

Surrogate Models for Particle Accelerators

Andreas Adelmann

FFA 2019 - Villigen - 21. November 2019



Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 1 / 24



- Motivation
- Polynomial Chaos Expansion (PCe)
- Artificial Neural Nets (ANNs)



References

- A. Adelmann, On Nonintrusive Uncertainty Quantification and Surrogate Model Construction in Particle Accelerator Modeling, SIAM/ASA Journal on Uncertainty Quantification, **7**(3), 2019
- N. Wiener, *The homogenous chaos*, Amer. J. Math. **30**, 897-936, (1938)
- A. Edelen, N. Neveu M. Frey Y, Huber, Ch. Mayes, A. Adelmann, Machine Learning to Enable Orders of Magnitude Speedup in Multi-Objective Optimizatin of Particle Accelerator Systems, submitted to PRAB, https://arxiv.org/abs/1903.07759
- N. Neveu, L. Spentzouris, A. Adelmann, Y, Ineichen, A. Kolano, C. Metzger-Kraus, C. Bekas, C, A. Curioni, P. Arbenz, Parallel general purpose multiobjective optimization framework with application to electron beam dynamics, Phys. Rev. Accel. Beams, 22(5), 054602, 2019



Motivation



FFA 2019 - Villigen - 21. November 2019 Page 4 / 24



Motivation



Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 4 / 24





Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 4 / 24





Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 5 / 24



Motivation Weak vs. Strong Scaling







FFA 2019 - Villigen - 21. November 2019 Page 6 / 24



Motivation Weak vs. Strong Scaling









FFA 2019 - Villigen - 21. November 2019 Page 7 / 24



Surrogate Model a Simple Definition

Surrogate models (SMs) approximate a computationally expensive simulator η . Suppose

 $y(x) = \eta(x), \quad x \in \mathbb{R}^n, \quad y \in \mathbb{R}^m$

then the SM is an approximation of the form

 $\hat{y}(x) = \hat{\eta}(x)$

such that

$$y(x) = \hat{y}(x) + \varepsilon$$

and $\hat{y}(x)$ cheap to evaluate.



A Complicated Example (PCe)

[AA, On Nonintrusive UQ and SM Construction ... (2019)]

• Goal: model halo
$$h_x = rac{\langle q_x^4
angle}{\langle q_x^2
angle^2} - C$$
 & $ilde{x}$

• Simplification: 3 design parameters

- **1** initial condition: $\langle xp_x \rangle$
- 2 collimator setting: ΔC_1
- ${f 0}$ rf phase setting: ϕ_1 .





A Complicated Example (PCe)

[AA, On Nonintrusive UQ and SM Construction ... (2019)]

• Goal: model halo
$$h_x = rac{\langle m{q}_x^4
angle}{\langle m{q}_x^2
angle^2} - C$$
 & $ilde{x}$

• Simplification: 3 design parameters

- **1** initial condition: $\langle xp_x \rangle$
- 2 collimator setting: ΔC_1
- 3 rf phase setting: ϕ_1 .



This extensive search in the 3 dimensional parameter space requires PIC models with enough particles to estimate halo at a given location.



Illustration of the Basic Ideas (PCe)

Let η be the simulator



$$u^n = \eta(\xi^n)$$
 , Villigen - 21. November 2019 Page 10 / 24



Polynomial Chaos Expansions (PCe) I

All square integrable, second-order random processes with finite variance output, $y(\xi) \in L_2(\Omega, \mathcal{F}, \mathcal{P})$, can be written as [N. Wiener]

$$y = \sum_{k=0}^{\infty} \alpha_k \Psi_k(\xi).$$

- y: Random Variable (RV)
- α_k PC coefficients (deterministic)
- Ψ_k : Hermite polynomial, ξ : Gaussian RV

Expansion in terms of functions of random variables multiplied with deterministic PC coefficients.



Polynomial Chaos Expansions (PCe) I [AA, On Nonintrusive UQ and SM Construction ... (2019)]

Algorithm: generate for each design variable, a PC surrogate model to order \boldsymbol{K}

- generate N samples (ξ^n) according to the sampling strategy of interest
- create the deterministic training points with high fidelity simulations (non-intrusive)

 $u^n = \eta(\xi^n).$

③ solve for α_k via

- orthogonal Galerkin-projection
- regression methods
- Bayesian Compressive Sensing



Polynomial Chaos Expansions (PCe) II [AA, On Nonintrusive UQ and SM Construction ... (2019)]



FFA 2019 - Villigen - 21. November 2019 Page 13 / 24



Predictions - h_x



FFA 2019 - Villigen - 21. November 2019 Page 14 / 24



Predictions of \tilde{x} with 95% CL





Predictions of \tilde{x} with 95% CL



Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 15 / 24



Sensitivities



Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 16 / 24



MOGA for the Argonne Wakefield Accelerator [N. Neveu, AA, et al. (2019)]



- Full 3D Start to End (S2E) needed
- OPAL Particle In Cell (PIC) model
- Very timeconsuming
- Parameter study / multi-objective optimisation expensive



MOGA for the Argonne Wakefield Accelerator [N. Neveu, AA, et al. (2019)]



- One 3D medium fidelity S2E 3600 (s) on 32 cores
- $3 \dots 7$ Qols, $6 \dots 15$ Dvars
- Genetic Algorithm setup: G = 200, I = 100



MOGA for the Argonne Wakefield Accelerator [N. Neveu, AA, et al. (2019)]



• OPAL MOGA: 24h on ≈ 5000 cores

Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 17 / 24



Machine Learning to Construct a cheap & accurate SM [A. Edelen et al (2019)]



• optimise parameters at a given location

- One 3D S2E 300 (s) on 8 cores
- 7 Qols, 7 Dvars
- MOGA (in OPAL): $G = 200, I = 100 \Rightarrow$ ground truth

Surrogate Models for Particle Accelerators

FFA 2019 - Villigen - 21. November 2019 Page 18 / 24



4 Step Process to Construct an ANN SM



- generate random sample
- split labeled data set (80%, 20%)
- 3 create ANN
- understand quality



Artificial Neural Network

- Fully connected and feed forward
- Hyperparameters
 - A lot of different architectures
 - Learning rate
- Best results using
 - 6-12-24-48-96-8
 - Adam optimizer with 0.0001 learn rate, trained for 30k epochs
 - Tanh as activation, no activation after output layer
 - Weights inverse proportional to the estimated density likelihood



Figure: Neural Network scheme https://towardsdatascience.com



Fidelity on the Test Data I



FFA 2019 - Villigen - 21. November 2019 Page 21 / 24



When all comes together





OPAL MOGA: 24h on ≈ 5000 cores





OPAL MOGA: 24h on ≈ 5000 cores

Train ANN once: 2-5h on ≈ 128 cores







OPAL MOGA: 24h on ≈ 5000 cores

Train ANN once: 2-5h on ≈ 128 cores





ANN & MOGA : ≈ 30 minutes \Rightarrow



OPAL MOGA: 24h on ≈ 5000 cores

Train ANN once: 2-5h on ≈ 128 cores

ANN & MOGA : ≈ 30 minutes \Rightarrow







FFA 2019 - Villigen - 21. November 2019 Page 23 / 24



OPAL MOGA: 24h on ≈ 5000 cores

Train ANN once: 2-5h on ≈ 128 cores

ANN & MOGA : ≈ 30 minutes \Rightarrow







Speedup > 1 000 000 & accurate



 Surrogate Models are the only way to achieve real-time performance & accuracy in complicated system!

• ANN & PCe are wonderful tools to achieve this

Much to learn robustness, training sizes, & accuracy