14th International Particle Accelerator Conference,Venice, Italy JACoW Publishing
ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-WEPL038

MODEL-INDEPENDENT DETERMINATION OF SOLENOID OFFSETS IN THE SEALAB INJECTOR * †

B. C. Kuske‡, Helmholtz-Zentrum Berlin, 12489 Berlin, Germany

Abstract

The SEALAB (SRF Electron Accelerator LABoratory) project at HZB is composed of an SRF photo gun and an SRF booster, followed by a diagnostic line and a recirculation path for ERL applications, formerly bERLinPro. In an SRF injector, only a single solenoid can be utilized to optimally focus the beam for small emittance. The alignment of the solenoid is crucial, as it is the dominant source of trajectory distortions in the facility. Polynomial Chaos Expansion (PCE) is a technique developed for risk management and uncertainty quantification. It is well suited for application in accelerators, although not well known. In this paper, PCE is used to set up surrogate models from calculated or measured data to determine the misalignment of the solenoid in SEALAB.

INTRODUCTION

The focus of SEALAB is the operation and use of the SRF gun and booster within the "Accelerator Research and Development" framework of the Helmholtz Association, ARD. Besides the successful operation of a 5 mA SRF injector, with an emittance < 1 mm mrad, also user experiments like Ultra-fast Electron Diffraction applications, UED, are foreseen.

In SRF gun injectors only a single solenoid can be utilized. It is positioned as close as possible to the gun cavity. In the case of the SEALAB gun module, the superconducting (sc) solenoid is operated at 4 K to 5 K, see Fig. 1.

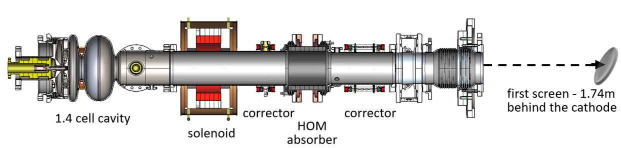

Figure 1: A sketch of the SEALAB gun. The sc solenoid (red) is located next to the 1.4-cell gun cavity. The first screen is located at 1.7 m behind the cathode.

The purpose of the solenoid is to focus the diverging beam coming out of the gun cavity. Its field is crucial for the emittance compensation. The exact positioning of the solenoid in the transverse directions (offsets and angles) with respect to the beam is vital for the trajectory. A misaligned solenoid also impacts the emittance, but more importantly, it leads to a highly non-linear and highly coupled dependence of the trajectory and the beam size on the solenoid field, which impedes operation and the interpretation of experimental results. Fig. 2 shows the motion of the beam on the first screen behind the solenoid for a transverse displacement of the solenoid by 0.25 mm in both directions, when the solenoid field is varied between −0.2 T to 0.2 T.

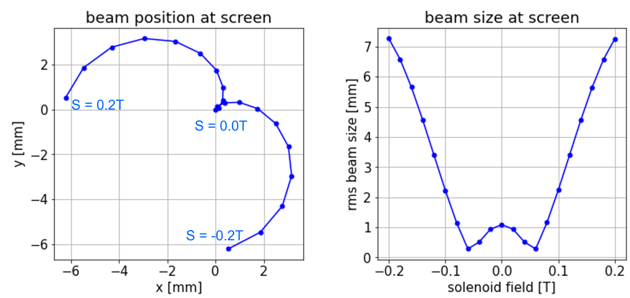

Figure 2: Left: The motion of the beam on the screen next to the gun module, for varying solenoid field and transverse solenoid offsets of 0.25 mm. Right: The related beam sizes.

The SEALAB solenoid is therefore mounted on a Hexapod mover, allowing for alignment in 6D space. Unluckily, the Hexapod mover lacks an absolute positioning system and the beam axis after cool-down might differ from the warm state, due to thermal effects. Therefore, a beam-based procedure to determine solenoid misalignment is needed for the commissioning of SEALAB.

POLYNOMIAL CHAOS EXPANSION

Polynomial Chaos Expansion (PCE) is a method for approximating a variable with unknown distribution by a series of polynomials, where the polynomials are functions of variables with known distribution. This paper introduces PCE as a helpful means during accelerator commissioning.

As PCE is a map between distributions, the application is not straightforward in a system like an injector, which usually has a single, optimized working point. Distributions need to be created first. In the example of solenoid misalignment, known distributions (Gaussian, uniform) of the transverse offsets and angles can be used to create an unknown distribution of beam positions on a screen. PCE can then deliver a surrogate model for the beam positions as a function of solenoid misalignment for a fixed solenoid field. Only relatively few data points are needed to set up the surrogate model. More important, the PCE approximation can be established exclusively from experimental data without relying on any machine model.

Detailed discussions on Uncertainty Propagation and PCE can be found in [1], [2], or [3] and references therein. The mathematical anchor of PCE is the Cameron-Martin theorem, which states that every random distribution (with finite

* Work supported by German Bundesministerium für Bildung und Forschung, Land Berlin, and grants of Helmholtz Association
† Topic submitted to PRAB
‡ bettina.kuske@helmholtz-berlin.de

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-WEPL038

variance) can be represented in terms of polynomial series developed around arbitrary canonical probabilistic distributions (uniform, Gaussian…), similar to the approximation of arbitrary functions by Taylor series.

Let z be a random input vector of dimension d, with a given probability density function $f(z) = \prod_{j=1}^{d} f_j(z_j)$, i.e. the problem depends on d variables. Let $u(z)$ be the system or model response to z, again a multi-dimensional random variable with finite variance. Then $u(z)$ can be written as

$$u(z) = \sum_{|i|=0}^{\infty} \alpha_i \Psi_i(z),$$

where i is a multi-index, $i \in \mathbb{N}^d$, and $\alpha_i \in \mathbb{R}$. The α_i have to be determined and represent the moments of the distribution of $u(z)$. The $\Psi_i(z)$ form the PCE basis composed of multivariate orthonormal polynomials. The polynomial basis to be used is determined by the distribution chosen for the random input data. Uniform distributions use Legendre polynomials, Gaussian distributions use Hermite polynomials [2]. Any truncation of the series at a certain polynomial order represents an approximation to the model response, a surrogate model. The coefficients α_i are computed by minimizing the mean square residual error between the model response and the truncated series, for details see [4].

The number of coefficients, $no(\alpha_i)$, used in the surrogate model is given by

$$no(\alpha_i) = \frac{(p+d)!}{p!\,d!},$$

where p is the order of the polynomial series and d is the number of variables. The approximate number of sample points, sp, that should be provided, depends on the method used to fit the polynomial coefficients to the data [2]. Using linear regression, the number is

$$sp_{lr} = (d-1) * no(\alpha_i).$$

The mathematical theory behind PCE has been coded in libraries like Chaospy for Python, or UQLAB in MATLAB.

THE SURROGATE MODEL

To simulate the experiment, an arbitrary, but fixed misalignment of the solenoid has to be assumed: $z_0 = (x_0, y_0, \theta_0, \phi_0) = (3\,\mathrm{mm}, -2\,\mathrm{mm}, -5\,\mathrm{mrad}, 7\,\mathrm{mrad})$, where θ and ϕ denote the angle with respect to the horizontal and vertical axis, respectively. For successful beam position measurements, the focal point of the solenoid is moved close to the location of the screen by using a solenoid field value S = 0.04 T.

The relative displacement of the solenoid with respect to its initial position is simulated using the sampler option of OPAL, [5]. The relative transverse offsets are ±5 mm, and the angular offsets are varied by ±10 mrad. For these $2^4 = 16$ cases, the beam is tracked from the cathode to the screen. In addition, random test cases have to be calculated. Fig. 3 summarizes the tracking calculations. Blue dots represent the relative offsets of the solenoid (left plot), the angles are not shown. The positions on the screen are displayed on the right plot. The circle marks the size of the screen. The red dots indicate 10 random test cases. The surrogate model is calculated in a Jupyter notebook, [6], using the `chaospy.py` package, [7], and the `scipy.stats` statistics package, [8]. The distributions of the additional displacements Δx, Δy, $\Delta\theta$, and $\Delta\phi$ are uniform. The normalized parameters of the distributions are calculated using `getattr(scipy.stats, distribution)`. From the distribution parameters, the joint distribution is calculated (`chaospy.J`). A generic polynomial of the respective order is set up using `chaospy.expansion.stieltjes(order, jointdistribution)`. Finally, a regression fit (`cp.fit_regression`) adjusts the coefficients of the polynomial to the tracking data.

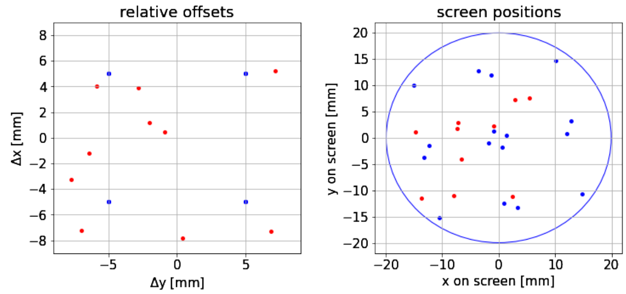

Figure 3: Left: The relative offsets used in the sample run (blue) and in the test run (red), (the angles are not shown). Right: The resulting beam positions on the screen.

For different polynomial orders, Table 1 displays the number of coefficients used in the expansions, the number of sample points needed to set up a reasonable model, and the relative error compared to tracking results for a single, arbitrary data point, indicating quick convergence. The difference between the 2^{nd} and 3^{rd} order polynomials is ≈0.02 %. 16 sample points correspond to a PCE of first order.

Table 1: Number of Coefficients and Samples

order	# α_i	# samples	error δx, δy [%]
1	5	15	-1.3156, 1.2445
2	15	45	-0.0219, 0.1696
3	35	105	-0.0006, -0.0013

The quality of the first- and second-order surrogate models are displayed in Table 2. The accuracy of the second-order model on the sample points is higher, but due to the insufficient number of samples, the test data is not well represented, showing large errors.

Table 2: Model Quality (L_2-norm)

	order	δx	δy
model data	1	0.0023	0.0013
model data	2	0.0001	0.0001
test data	1	0.0035	0.0031
test data	2	2.1398	3.9318

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-WEPL038

Misalignment Determination

Surrogate models for two field values are needed to determine the misalignment of the solenoid. For a fixed solenoid field value, the x and y positions of the beam on the screen can be represented by a first-order polynomial of the 4 misalignment components:

$$x = a_x \Delta x + b_x \Delta y + c_x \Delta\theta + d_x \Delta\phi + e_x$$
$$y = a_y \Delta x + b_y \Delta y + c_y \Delta\theta + d_y \Delta\phi + e_y,$$

where e_x and e_y are the beam's positions for z_0. The coefficients are determined by the PCE. For two solenoid field values, we look for the misalignment that satisfies :

$$\begin{pmatrix} e_{x,1} \\ e_{y,1} \\ e_{x,2} \\ e_{y,2} \end{pmatrix} = \begin{pmatrix} a_{x,1} & b_{x,1} & c_{x,1} & d_{x,1} \\ a_{y,1} & b_{y,1} & c_{y,1} & d_{y,1} \\ a_{x,2} & b_{x,2} & c_{x,2} & d_{x,2} \\ a_{y,2} & b_{y,2} & c_{y,2} & d_{y,2} \end{pmatrix} \times \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta\theta \\ \Delta\phi \end{pmatrix}$$

The solution $z_{fit} = (3.0097, -2.0055, -5.0079, 7.0012)$, has an absolute error < 10 µm (offsets) and < 8 µrad (angles).

Error Analysis

Errors occur in the PCE model and the experiment. The Hexapod mover is specified with a µm, resp. µrad precision and repeatability. The error analysis, therefore, concentrates on the modeling error and the position measurement error.

Modeling Error: 10 different randomly misaligned solenoids were investigated. Fig. 4 shows the initial misalignment on the left, offsets in red, and angles in blue. The absolute error, δz, of the fitted misalignment, z_{fit}, is displayed on the right. The RMS error is better than 5 µm for the offsets and 4 µrad for the angles.

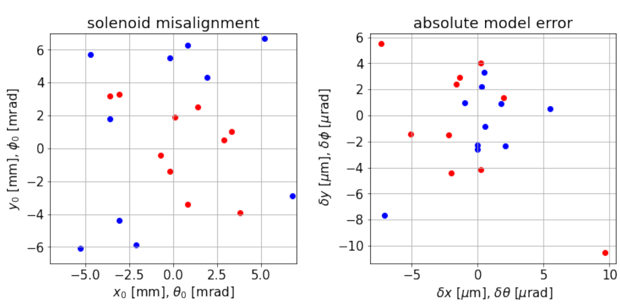

Figure 4: Left: 10 random solenoid misalignments (red) and angles (blue). Right: the absolute model error for each case.

Experimental Error: Gaussian distributed errors are added to the (calculated) positions on the screen. For 50 random error distributions, the PCE model is set up and the solenoid misalignment is determined. In Fig. 5 the result is shown, separately for each component of z_{fit}. The colored circles show the absolute error of each case, the black bars indicate the RMS error for increasing measurement errors of 0.001, 0.01, 0.05, and 0.1 mm. For a realistic position measurement error of 10 µm, the transverse RMS position error is 2.3 µm, and the RMS angular error is ≈12 µrad. The measurement error translates linearly into the uncertainty of the solenoid position.

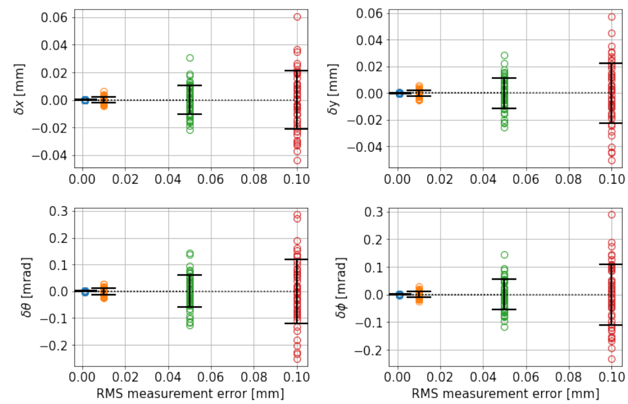

Figure 5: Absolute error of 50 seeds (colored circles) and RMS error (black error bars) for each component of z_{fit}.

It should be mentioned, that the scope of PCE reaches far beyond linear models. E.g., the beam motion shown in Fig. 2 (left) can be reproduced well, using a 7^{th} order polynomial, [9]. Successful models using a 6^{th} order PCE have been set up to predict the bunch length or the emittance of the bERLinPro injector, including the field flatness in the gun, cathode position, laser pulse length, and spot size, the gun voltage and the solenoid field as input parameters, [10].

CONCLUSION

A new method to determine solenoid misalignment in low-emittance injectors is proposed. It differs from existing methods in keeping the solenoid strength fixed while varying the alignment of the solenoid. The method, therefore, depends on a remote-controlled 4D positioning system that allows for adjustment of the solenoid position.

For a constant solenoid field, the beam's position depends dominantly linear on each separate misalignment. The first-order PCE decomposes the motion of the beam into these linear contributions.

The method is completely model-independent. It uses relative deviations from the solenoid's position and beam position measurements. Both can be very accurate so the accuracy of the method is high. The time needed to re-position the solenoid 2 x 16 times and to measure the respective beam position is estimated to be much less than 1 h. The numerical calculation of the misalignment from the measured data takes no time. So besides being highly accurate, the method also is fast. The injector of bERLinPro will be ready for commissioning towards the end of 2023. To our knowledge, this will be the first time where PCE is applied online in an operating accelerator.

ACKNOWLEDGEMENT

I thank A. Adelmann and M. Frey for their support with PCE and J. Völker for sharing his experimental experience.

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-WEPL038

REFERENCES

[1] A. Adelmann, ”On Nonintrusive Uncertainty Quantification and Surrogate Model Construction in Particle Accelerator Modeling”, *SIAM/ASA Journal on Uncertainty Quantification*, 7(2), (2019), doi:10.1137/16M1061928

[2] M. Frey, A. Adelmann, ”Global Sensitivity Analysis on Numerical Solver Parameters of Particle-In-Cell Models in Particle Accelerator Systems”, Paul Scherrer Institut, CH-5232 Villigen, Switzerland, PhD Thesis, arXiv:2001.10411v4 [physics.comp-ph] 2 Sep 2020.

[3] Emily Gorcenski, ”Polynomial Chaos: A Technique for modelling uncertainty”, PyData meeting Berlin 2017, https://www.youtube.com/watch?v=Z-Qio-n6yPc

[4] B. Sudret, ”Surrogate models for global sensitivity analysis - Old and New”, *8th International Conference on Sensitivity Analysis of Model Output*, La Reunion, France, Nov. 2016.

[5] A. Adelmann, P. Calvo, M. Frey, A. Gsell, U. Locans, C.j. Metzger-Kraus, N. Neveu, C. Rogers, S. Russell, S. Sheehy, J. Snuverink, D. Winklehner, ”OPAL a Versatile Tool for Charged Particle Accelerator Simulations”, arXiv e-prints, May 2019, arXiv:1905.06654, https://ui.adsabs.harvard.edu/abs/2019arXiv190506654A

[6] https://jupyter.org/

[7] https://chaospy.readthedocs.io/en/master/

[8] https://docs.scipy.org/doc/scipy/reference/stats.html

[9] B. Kuske, ”Using Polynomial Chaos Expansion to determine Solenoid misalignment during the bERLinPro Commissioning”, talk at the Energy Recovery Workshop 2022, Ithaca, NY, USA, to be published.

[10] B. Kuske, ”Commissioning of the bERLinPro Diagnostics Line using Machine Learning”, *Proceedings of the 63rd ICFA Advanced Beam Dynamics Workshop on Energy Recovery Linacs*, Berlin, Germany, September 2019, paper WECOYBS04, doi:10.18429/JACoW-ERL2019-WECOYBS04