The Journal of Open Source Software

doped: Python toolkit for robust and repeatable charged defect supercell calculations

Seán R. Kavanagh [1,2¶], **Alexander G. Squires** [3], **Adair Nicolson** [2], **Irea Mosquera-Lois** [1], **Alex M. Ganose** [4], **Bonan Zhu** [2], **Katarina Brlec** [2], **Aron Walsh** [1], **and David O. Scanlon** [3]

1 Thomas Young Centre and Department of Materials, Imperial College London, United Kingdom **2** Thomas Young Centre and Department of Chemistry, University College London, United Kingdom **3** School of Chemistry, University of Birmingham, Birmingham, United Kingdom **4** Department of Chemistry, Imperial College London, London, United Kingdom ¶ Corresponding author

DOI: 10.21105/joss.06433

Software

- Review
- Repository
- Archive

Editor: Rachel Kurchin

Reviewers:

- @yuan-gist
- @atimmins7

Submitted: 27 February 2024
Published: 15 April 2024

Summary

Defects are a universal feature of crystalline solids, dictating the key properties and performance of many functional materials. Given their crucial importance yet inherent difficulty in measuring experimentally, computational methods (such as DFT and ML/classical force-fields) are widely used to predict defect behaviour at the atomic level and the resultant impact on macroscopic properties. Here we report doped, a Python package for the generation, pre-/post-processing, and analysis of defect supercell calculations. doped has been built to implement the defect simulation workflow in an efficient and user-friendly – yet powerful and fully-flexible – manner, with the goal of providing a robust general-purpose platform for conducting reproducible calculations of solid-state defect properties.

Statement of need

The materials science sub-field of computational defect modelling has seen considerable growth in recent years, driven by the crucial importance of these species in functional materials and the major advances in computational methodologies and resources facilitating their accurate simulation. Software which enables researchers to efficiently and accurately perform these calculations, while allowing for in-depth target analyses of the resultant data, is thus of significant value to the community. Indeed there are many critical stages in the computational workflow for defects, which when performed manually not only consume significant researcher time and effort but also leave room for human error – particularly for newcomers to the field. Moreover, there are growing efforts to perform high-throughput investigations of defects in solids (Broberg et al., 2023; Xiong et al., 2023; Yuan et al., 2024), necessitating robust, user-friendly, and efficient software implementing this calculation workflow.

Given this importance of defect simulations and the complexity of the workflow, a number of software packages have been developed with the goal of managing pre- and post-processing of defect calculations, including work on the HADES/METADISE codes from the 1970s (Parker et al., 2004), to more recent work from Kumagai et al. (2021), Broberg et al. (2018), Shen & Varley (2024), Neilson & Murphy (2022), Arrigoni & Madsen (2021), Goyal et al. (2017), M. Huang et al. (2022), Péan et al. (2017) and Naik & Jain (2018).[1] While each of these codes have their strengths, they do not include the full suite of functionality provided by doped – some of which is discussed below – nor adopt the same focus on user-friendliness (along with

[1]Some of these packages are no longer maintained, not compatible with high-throughput architectures, and/or are closed-source/commercial packages.

sanity-checking warnings and error catching) and efficiency with full flexibility and wide-ranging functionality, targeting expert-level users and newcomers to the field alike.

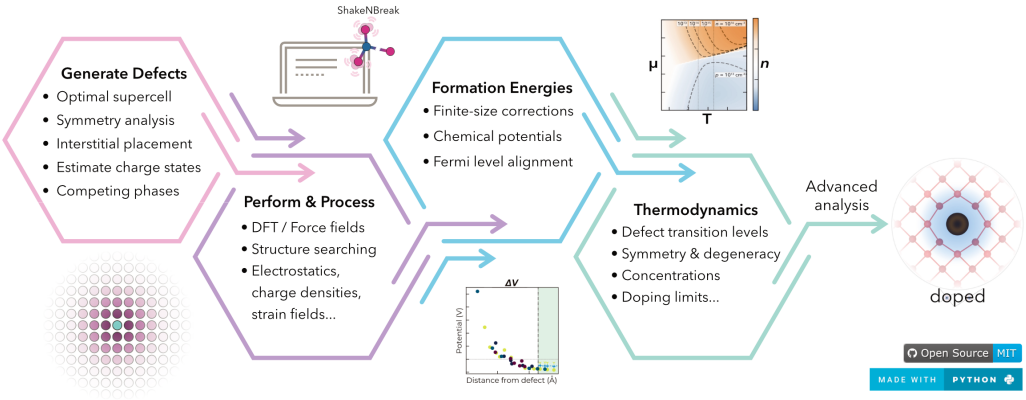

Figure 1: Schematic workflow of a computational defect investigation using doped.

doped

doped is a Python package for the generation, pre-/post-processing, and analysis of defect supercell calculations, as depicted in Figure 1. The design philosophy of doped has been to implement the defect simulation workflow in an efficient, reproducible, and user-friendly – yet powerful and fully-customisable – manner, combining reasonable defaults with full user control for each parameter in the workflow. As depicted in Figure 1, the core functionality of doped is the generation of defect supercells and competing phases, writing calculation input files, parsing calculation outputs, and analysing/plotting defect-related properties. This functionality and recommended usage of doped is demonstrated in the tutorials on the documentation website.

The Journal of Open Source Software

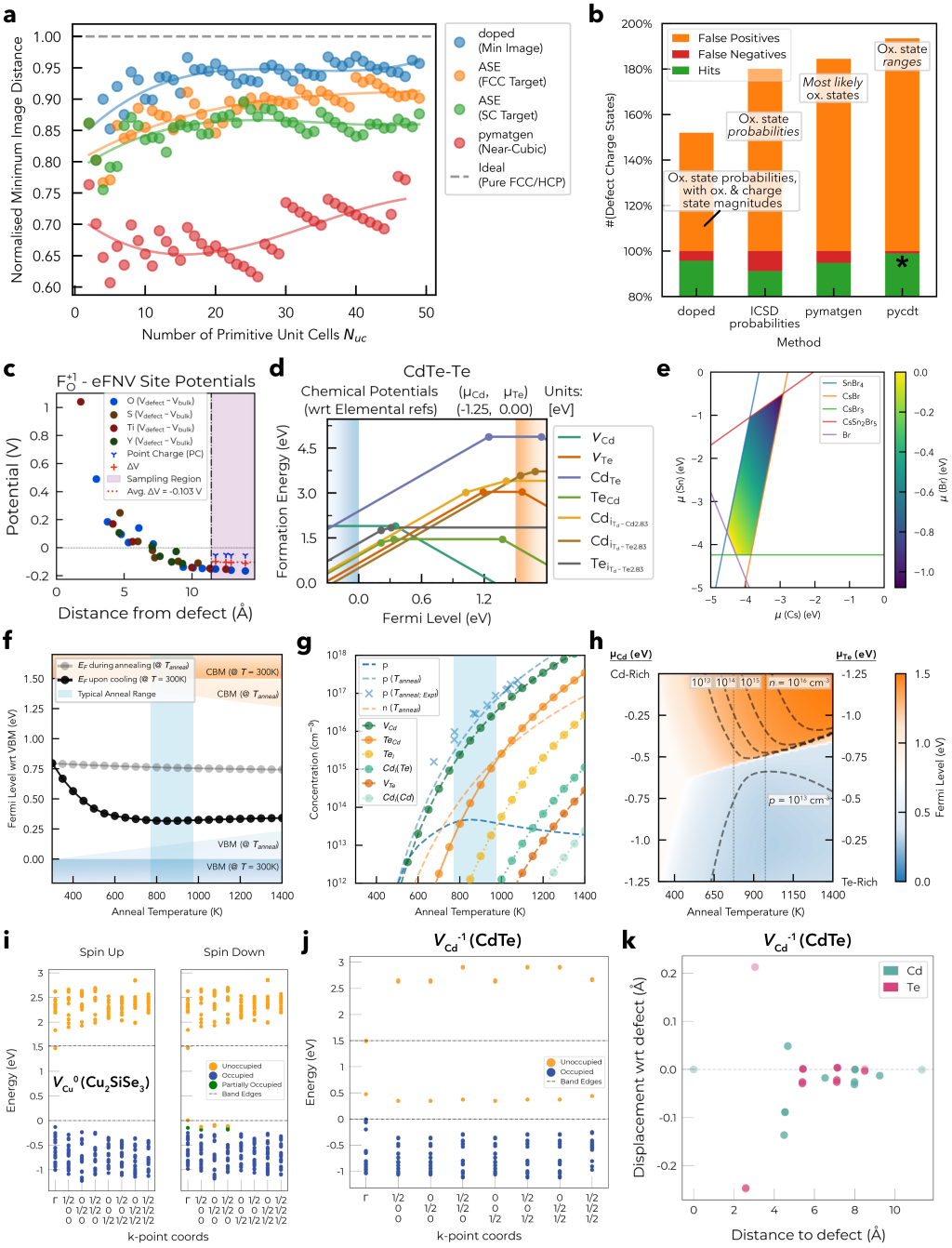

Figure 2: Performance and example outputs from doped. **(a)** Average minimum periodic image distance, normalised by the ideal image distance (i.e. for a close-packed face-centred cubic (FCC) cell), vs. number of unit cells for supercell generation algorithms in doped, ASE, and pymatgen. "SC" = simple cubic and "HCP" = hexagonal close-packed. **(b)** Average performance of various charge state estimation routines. "ICSD probabilities" refers to a model based oxidation state probabilities, as given by their occurrence in the ICSD database. Asterisk indicates that pyCDT "false *negatives*" are underestimated as the majority of this test set used the pyCDT charge state ranges. "Ox. state" = oxidation state. Example **(c)** Kumagai-Oba (eFNV) finite-size correction plot, **(d)** defect formation energy diagram, **(e)** chemical potential / stability region, **(f)** Fermi level vs. annealing temperature, **(g)** defect/carrier concentrations vs. annealing temperature and **(h)** Fermi level / carrier concentration heatmap plots from doped. Automated plots of single-particle eigenvalues from DFT supercell calculations for **(i)** V_{Cu}^{0} in Cu_2SiSe_3 and **(j)** V_{Cd}^{-1} in CdTe. **(k)** Automated site displacement analysis, plotting atomic displacements with respect to the defect site against distance to the defect site, for V_{Cd}^{-1} in CdTe. Data and code to reproduce these plots is provided in the docs/JOSS folder of the doped GitHub repository.

Some key advances of doped include:

- **Supercell Generation:** When choosing a simulation supercell for charged defects in materials, we typically want to maximise the minimum distance between periodic images of the defect (to reduce finite-size errors) while keeping the supercell to a tractable number of atoms/electrons to calculate. Common approaches are to choose a near-cubic integer expansion of the unit cell (Ong et al., 2013), or to use a cell shape metric to search for optimal supercells (Larsen et al., 2017). Building on these and instead integrating an efficient algorithm for calculating minimum image distances, doped directly optimises the supercell choice for this goal – often identifying non-trivial 'root 2'/'root 3' type supercells. As illustrated in Figure 2a, this leads to a significant reduction in the supercell size (and thus computational cost) required to achieve a threshold minimum image distance.
  - Over a test set of simple cubic, trigonal, orthorhombic, monoclinic and face-centred cubic unit cells, the doped algorithm is found to give mean improvements of 35.2%, 9.1% and 6.7% in the minimum image distance for a given (maximum) number of unit cells as compared to the pymatgen cubic supercell algorithm, the ASE optimal cell shape algorithm with simple-cubic target shape, and ASE with FCC target shape respectively – in the range of 2-20 unit cells. For 2-50 unit cells (for which the mean values across this test set are plotted in Figure 2a), this becomes 36.0%, 9.3% and 5.6% respectively. Given the approximately cubic scaling of DFT computational cost with the number of atoms, these correspond to significant reductions in cost (~20-150%).
  - As always, the user has full control over supercell generation in doped, with the ability to specify/adjust constraints on the minimum image distance, number of atoms or transformation matrix, or to simply provide a pre-generated supercell if desired.
- **Charge-state Estimation:** Defects in solids can adopt various electronic charge states. However, the set of stable charge states for a given defect is typically not known *a priori*, so one must choose a set of *possible* defect charge states to calculate – usually relying on some form of chemical intuition. In this regard, extremal defect charge states that are calculated but do not end up being stable can be considered 'false positives' or 'wasted' calculations,[2] while charge states which are stable but were not calculated can be considered 'false negatives' or 'missed' calculations. doped builds on other routines which use known elemental oxidation states to additionally account for oxidation state *probabilities*, the electronic state of the host crystal and charge state magnitudes. Implementing these features in a simple cost function, we find a significant improvement in terms of both efficiency (reduced false positives) and completeness (reduced false negatives) for this charge state estimation, as shown in Figure 2b.[3]

  Again, this step is fully-customisable. The user can tune the probability threshold at which to include charge states or manually specify defect charge states. All probability factors computed are available to the user and saved to the defect JSON files for full reproducibility.
- **Efficient Competing Phase Selection:** Elemental chemical potentials (a key term in the defect formation energy) are limited by the secondary phases which border the host compound on the phase diagram. These bordering phases are known as competing phases, and their total energies must be calculated to determine the chemical potential limits. Only the elemental reference phases and compounds which border the host on the phase diagram need to be calculated, rather than the full phase diagram.

[2] Note that *unstable* defect charge states which are intermediate between *stable* charge states (e.g. X^0 for a defect X with a (+1/-1) negative-U level) should still be calculated and are *not* considered false positives.

[3] Given sufficient data, a machine learning model could likely further improve the performance of this charge state estimation.

doped aims to improve the efficiency of this step by querying the Materials Project database (containing both experimentally-measured and theoretically-predicted crystal structures), and pulling only compounds which *could border the host material* within a user-specified error tolerance for the semi-local DFT database energies (0.1 eV/atom by default), along with the elemental reference phases. The necessary *k*-point convergence step for these compounds is also implemented in a semi-automated fashion to expedite this process.

  - With the parsed chemical potentials in doped, the user can easily select various X-poor/rich chemical conditions, or scan over a range of chemical potentials (growth conditions) as shown in Figure 2e,h.

- **Automated Symmetry & Degeneracy Handling:** doped automatically determines the point symmetry of both initial (un-relaxed) and final (relaxed) defect configurations, and computes the corresponding orientational (and spin) degeneracy factors. This functionality is also offered in the form of standalone functions which do not require the defect calculations to have been generated/parsed with doped. This is a key pre-factor in the defect concentration equation:

$$N_D = gN_s \exp(-E_f/k_BT) \tag{1}$$

where g is the product of all degeneracy factors, N_s is the concentration of lattice sites for that defect, E_f is the defect formation energy and N_D is the defect concentration. g can affect predicted defect/carrier concentrations by up to two or three orders of magnitude (Kavanagh, Scanlon, et al., 2022; Mosquera-Lois, Kavanagh, Klarbring, et al., 2023), and is often overlooked in defect calculations, partly due to the (previous) requirement of significant manual effort and knowledge of group theory.

- **Automated Compatibility Checking:** When parsing defect calculations, doped automatically checks that calculation parameters which could affect the defect formation energy (e.g. *k*-point grid, energy cutoff, pseudopotential choice, exchange fraction, Hubbard U etc.) are consistent between the defect and reference calculations. This is a common source of accidental error in defect calculations, and doped provides informative warnings if any inconsistencies are detected.

- **Thermodynamic Analysis:** doped provides a suite of flexible tools for the analysis of defect thermodynamics, including formation energy diagrams (Figure 2d), equilibrium & non-equilibrium Fermi level solving (Figure 2f), doping analysis (Figure 2g,h), Brouwer-type diagrams etc. These include physically-motivated (but tunable) grouping of defect sites, full inclusion of metastable states, support for complex system constraints, optimisation over high-dimensional chemical & temperature space and highly customisable plotting. In-depth examples are provided in the tutorials.

- **Finite-Size Corrections:** Both the isotropic Freysoldt (FNV) (Freysoldt et al., 2009) and anisotropic Kumagai (eFNV) (Kumagai & Oba, 2014) image charge corrections are implemented automatically in doped, with tunable sampling radii / sites (which may be desirable for e.g. layered materials), automated correction plotting (to visualise/analyse convergence; Figure 2c), and automatic sampling error estimation.

- **Reproducibility & Tabulation:** doped has been built to support and encourage reproducibility, with all input parameters and calculation results saved to lightweight `JSON` files. This allows for easy sharing of calculation inputs/outputs and reproducible analysis. Several tabulation functions are also provided to facilitate the quick summarising of key quantities as exemplified in the tutorials (including defect formation energy contributions, charge transition levels (with/without metastable states), symmetry, degeneracy and multiplicity factors, defect/carrier concentrations, chemical potential limits, dopability limits, doping windows...) to aid transparency, reproducibility, comparisons with other

works, and general analysis. The use of these tabulated outputs in supporting information of publications is encouraged.

- **High-Throughput Compatibility:** `doped` is built to be compatible with high-throughput architectures such as atomate(2) (Mathew et al., 2017) or AiiDA (Huber et al., 2020), aided by its object-oriented Python framework, JSON-serializable classes and sub-classed `pymatgen` objects. Examples are provided on the documentation website.
- **ShakeNBreak:** `doped` is natively interfaced with our defect structure-searching code `ShakeNBreak` (Mosquera-Lois et al., 2022), seamlessly incorporating this phase in the defect calculation workflow. This step can optionally be skipped or an alternative structure-searching approach readily implemented.

Some additional features of `doped` include directional-dependent site displacement (local strain) analysis, deterministic & informative defect naming, molecule generation for gaseous competing phases, multiprocessing for expedited generation & parsing, shallow defect analysis (via `pydefect` (Kumagai et al., 2021)), Wyckoff site analysis (including *arbitrary/interstitial* sites), controllable defect site placement to aid visualisation and more.

The defect generation and thermodynamic analysis components of `doped` are agnostic to the underlying software used for the defect supercell calculations. Direct calculation I/O is fully-supported for `VASP` (Kresse & Furthmüller, 1996), while input defect structure files can be generated for several widely-used DFT codes, including `FHI-aims` (Blum et al., 2009), `CP2K` (Kühne et al., 2020), `Quantum Espresso` (Giannozzi et al., 2009) and `CASTEP` (Clark et al., 2005) via the `pymatgen` `Structure` object. Full support for calculation I/O with other DFT codes may be added in the future if there is sufficient demand. Moreover, `doped` is built to be readily compatible with other computational toolkits for advanced defect characterisation, such as `ShakeNBreak` for defect structure-searching, `py-sc-fermi` for advanced thermodynamic analysis under complex constraints (Squires et al., 2023), `easyunfold` for analysing defect/dopant-induced electronic structure changes (Zhu et al., 2024) or `CarrierCapture.jl/nonrad` for non-radiative recombination calculations (Kim et al., 2020; Turiansky et al., 2021).

`doped` has been used to manage the defect simulation workflow in a number of publications thus far, including Wang et al. (2024), Cen et al. (2023), Nicolson et al. (2023), Li et al. (2024), Kumagai et al. (2023), Woo et al. (2023), Wang et al. (2023), Mosquera-Lois & Kavanagh (2021), Mosquera-Lois, Kavanagh, Walsh, et al. (2023), Mosquera-Lois et al. (2024), Y.-T. Huang et al. (2022), Dou et al. (2024), Liga et al. (2023), Willis, Spooner, et al. (2023), Willis, Claes, et al. (2023), Krajewska et al. (2021), Kavanagh et al. (2021), Kavanagh, Savory, et al. (2022).

CRediT Author Contributions

Seán R. Kavanagh: Conceptualisation, Methodology, Software, Writing, Project Administration. **Alex G. Squires:** Code for complex doping analysis. **Adair Nicolson:** Code for shallow defect analysis. **Irea Mosquera-Lois:** Code for local strain analysis. **Katarina Brlec:** Competing phases code refactoring. **Aron Walsh & David Scanlon:** Funding Acquisition, Management, Ideas & Discussion. **All authors:** Feedback, Code Contributions, Writing – Review & Editing.

Acknowledgements

`doped` has benefited from feature requests and feedback from many members of the Walsh and Scanlon research groups, including (but not limited to) Xinwei Wang, Sabrine Hachmioune, Savya Aggarwal, Daniel Sykes, Chris Savory, Jiayi Cen, Lavan Ganeshkumar, Ke Li, Kieran Spooner and Luisa Herring-Rodriguez. S.R.K thanks Dr. Christoph Freysoldt and Prof. Yu Kumagai for useful discussions regarding the implementation of image charge corrections.

The initial development of doped was inspired by the pyCDT package from Broberg et al. (2018), while the original colour scheme for defect formation energy plots was inspired by work from Drs. Adam J. Jackson and Alex M. Ganose. doped makes extensive use of Python objects from the widely-used pymatgen (Ong et al., 2013) package (such as structure representations and VASP I/O handling), as well as crystal symmetry functions from spglib (Togo & Tanaka, 2018).

S.R.K. and A.N. acknowledge the EPSRC Centre for Doctoral Training in the Advanced Characterisation of Materials (CDTACM)(EP/S023259/1) for funding PhD studentships. DOS acknowledges support from the EPSRC (EP/N01572X/1) and from the European Research Council, ERC (Grant No. 758345). The PRAETORIAN project was funded by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee (EP/Y019504/1). This work used the ARCHER2 UK National Supercomputing Service (https://www.archer2.ac.uk), via our membership of the UK's HEC Materials Chemistry Consortium, which is funded by the EPSRC (EP/L000202, EP/R029431 and EP/T022213), the UK Materials and Molecular Modelling (MMM) Hub (Young EP/T022213).

References

Arrigoni, M., & Madsen, G. K. H. (2021). Spinney: Post-processing of first-principles calculations of point defects in semiconductors with Python. *Computer Physics Communications*, *264*, 107946. https://doi.org/10.1016/j.cpc.2021.107946

Blum, V., Gehrke, R., Hanke, F., Havu, P., Havu, V., Ren, X., Reuter, K., & Scheffler, M. (2009). Ab initio molecular simulations with numeric atom-centered orbitals. *Computer Physics Communications*, *180*(11), 2175–2196. https://doi.org/10.1016/j.cpc.2009.06.022

Broberg, D., Bystrom, K., Srivastava, S., Dahliah, D., Williamson, B. A. D., Weston, L., Scanlon, D. O., Rignanese, G.-M., Dwaraknath, S., Varley, J., Persson, K. A., Asta, M., & Hautier, G. (2023). High-throughput calculations of charged point defect properties with semi-local density functional theory – performance benchmarks for materials screening applications. *Npj Computational Materials*, *9*(1), 1–12. https://doi.org/10.1038/s41524-023-01015-6

Broberg, D., Medasani, B., Zimmermann, N. E. R., Yu, G., Canning, A., Haranczyk, M., Asta, M., & Hautier, G. (2018). PyCDT: A python toolkit for modeling point defects in semiconductors and insulators. *Computer Physics Communications*, *226*, 165–179. https://doi.org/10.1016/j.cpc.2018.01.004

Cen, J., Zhu, B., R. Kavanagh, S., G. Squires, A., & O. Scanlon, D. (2023). Cation disorder dominates the defect chemistry of high-voltage $LiMn_{1.5}Ni_{0.5}O_4$(LMNO) spinel cathodes. *Journal of Materials Chemistry A*, *11*(25), 13353–13370. https://doi.org/10.1039/D3TA00532A

Clark, S. J., Segall, M. D., Pickard, C. J., Hasnip, P. J., P., M. I. J., Refson, K., & Payne, M. C. (2005). First principles methods using CASTEP. *Zeitschrift Für Kristallographie - Crystalline Materials*, *220*(5-6), 567–570. https://doi.org/10.1524/zkri.220.5.567.65075

Dou, W., Spooner, K., Kavanagh, S., Zhou, M., & Scanlon, D. O. (2024). Giant Band Degeneracy via Orbital Engineering Enhances Thermoelectric Performance from $Sb_2Si_2Te_6$ to $Sc_2Si_2Te_6$. *ChemRxiv*. https://doi.org/10.26434/chemrxiv-2024-hm6vh

Freysoldt, C., Neugebauer, J., & Walle, C. V. de. (2009). Fully ab initio finite-size corrections for charged-defect supercell calculations. *Physical Review Letters*, *102*, 016402. https://doi.org/10.1103/PhysRevLett.102.016402

Giannozzi, P., Baroni, S., Bonini, N., Calandra, M., Car, R., Cavazzoni, C., Ceresoli, D., Chiarotti, G. L., Cococcioni, M., Dabo, I., Corso, A. D., Gironcoli, S. de, Fabris, S., Fratesi,

G., Gebauer, R., Gerstmann, U., Gougoussis, C., Kokalj, A., Lazzeri, M., … Wentzcovitch, R. M. (2009). QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, *21*(39), 395502. https://doi.org/10.1088/0953-8984/21/39/395502

Goyal, A., Gorai, P., Peng, H., Lany, S., & Stevanović, V. (2017). A computational framework for automation of point defect calculations. *Computational Materials Science*, *130*, 1–9. https://doi.org/10.1016/j.commatsci.2016.12.040

Huang, M., Zheng, Z., Dai, Z., Guo, X., Wang, S., Jiang, L., Wei, J., & Chen, S. (2022). DASP: Defect and dopant ab-initio simulation package. *Journal of Semiconductors*, *43*, 42101. https://doi.org/10.1088/1674-4926/43/4/042101

Huang, Y.-T., Kavanagh, S. R., Righetto, M., Rusu, M., Levine, I., Unold, T., Zelewski, S. J., Sneyd, A. J., Zhang, K., Dai, L., Britton, A. J., Ye, J., Julin, J., Napari, M., Zhang, Z., Xiao, J., Laitinen, M., Torrente-Murciano, L., Stranks, S. D., … Hoye, R. L. Z. (2022). Strong absorption and ultrafast localisation in $NaBiS_2$ nanocrystals with slow charge-carrier recombination. *Nature Communications*, *13*(1), 4960. https://doi.org/10.1038/s41467-022-32669-3

Huber, S. P., Zoupanos, S., Uhrin, M., Talirz, L., Kahle, L., Häuselmann, R., Gresch, D., Müller, T., Yakutovich, A. V., Andersen, C. W., Ramirez, F. F., Adorf, C. S., Gargiulo, F., Kumbhar, S., Passaro, E., Johnston, C., Merkys, A., Cepellotti, A., Mounet, N., … Pizzi, G. (2020). AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. *Scientific Data*, *7*(300), 1–18. https://doi.org/10.1038/s41597-020-00638-4

Kavanagh, S. R., Savory, C. N., Liga, S. M., Konstantatos, G., Walsh, A., & Scanlon, D. O. (2022). Frenkel Excitons in Vacancy-Ordered Titanium Halide Perovskites (Cs_2TiX_6). *The Journal of Physical Chemistry Letters*, *13*(47), 10965–10975. https://doi.org/10.1021/acs.jpclett.2c02436

Kavanagh, S. R., Scanlon, D. O., Walsh, A., & Freysoldt, C. (2022). Impact of metastable defect structures on carrier recombination in solar cells. *Faraday Discussions*, *239*(0), 339–356. https://doi.org/10.1039/D2FD00043A

Kavanagh, S. R., Walsh, A., & Scanlon, D. O. (2021). Rapid Recombination by Cadmium Vacancies in CdTe. *ACS Energy Letters*, *6*(4), 1392–1398. https://doi.org/10.1021/acsenergylett.1c00380

Kim, S., Hood, S. N., Gerwen, P. van, Whalley, L. D., & Walsh, A. (2020). CarrierCapture.jl: Anharmonic Carrier Capture. *Journal of Open Source Software*, *5*(47), 2102. https://doi.org/10.21105/joss.02102

Krajewska, C. J., Kavanagh, S. R., Zhang, L., Kubicki, D. J., Dey, K., Gałkowski, K., Grey, C. P., Stranks, S. D., Walsh, A., Scanlon, D. O., & Palgrave, R. G. (2021). Enhanced visible light absorption in layered $Cs_3Bi_2Br_9$ through mixed-valence Sn(II)/Sn(IV) doping. *Chemical Science*, *12*(44), 14686–14699. https://doi.org/10.1039/D1SC03775G

Kresse, G., & Furthmüller, J. (1996). Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Physical Review B*, *54*(16), 11169. https://doi.org/10.1103/PhysRevB.54.11169

Kühne, T. D., Iannuzzi, M., Del Ben, M., Rybkin, V. V., Seewald, P., Stein, F., Laino, T., Khaliullin, R. Z., Schütt, O., Schiffmann, F., Golze, D., Wilhelm, J., Chulkov, S., Bani-Hashemian, M. H., Weber, V., Borštnik, U., Taillefumier, M., Jakobovits, A. S., Lazzaro, A., … Hutter, J. (2020). CP2K: An electronic structure and molecular dynamics software package - Quickstep: Efficient and accurate electronic structure calculations. *The Journal of Chemical Physics*, *152*(19), 194103. https://doi.org/10.1063/5.0007045

Kumagai, Y., Kavanagh, S. R., Suzuki, I., Omata, T., Walsh, A., Scanlon, D. O., & Morito, H. (2023). Alkali Mono-Pnictides: A New Class of Photovoltaic Materials by Element Mutation. *PRX Energy*, *2*(4), 043002. https://doi.org/10.1103/PRXEnergy.2.043002

Kumagai, Y., & Oba, F. (2014). Electrostatics-based finite-size corrections for first-principles point defect calculations. *Physical Review B*, *89*(19), 195205. https://doi.org/10.1103/PhysRevB.89.195205

Kumagai, Y., Tsunoda, N., Takahashi, A., & Oba, F. (2021). Insights into oxygen vacancies from high-throughput first-principles calculations. *Physical Review Materials*, *5*, 123803. https://doi.org/10.1103/PhysRevMaterials.5.123803

Larsen, A. H., Mortensen, J. J., Blomqvist, J., Castelli, I. E., Christensen, R., Du\lak, M., Friis, J., Groves, M. N., Hammer, B., Hargus, C., Hermes, E. D., Jennings, P. C., Jensen, P. B., Kermode, J., Kitchin, J. R., Kolsbjerg, E. L., Kubal, J., Kaasbjerg, K., Lysgaard, S., … Jacobsen, K. W. (2017). The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter*, *29*(27), 273002. https://doi.org/10.1088/1361-648X/aa680e

Li, K., Willis, J., Kavanagh, S. R., & Scanlon, D. O. (2024). Computational Prediction of an Antimony-Based n-Type Transparent Conducting Oxide: F-Doped Sb_2O_5. *Chemistry of Materials*, *36*(6), 2907–2916. https://doi.org/10.1021/acs.chemmater.3c03257

Liga, S. M., Kavanagh, S. R., Walsh, A., Scanlon, D. O., & Konstantatos, G. (2023). Mixed-Cation Vacancy-Ordered Perovskites ($Cs_2Ti_{1-x}Sn_xX_6$; X = I or Br): Low-Temperature Miscibility, Additivity, and Tunable Stability. *The Journal of Physical Chemistry C*, *127*(43), 21399–21409. https://doi.org/10.1021/acs.jpcc.3c05204

Mathew, K., Montoya, J. H., Faghaninia, A., Dwarakanath, S., Aykol, M., Tang, H., Chu, I., Smidt, T., Bocklund, B., Horton, M., Dagdelen, J., Wood, B., Liu, Z.-K., Neaton, J., Ong, S. P., Persson, K., & Jain, A. (2017). Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows. *Computational Materials Science*, *139*, 140–152. https://doi.org/10.1016/j.commatsci.2017.07.030

Mosquera-Lois, I., & Kavanagh, S. R. (2021). In search of hidden defects. *Matter*, *4*(8), 2602–2605. https://doi.org/10.1016/j.matt.2021.06.003

Mosquera-Lois, I., Kavanagh, S. R., Ganose, A. M., & Walsh, A. (2024). Machine-learning structural reconstructions for accelerated point defect calculations. *arXiv*, *arXiv:2401.12127*. https://doi.org/10.48550/arXiv.2401.12127

Mosquera-Lois, I., Kavanagh, S. R., Klarbring, J., Tolborg, K., & Walsh, A. (2023). Imperfections are not 0 K: Free energy of point defects in crystals. *Chemical Society Reviews*, *52*(17), 5812–5826. https://doi.org/10.1039/D3CS00432E

Mosquera-Lois, I., Kavanagh, S. R., Walsh, A., & Scanlon, D. O. (2022). ShakeNBreak: Navigating the defect configurational landscape. *Journal of Open Source Software*, *7*(80), 4817. https://doi.org/10.21105/joss.04817

Mosquera-Lois, I., Kavanagh, S. R., Walsh, A., & Scanlon, D. O. (2023). Identifying the ground state structures of point defects in solids. *Npj Computational Materials*, *9*(1), 1–11. https://doi.org/10.1038/s41524-023-00973-1

Naik, M. H., & Jain, M. (2018). CoFFEE: Corrections For Formation Energy and Eigenvalues for charged defect simulations. *Computer Physics Communications*, *226*, 114–126. https://doi.org/10.1016/j.cpc.2018.01.011

Neilson, W. D., & Murphy, S. T. (2022). DefAP: A Python code for the analysis of point defects in crystalline solids. *Computational Materials Science*, *210*, 111434. https://doi.org/10.1016/j.commatsci.2022.111434

Nicolson, A., Kavanagh, S. R., Savory, C. N., Watson, G. W., & Scanlon, D. O. (2023). Cu_2SiSe_3 as a promising solar absorber: Harnessing cation dissimilarity to avoid killer antisites. *Journal of Materials Chemistry A*, *11*(27), 14833–14839. https://doi.org/10.1039/D3TA02429F

Ong, S. P., Richards, W. D., Jain, A., Hautier, G., Kocher, M., Cholia, S., Gunter, D., Chevrier, V. L., Persson, K. A., & Ceder, G. (2013). Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, *68*, 314–319. https://doi.org/10.1016/j.commatsci.2012.10.028

Parker, S. C., Cooke, D. J., Kerisit, S., Marmier, A. S., Taylor, S. L., & Taylor, S. N. (2004). From HADES to PARADISE—atomistic simulation of defects in minerals. *Journal of Physics: Condensed Matter*, *16*(27), S2735. https://doi.org/10.1088/0953-8984/16/27/010

Péan, E., Vidal, J., Jobic, S., & Latouche, C. (2017). Presentation of the PyDEF post-treatment Python software to compute publishable charts for defect energy formation. *Chemical Physics Letters*, *671*, 124–130. https://doi.org/10.1016/j.cplett.2017.01.001

Shen, J.-X., & Varley, J. (2024). Pymatgen-analysis-defects: A Python package for analyzing point defects in crystalline materials. *Journal of Open Source Software*, *9*(93), 5941. https://doi.org/10.21105/joss.05941

Squires, A. G., Scanlon, D. O., & Morgan, B. J. (2023). Py-sc-fermi: Self-consistent Fermi energies and defect concentrations from electronic structure calculations. *Journal of Open Source Software*, *8*(82), 4962. https://doi.org/10.21105/joss.04962

Togo, A., & Tanaka, I. (2018). `spglib`: A software library for crystal symmetry search. *arXiv*, *arXiv:1808.01590*. https://doi.org/10.48550/arXiv.1808.01590

Turiansky, M. E., Alkauskas, A., Engel, M., Kresse, G., Wickramaratne, D., Shen, J.-X., Dreyer, C. E., & Van de Walle, C. G. (2021). Nonrad: Computing nonradiative capture coefficients from first principles. *Computer Physics Communications*, *267*, 108056. https://doi.org/10.1016/j.cpc.2021.108056

Wang, X., Kavanagh, S. R., Scanlon, D. O., & Walsh, A. (2023). Four-electron negative-*U* vacancy defects in antimony selenide. *Physical Review B*, *108*(13), 134102. https://doi.org/10.1103/PhysRevB.108.134102

Wang, X., Kavanagh, S. R., Scanlon, D. O., & Walsh, A. (2024). Upper efficiency limit of Sb_2Se_3 solar cells. *arXiv*, *arXiv:2402.04434*. https://doi.org/10.48550/arXiv.2402.04434

Willis, J., Claes, R., Zhou, Q., Giantomassi, M., Rignanese, G.-M., Hautier, G., & Scanlon, D. O. (2023). Limits to Hole Mobility and Doping in Copper Iodide. *Chemistry of Materials*, *35*(21), 8995–9006. https://doi.org/10.1021/acs.chemmater.3c01628

Willis, J., Spooner, K. B., & Scanlon, D. O. (2023). On the possibility of p-type doping in barium stannate. *Applied Physics Letters*, *123*(16), 162103. https://doi.org/10.1063/5.0170552

Woo, Y. W., Li, Z., Jung, Y.-K., Park, J.-S., & Walsh, A. (2023). Inhomogeneous Defect Distribution in Mixed-Polytype Metal Halide Perovskites. *ACS Energy Letters*, *8*(1), 356–360. https://doi.org/10.1021/acsenergylett.2c02306

Xiong, Y., Bourgois, C., Sheremetyeva, N., Chen, W., Dahliah, D., Song, H., Zheng, J., Griffin, S. M., Sipahigil, A., & Hautier, G. (2023). High-throughput identification of spin-photon interfaces in silicon. *Science Advances*, *9*(40), eadh8617. https://doi.org/10.1126/sciadv.adh8617

Yuan, Z., Dahliah, D., Hasan, M. R., Kassa, G., Pike, A., Quadir, S., Claes, R., Chandler, C., Xiong, Y., Kyveryga, V., Yox, P., Rignanese, G.-M., Dabo, I., Zakutayev, A., Fenning, D. P., Reid, O. G., Bauers, S., Liu, J., Kovnir, K., & Hautier, G. (2024). Discovery of the Zintl-phosphide $BaCd_2P_2$ as a long carrier lifetime and stable solar absorber. *Joule*. https://doi.org/10.1016/j.joule.2024.02.017

Zhu, B., Kavanagh, S. R., & Scanlon, D. (2024). Easyunfold: A Python package for unfolding electronic band structures. *Journal of Open Source Software*, *9*(93), 5974. https://doi.org/10.21105/joss.05974