

PASCaL Python: A Principal Axis Strain Calculator

Monthakan Lertkiattrakul¹, Matthew L. Evans², and Matthew J. Cliffe¹

¹ School of Chemistry, University Park, Nottingham, NG7 2RD, United Kingdom ² Institut de la Matière Condensée et des Nanosciences, Université catholique de Louvain, Chemin des Étoiles 8, Louvain-la-Neuve 1348, Belgium

DOI: [10.21105/joss.05556](https://doi.org/10.21105/joss.05556)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: Sophie Beck ↗ 

Reviewers:

- [@MarDiehl](#)
- [@lbluque](#)
- [@hungpham2017](#)

Submitted: 30 May 2023

Published: 23 October 2023

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Summary

The response of crystalline materials to external stimuli: whether temperature, pressure or electrochemical potential, is critical for both our understanding of materials and their use. This information can be readily obtained through in-situ diffraction experiments, however if the intrinsic anisotropy of crystals is not taken into account, the true behaviour of crystals can be overlooked. This is particularly true for anomalous mechanical properties of great topical interest, such as negative linear or area compressibility (Cairns & Goodwin, 2015; Hodgson et al., 2014), negative thermal expansion (Chen et al., 2015) or strongly anisotropic electrochemical strain (Kondrakov et al., 2017).

We have developed PASCaL, Principal Axis Strain Calculator, a widely used web tool that implements the rapid calculation of principal strains and fitting to many common models for equations of state. It provides a simple web form user interface designed to be able to be used by all levels of experience. This new version of PASCaL is written in Python using the standard scientific Python stack (Harris et al., 2020; Virtanen et al., 2020), is released open source under the MIT license, and significantly extends the feature set of the original closed-source Fortran, Perl and Gnuplot webtool (Cliffe & Goodwin, 2012). Significant additional attention has been paid to testing, documentation, modularisation and reproducibility, enabling the main app functionality to now also be accessed directly through a Python API. The web app is deployed online at <https://www.pascalapp.co.uk> with the associated source code and documentation available on GitHub at [MJCliffe/PASCaL](https://github.com/MJCliffe/PASCaL).

Statement of Need

Characterising the anisotropic strain response of a crystalline material requires calculating not just the deformation along unit cell axes directions, which for low symmetry crystals have an arbitrary orientation relative to the mechanical properties, but also how changes in unit cell angles affect the principal (eigen-)strains and their orientations (eigenvectors). A great deal of additional information can be extracted from the stimulus-dependence of these principal strains, and fitting this dependence to equations-of-state can produce parameters of fundamental importance (e.g., compressibilities). Software capable of transforming experimental lattice parameters into the principal strains and the derived underlying thermodynamic properties (e.g. expansivities or compressibilities) is therefore of great utility to experimental scientists studying the response of crystals.

PASCaL is designed to be user-friendly and so accessible for the broader materials chemistry community: with a simple plain-text input, a limited number of options for each data-type, and easy to read and export output. It can calculate strains for any dataset containing crystallographic unit cell data as a function of an external variable, but has three standard options for variable temperature, pressure and electrochemical data, which selects the appropriate units and equations of state. This new version of PASCaL (v2.0) introduces a number of new

features including more accurate eigenvector matching, capabilities for electrochemical data, and more robust fitting to equation-of-state.

Description of the software

PASCaL takes unit cell parameters (in standard $a, b, c, \alpha, \beta, \gamma$ format), as a function of an external stimulus (temperature, pressure or electrochemical charge) as input.

The strain is calculated from the unit cell parameters in the conventional fashion ([Giacovazzo et al., 2011](#)). First, the unit cell parameters for each data point are orthogonalised using the Institute of Radio Engineers convention, where z is parallel to the c crystallographic axis, x is parallel to a^* , and $y = z \times x$. The strain, ϵ , is calculated from the symmetrised change between the initial data point and the each data point (Lagrangian) or between each data point and the initial data point (Eulerian) with options for either finite or infinitesimal strain.

The strain matrix is then diagonalised, to yield the principal strains (eigenvalues), and their directions (eigenvectors). In general, the ordering of the principal axes from any diagonalisation algorithm will not remain constant over the data series, particularly where the underlying data are noisy. The axes are therefore matched such that the directions remain consistent with those of the first data points.

These matched and sorted strain eigenvalues are then fitted to equations of state appropriate to the data type. All data types are first fitted linearly. Pressure datasets, which are highly non-linear, are additionally fitted to the widely used empirical equation-of-state: $\epsilon(p) = \epsilon_0 + \lambda(p - p_c)^\nu$ ([Goodwin et al., 2008](#)). For electrochemical datasets, where datasets are non-linear but analytic equations-of-state are not widely used, the data are fitted by Chebyshev polynomials with an appropriate regularisation procedure.

Linear fits are carried out on volumetric data, with additional fitting carried out for pressure datasets to the Birch-Murnaghan equations of state (second order, third order, and third order with critical pressure correction) ([Birch, 1947](#); [Sata et al., 2002](#)) and for electrochemical datasets using Chebyshev polynomials. The errors in calculated parameters and their derivatives with respect to the stimuli (i.e., compressibilities) are calculated using the residuals of the fits, as the propagation of the errors in the unit cell parameters through to the errors in the diagonalised strain is not well-defined in general. Errors in the stimulus are used solely to weight the relative importance of data points.

The output page consists of a number of tables of physically meaningful coefficients and interactive graphics ([Plotly Technologies Inc, 2015](#)) showing the stimuli dependence of the principal strains, their derivatives (for pressure and electrochemical data), and an indicatrix representation of the expansivity/compressibility.

Comparison with other software

There are a number of other software packages that can calculate principal strains and fit equations of state. [EOSfit](#) is a suite of programs designed to calculate thermal expansion and equations of state as a function of pressure and recent versions include principal strain calculation also. Standalone calculation of strain is available in the no-longer maintained WinStrain. EOSfit is an offline, closed source package that has a GUI and command line interface. [STRAIN](#) is a closed-source online tool included on the Bilbao Crystallographic Server, and can calculate strain calculations for a single pair of lattice parameters. [ELATE](#) is not designed for calculation of strains, but is a useful open-source tool for analysing full elastic constant tensors. ([Gaillac et al., 2016](#))

Acknowledgements

M.L. and M.J.C. acknowledge support from the Hobday bequest to the School of Chemistry, University of Nottingham and the University of Nottingham Propulsion Futures Beacon. M.L.E. thanks the BEWARE scheme of the Wallonia-Brussels Federation for funding under the European Commission's Marie Curie-Skłodowska Action (COFUND 847587). All authors also thank Ross Shonfield for help deploying the app and Madeleine Geers, Iain Oswald and Emily Meekel for their help testing the algorithms.

References

- Birch, F. (1947). Finite Elastic Strain of Cubic Crystals. *Physical Review*, 71(11), 809–824. <https://doi.org/10.1103/PhysRev.71.809>
- Cairns, A. B., & Goodwin, A. L. (2015). Negative linear compressibility. *Physical Chemistry Chemical Physics*, 17(32), 20449–20465. <https://doi.org/10.1039/C5CP00442J>
- Chen, J., Hu, L., Deng, J., & Xing, X. (2015). Negative thermal expansion in functional materials: Controllable thermal expansion by chemical modifications. *Chemical Society Reviews*, 44(11), 3522–3567. <https://doi.org/10.1039/C4CS00461B>
- Cliffe, M. J., & Goodwin, A. L. (2012). PASCAL: A principal axis strain calculator for thermal expansion and compressibility determination. *Journal of Applied Crystallography*, 45(6), 1321–1329. <https://doi.org/10.1107/S0021889812043026>
- Gaillac, R., Pullumbi, P., & Coudert, F.-X. (2016). ELATE: An open-source online application for analysis and visualization of elastic tensors. *Journal of Physics: Condensed Matter*, 28(27), 275201. <https://doi.org/10.1088/0953-8984/28/27/275201>
- Giacovazzo, C., Monaco, H. L., Artioli, G., Viterbo, D., Milanesio, M., Ferraris, G., Gilli, G., Gilli, P., Zanotti, G., & Catti, M. (2011). *Fundamentals of Crystallography* (C. Giacovazzo, Ed.; 3rd ed.). International Union of Crystallography. <https://doi.org/10.1093/acprof:oso/9780199573653.001.0001>
- Goodwin, A. L., Keen, D. A., & Tucker, M. G. (2008). Large negative linear compressibility of $\text{Ag}_3[\text{Co}(\text{CN})_6]$. *Proceedings of the National Academy of Sciences of the United States of America*, 105(48), 18708–18713. <https://doi.org/10.1073/pnas.0804789105>
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Hodgson, S. A., Adamson, J., Hunt, S. J., Cliffe, M. J., Cairns, A. B., Thompson, A. L., Tucker, M. G., Funnell, N. P., & Goodwin, A. L. (2014). Negative area compressibility in silver(I) tricyanomethanide. *Chemical Communications*, 50(40), 5264–5266. <https://doi.org/10.1039/c3cc47032f>
- Kondrakov, A. O., Schmidt, A., Xu, J., Geßwein, H., Mönig, R., Hartmann, P., Sommer, H., Brezesinski, T., & Janek, J. (2017). Anisotropic Lattice Strain and Mechanical Degradation of High- and Low-Nickel NCM Cathode Materials for Li-Ion Batteries. *The Journal of Physical Chemistry C*, 121(6), 3286–3294. <https://doi.org/10.1021/acs.jpcc.6b12885>
- Plotly Technologies Inc. (2015). *Collaborative data science*. <https://plot.ly>
- Sata, N., Shen, G., Rivers, M., & Sutton, S. (2002). Pressure-volume equation of state of the high-pressure B2 phase of NaCl. *Physical Review B*, 65(10), 104114. <https://doi.org/10.1103/PhysRevB.65.104114>

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ... van Mulbregt, P. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272. <https://doi.org/10.1038/s41592-019-0686-2>