput. Sci. Eng.* **14**, 51–57 (2012).
- [19] S. Kirklin, J. E. Saal, B. Meredig, A. Thompson, J. W. Doak, M. Aykol, S. Rühl, and C. Wolverton, *The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies*, *Npj Computational Materials* **1**, 15010 EP– (2015).
- [20] S. Curtarolo, W. Setyawan, G. L. W. Hart, M. Jahnátek, R. V. Chepulskii, R. H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M. J. Mehl, H. T. Stokes, D. O. Demchenko, and D. Morgan, *AFLOW: An automatic framework for high-throughput materials discovery*, *Comput. Mater. Sci.* **58**, 218–226 (2012).
- [21] W. Setyawan and S. Curtarolo, *High-throughput electronic band structure calculations: Challenges and tools*, *Comput. Mater. Sci.* **49**, 299–312 (2010).
- [22] K. Yang, C. Oses, and S. Curtarolo, *Modeling Off-Stoichiometry Materials with a High-Throughput Ab-Initio Approach*, *Chem. Mater.* **28**, 6484–6492 (2016).
- [23] O. Levy, M. Jahnátek, R. V. Chepulskii, G. L. W. Hart, and S. Curtarolo, *Ordered Structures in Rhenium Binary Alloys from First-Principles Calculations*, *J. Am. Chem. Soc.* **133**, 158–163 (2011).
- [24] O. Levy, G. L. W. Hart, and S. Curtarolo, *Structure maps for hcp metals from first-principles calculations*, *Phys. Rev. B* **81**, 174106 (2010).
- [25] O. Levy, G. L. W. Hart, and S. Curtarolo, *Uncovering Compounds by Synergy of Cluster Expansion and High-Throughput Methods*, *J. Am. Chem. Soc.* **132**, 4830–4833 (2010).

- [26] G. L. W. Hart, S. Curtarolo, T. B. Massalski, and O. Levy, *Comprehensive Search for New Phases and Compounds in Binary Alloy Systems Based on Platinum-Group Metals, Using a Computational First-Principles Approach*, Phys. Rev. X **3**, 041035 (2013).
- [27] O. Isayev, D. Fourches, E. N. Muratov, C. Oses, K. Rasch, A. Tropsha, and S. Curtarolo, *Materials Cartography: Representing and Mining Materials Space Using Structural and Electronic Fingerprints*, Chem. Mater. **27**, 735–743 (2015).
- [28] P. Nath, J. J. Plata, D. Usanmaz, R. Al Rahal Al Orabi, M. Fornari, M. Buongiorno Nardelli, C. Toher, and S. Curtarolo, *High-throughput prediction of finite-temperature properties using the quasi-harmonic approximation*, Comput. Mater. Sci. **125**, 82–91 (2016).
- [29] C. Toher, J. J. Plata, O. Levy, M. de Jong, M. D. Asta, M. Buongiorno Nardelli, and S. Curtarolo, *High-throughput computational screening of thermal conductivity, Debye temperature, and Grüneisen parameter using a quasiharmonic Debye Model*, Phys. Rev. B **90**, 174107 (2014).
- [30] R. H. Taylor, F. Rose, C. Toher, O. Levy, K. Yang, M. Buongiorno Nardelli, and S. Curtarolo, *A RESTful API for exchanging Materials Data in the AFLOWLIB.org consortium*, Comput. Mater. Sci. **93**, 178–192 (2014).
- [31] C. E. Calderon, J. J. Plata, C. Toher, C. Oses, O. Levy, M. Fornari, A. Natan, M. J. Mehl, G. L. W. Hart, M. Buongiorno Nardelli, and S. Curtarolo, *The AFLOW standard for high-throughput materials science calculations*, Comput. Mater. Sci. **108 Part A**, 233–238 (2015).
- [32] L. A. Agapito, A. Ferretti, A. Calzolari, S. Curtarolo, and M. Buongiorno Nardelli, *Effective and accurate representation of extended Bloch states on finite Hilbert spaces*, Phys. Rev. B **88**, 165127 (2013).
- [33] L. A. Agapito, S. Ismail-Beigi, S. Curtarolo, M. Fornari, and M. Buongiorno Nardelli, *Accurate tight-binding Hamiltonian matrices from ab initio calculations: Minimal basis sets*, Phys. Rev. B **93**, 035104 (2016).
- [34] L. A. Agapito, M. Fornari, D. Ceresoli, A. Ferretti, S. Curtarolo, and M. Buongiorno Nardelli, *Accurate Tight-Binding Hamiltonians for 2D and Layered Materials*, Phys. Rev. B **93**, 125137 (2016).
- [35] L. A. Agapito, S. Curtarolo, and M. Buongiorno Nardelli, *Reformulation of DFT + U as a Pseudohybrid Hubbard Density Functional for Accelerated Materials Discovery*, Phys. Rev. X **5**, 011006 (2015).
- [36] P. D’Amico, L. Agapito, A. Catellani, A. Ruini, S. Curtarolo, M. Fornari, M. Buongiorno Nardelli, and A. Calzolari, *Accurate ab initio tight-binding Hamiltonians: Effective tools for electronic transport and optical spectroscopy from first principles*, Phys. Rev. B **94**, 165166 (2016).
- [37] R. Golezorkhtabar, P. Pavone, J. Spitaler, P. Puschnig, and C. Draxl, *ElaStic: A tool for calculating second-order elastic constants from first principles*, Comput. Phys. Commun. **184**, 1861–1873 (2013).
- [38] H. T. Stokes and D. M. Hatch, *FINDSYM: program for identifying the space-group symmetry of a crystal*, J. Appl. Cryst. **38**, 237–238 (2005).
- [39] M. Buongiorno Nardelli, *PAOπ: a python utility to construct and operate on Hamiltonians from the Projections of DFT wavefunctions on Atomic Orbital bases*, in preparation .
- [40] W. W. Peterson and D. T. Brown, *Cyclic Codes for Error Detection*, Proceedings of the IRE **49**, 228–235 (1961).
- [41] P. Gopal, M. Fornari, S. Curtarolo, L. A. Agapito, L. S. I. Liyanage, and M. Buongiorno Nardelli, *Improved predictions of the physical properties of Zn- and Cd-based wide band-gap semiconductors: A validation of the ACBN0 functional*, Phys. Rev. B **91**, 245202 (2015).
- [42] M. Shishkin and H. Sato, *Self-consistent parametrization of DFT + U framework using linear response approach: Application to evaluation of redox potentials of battery cathodes*, Phys. Rev. B **93**, 085135 (2016).
- [43] J. T. Lewis, A. Lehoczky, and C. V. Briscoe, *Elastic Constants of the Alkali Halides at 4.2K*, Phys. Rev. **161**, 877–887 (1967).
- [44] G. D. Mahan, *Elastic constants of alkali halides: Multipole expansion*, Phys. Rev. B **29**, 5849–5858 (1984).
- [45] R. Al Rahal Al Orabi, E. Orisakwe, D. Wee, B. Fontaine, R. Gautier, J.-F. Halet, and M. Fornari, *Prediction of high thermoelectric potential in AMN2 layered nitrides: electronic structure, phonons, and anharmonic effects*, J. Mater. Chem. A **3**, 9945–9954 (2015).
- [46] R. L. Henderson, *Job scheduling under the Portable Batch System* (Springer Berlin Heidelberg, Berlin, Heidelberg, 1995), pp. 279–294.
- [47] G. Farault, R. Gautier, C. F. Baker, A. Bowman, and D. H. Gregory, *Crystal Chemistry and Electronic Structure of the Metallic Ternary Nitride, SrTiN₂*, Chem. Mater. **15**, 3922–3929 (2003).

- [48] D. H. Gregory, M. G. Barker, P. P. Edwards, M. Slaski, and D. J. Siddons, *Synthesis, Structure, and Magnetic Properties of the New Ternary Nitride BaHfN₂ and of the BaHf_{1-x}Zr_xN₂ Solid Solution*, Journal of Solid State Chemistry **137**, 62–70 (1998).
- [49] I. Ohkubo and T. Mori, *Two-Dimensional Layered Complex Nitrides as a New Class of Thermoelectric Materials*, Chem. Mater. **26**, 2532–2536 (2014).
- [50] D. H. Gregory, M. G. Barker, P. P. Edwards, and D. J. Siddons, *Synthesis and Structure of the New Ternary Nitride SrTiN₂*, Inorg. Chem. **37**, 3775–3778 (1998).
- [51] A. A. Averkin, Y. A. Logachev, A. V. Petrov, and N. S. Tsyapkina, *Thermoconductivity of alkali-halide crystals under pressure*, Fizika Tverdogo Tela **19**, 1692–1701 (1977).
- [52] G. A. Slack and R. G. Ross, *Thermal conductivity under pressure and through phase transitions in solid alkali halides. II. Theory*, J. Phys. C: Solid State Phys. **18**, 3957 (1985).
- [53] Y. Michihiro, K. Itsuki, S. Endou, K. Nakamura, and T. Ohno, *Effects of the multipole polarization on the elastic constants and ionic conductivity in cubic structure crystals*, Solid State Ion. **181**, 572–577 (2010).
- [54] P.-O. Löwdin, *Quantum Theory of Many-Particle Systems. II. Study of the Ordinary Hartree-Fock Approximation*, Phys. Rev. **97**, 1490–1508 (1955).
- [55] U. Schröder, *A new model for lattice dynamics (“breathing shell model”)*, Solid State Commun. **4**, 347–349 (1966).