




# PyThia: A Python package for Uncertainty Quantification based on non-intrusive polynomial chaos expansions

Nando Hegemann <sup>1\*</sup> and Sebastian Heidenreich <sup>1\*</sup>

<sup>1</sup> Physikalisch-Technische Bundesanstalt, Germany ¶ Corresponding author \* These authors contributed equally.

DOI: [10.21105/joss.05489](https://doi.org/10.21105/joss.05489)

## Software

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

Editor: [Vissarion Fisikopoulos](#) 

## Reviewers:

- [@ziyiin97](#)
- [@timokoch](#)

Submitted: 10 May 2023

Published: 14 September 2023

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

PyThia is a Python package for quantifying uncertainties by computing polynomial chaos surrogates for computationally expensive parametric models (e.g., parametric partial differential equations) and for performing a local and global sensitivity analysis of those parameters. At its core, PyThia aims to be close to mathematical foundations, general in its applications, and easy to use by non-experts. To achieve broad applicability in many areas, PyThia relies on a non-intrusive regression approach ([Farchmin et al., 2019](#); [Sudret, 2008](#)). This allows it to compute global polynomial approximations of complex physical models by accessing samples of inputs and outputs alone. In particular, this means that PyThia can be used without any adaptation of legacy software, independent of operating system, programming language, involved proprietary software, or computing device.

PyThia covers polynomial chaos expansions for arbitrary combinations of uniform, normal, Gamma, and Beta distributed parameters. The polynomial chaos expansion is handled in a sparse manner, meaning that the user can specify the desired expansion terms manually or automatically through a heuristic approach. PyThia is able to compute a surrogate for any kind of input/output training data. However, as it is designed for high-dimensional inverse problems, random sampled data are typically preferred over approaches such as quadrature points or sparse grids. Sample generation, according to a user-specified density or directly from the predefined parameter domains, is included in PyThia and can be used to create training inputs for the computationally expensive physical model of interest. It is even possible to create input samples according to an optimal distribution with respect to the polynomial chaos expansion of the stochastic parameters ([Cohen & Migliorati, 2017](#); [Farchmin et al., 2020](#)). The derived surrogate can then be used as an approximation to the physical model in parameter reconstruction tasks ([Farchmin et al., 2020](#)) or for local and global sensitivity analysis ([Farchmin et al., 2019](#); [Sudret, 2008](#)). The latter comes without any computational overhead due to the use of an orthonormal polynomial basis in the chaos expansion.

## Statement of need

In many physical applications it is only possible to infer the quantities of interest indirectly, which often involves computationally expensive numerical simulations to solve the inverse problem ([Andrle et al., 2019](#); [Jäger et al., 2021](#)). As a way to circumvent this, surrogates to approximate the numerical model can be derived. In particular, surrogates based on polynomial chaos expansions are very versatile as they can be used to approximate any function with a second moment ([Xiu & Karniadakis, 2002](#)), even in very high-dimensional spaces ([Eigel et al., 2014, 2023a, 2023b](#); [Schwab & Stuart, 2012](#)). In contrast to similar model classes such as neural networks, it is possible to exploit the mathematical structure of the polynomial

chaos expansion to compute moments (e.g. mean and variance), marginals, global parameter sensitivities (Sudret, 2008) and arbitrary derivatives of the surrogate analytically without any computational overhead. Additionally, the linear regression used to compute the polynomial chaos surrogate can be done in a non-intrusive way without the need to change the underlying numerical model by relying on function evaluations in randomly drawn samples. Using the polynomial chaos basis, these samples can be drawn from an optimal distribution, which even yields upper bounds for the required number of function evaluations needed to compute the surrogate.

## Target Audience

PyThia is based on the numpy and scipy packages only and is thus very easy to install and run in almost any Python environment. Moreover, to build a polynomial chaos surrogate of an arbitrary function, PyThia relies on a priori generated function evaluations. This non-intrusive approach allows for maximal interoperability of PyThia with other code environments, i.e., different languages, machines and even proprietary software, as no interface between PyThia and the model computation framework has to be implemented. In combination with the flexible choice of expansion multiindex set and the possibility to generate an optimal experimental design, PyThia provides an easy to use but still very flexible tool for arbitrary function approximation. Hence the primary target audience for PyThia are experimental physicists and engineers, who often do not have the extensive mathematical background required to fine tune complicated algorithms and who typically face very intricate simulation frameworks which cannot be adapted easily. For that reason PyThia has been utilized in meteorological applications ranging from shape reconstruction of circuit elements on photolithography masks (Casfor et al., 2020; Farchmin et al., 2019, 2020) to sensitivity analysis for modeling electrocardiography signals in humans (Winkler et al., 2022).

## State of the Field

Many software packages covering uncertainty quantification and function approximation using polynomial chaos are available both in python, e.g., (Baudin et al., 2017; Feinberg & Langtangen, 2015; Mohammadi et al., 2021; Weise et al., 2020), and other programming languages, e.g., (Mandelli et al., 2012–2023; Marelli & Sudret, 2018–2023).

One of the main differences between PyThia and other software packages is the possibility to specify the multiindex set manually. Typically, the multiindex set is either assembled directly or iteratively based on an  $\ell_q$ -bound of the multiindices (Lüthen et al., 2021). The pygpc library also allows for direct (sparse) specification of the multiindex set, but does not contain any routines to assemble these sets. Separating the creation of the multiindex set from the polynomial chaos approximation has the advantage that it is very accessible to test different types of surrogate model classes and provides a flexible interface for different schemes to choose an optimal set. Exemplary, PyThia implements a heuristic to choose the index set based on an estimate of the Sobol indices, but other approaches such as LASSO (Tibshirani, 1996), LARS (Efron et al., 2004) or estimator based adaptive choices (Eigel et al., 2023b) are possible as well.

Another important feature concerns the experimental design for the polynomial chaos expansion. As deterministic quadrature rules suffer from the curse of dimensionality, most libraries implement random or pseudo random sampling schemes for the polynomial chaos regression. However, as random sampling typically yields an extensive amount of evaluations of a potentially expensive model, it is crucial to choose a sampling scheme which requires as least samples as possible. To our knowledge PyThia is the only available software package providing the optimal weighted least-squares sampling scheme described in (Cohen & Migliorati, 2017), which gives provable optimal upper bounds for the required number of samples.

## Acknowledgements

The authors would like to acknowledge funding for the development of the PyThia UQ Toolbox from the German Central Innovation Program (ZIM) No. ZF4014017RR7 and the European project 20IND04-ATMOC. This project (20IND04-ATMOC) has received funding from the EMPIR programme co-financed by the Participating States and from the European Union's Horizon 2020 research and innovation programme.

## References

- Andrle, A., Hönicke, P., Schneider, P.-I., Kayser, Y., Hammerschmidt, M., Burger, S., Scholze, F., Beckhoff, B., & Soltwisch, V. (2019). Grazing incidence x-ray fluorescence based characterization of nanostructures for element sensitive profile reconstruction. In B. Bodermann, K. Frenner, & R. M. Silver (Eds.), *Modeling aspects in optical metrology VII*. SPIE. <https://doi.org/10.1117/12.2526082>
- Baudin, M., Dutfoy, A., Iooss, B., & Popelin, A.-L. (2017). OpenTURNS: An industrial software for uncertainty quantification in simulation. In *Handbook of uncertainty quantification* (pp. 2001–2038). Springer International Publishing. [https://doi.org/10.1007/978-3-319-12385-1\\_64](https://doi.org/10.1007/978-3-319-12385-1_64)
- Casfor, M., Farchmin, N., Pflüger, M., Nikolaev, K., Soltwisch, V., Heidenreich, S., Laubis, C., Kolbe, M., & Scholze, F. (2020). Sensitivity analysis for the detection of pitchwalk in self-aligned quadruple patterning by GISAXS. In O. Adan & J. C. Robinson (Eds.), *Metrology, inspection, and process control for microlithography XXXIV* (Vol. 11325, p. 113251D). International Society for Optics; Photonics; SPIE. <https://doi.org/10.1117/12.2552037>
- Cohen, A., & Migliorati, G. (2017). Optimal weighted least-squares methods. *SMAI Journal of Computational Mathematics*, 3, 181–203. <https://doi.org/10.5802/smai-jcm.24>
- Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32(2). <https://doi.org/10.1214/009053604000000067>
- Eigel, M., Farchmin, N., Heidenreich, S., & Trunschke, P. (2023a). Efficient approximation of high-dimensional exponentials by tensor networks. *International Journal for Uncertainty Quantification*, 13(1), 25–51. <https://doi.org/10.1615/int.j.uncertaintyquantification.2022039164>
- Eigel, M., Farchmin, N., Heidenreich, S., & Trunschke, P. (2023b). Adaptive nonintrusive reconstruction of solutions to high-dimensional parametric PDEs. *SIAM Journal on Scientific Computing*, 45(2), A457–A479. <https://doi.org/10.1137/21m1461988>
- Eigel, M., Gittelsohn, C. J., Schwab, C., & Zander, E. (2014). Adaptive stochastic Galerkin FEM. *Computer Methods in Applied Mechanics and Engineering*, 270, 247–269. <https://doi.org/10.1016/j.cma.2013.11.015>
- Farchmin, N., Hammerschmidt, M., Schneider, P.-I., Wurm, M., Bodermann, B., Bär, M., & Heidenreich, S. (2019). Efficient global sensitivity analysis for silicon line gratings using polynomial chaos. In B. Bodermann, K. Frenner, & R. M. Silver (Eds.), *Modeling aspects in optical metrology VII* (Vol. 11057, pp. 115–121). International Society for Optics; Photonics; SPIE. <https://doi.org/10.1117/12.2525978>
- Farchmin, N., Hammerschmidt, M., Schneider, P.-I., Wurm, M., Bodermann, B., Bär, M., & Heidenreich, S. (2020). Efficient Bayesian inversion for shape reconstruction of lithography masks. *Journal of Micro/Nanolithography, MEMS, and MOEMS*, 19(02), 1. <https://doi.org/10.1117/1.jmm.19.2.024001>
- Feinberg, J., & Langtangen, H. P. (2015). Chaospy: An open source tool for designing methods of uncertainty quantification. *Journal of Computational Science*, 11, 46–57.

<https://doi.org/10.1016/j.jocs.2015.08.008>

Jäger, K., Sutter, J., Schneider, P.-I., Hammerschmidt, M., & Becker, C. (2021). *Optical simulations of nanotextured perovskite/silicon tandem solar cell*. HZB Data Service. <https://doi.org/10.5442/ND000005>

Lüthen, N., Marelli, S., & Sudret, B. (2021). Sparse polynomial chaos expansions: Literature survey and benchmark. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2), 593–649. <https://doi.org/10.1137/20m1315774>

Mandelli, D., Wang, C., Cogliati, J., Talbot, P., Abdo, M., McDowell, D., Yoshiura, R., & Alfonsi, A. (2012–2023). *RAVEN: Risk Analysis Virtual Environment*. <https://raven.inl.gov/>.

Marelli, S., & Sudret, B. (2018–2023). *UQLab: The Framework for Uncertainty Quantification*. <https://www.uqlab.com/>.

Mohammadi, F., Eggenweiler, E., Flemisch, B., Oladyskin, S., Rybak, I., Schneider, M., & Weishaupt, K. (2021). *A surrogate-assisted uncertainty-aware Bayesian validation framework and its application to coupling free flow and porous-medium flow*. arXiv. <https://doi.org/10.48550/ARXIV.2106.13639>

Schwab, C., & Stuart, A. M. (2012). Sparse deterministic approximation of Bayesian inverse problems. *Inverse Problems*, 28(4), 045003. <https://doi.org/10.1088/0266-5611/28/4/045003>

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7), 964–979. <https://doi.org/10.1016/j.res.2007.04.002>

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1), 267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>

Weise, K., Poßner, L., Müller, E., Gast, R., & Knösche, T. R. (2020). Pygpc: A sensitivity and uncertainty analysis toolbox for python. *SoftwareX*, 11, 100450. <https://doi.org/10.1016/j.softx.2020.100450>

Winkler, B., Nagel, C., Farchmin, N., Heidenreich, S., Loewe, A., Dössel, O., & Bär, M. (2022). Global sensitivity analysis and uncertainty quantification for simulated atrial electrocardiograms. *Metrology*, 3(1), 1–28. <https://doi.org/10.3390/metrology3010001>

Xiu, D., & Karniadakis, G. E. (2002). The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2), 619–644. <https://doi.org/10.1137/s1064827501387826>