RAPID DESIGN OPTIMIZATION USING ALTAIR® PHYSICSAI™

Shubhamkar Kulkarni – Product Specialist, Engineering Data Science, Analytics and IoT Development, Altair / October 29, 2024

Introduction

Computer-aided engineering (CAE) has revolutionized product design by enabling the creation of accurate virtual models that reduce the need for physical testing and shorten the design cycle. Companies' reliance on CAE is increasing daily, with some companies envisioning an entirely virtual development process in the near future. Despite its many advantages, CAE tools remain underused, usually incorporated only in the later stages of design. The main limitations for these tools are the significant time and high-level expertise required to use them. Undoubtedly, CAE has huge potential if companies can democratize it by making it easier to use and deploy. Altair® physicsAI™ does just this and brings the power of artificial intelligence (AI) to CAE simulation.

Presently, engineers evaluate design variants mainly through design of experiments (DOE) on parametrized models. Parameterizing models is usually an extra step that limits variety and needs careful monitoring to ensure simulation feasibility. As the number of parameters increases, the required number of solver evaluations scales up rapidly. Additionally, traditional regression models abstract simple mathematical relationships between parameters and performance indicators and may not capture the manifold dependencies. On the contrary, physicsAI utilizes the power of Altair's proprietary geometric deep learning engine to capture rich contextual information from existing simulation data without the need for parameterization. physicsAI enables rapid evaluation of result fields, time dependent responses and key point indicators for new design variants.

In this paper, we demonstrate the effective use of a physicsAI machine learning model for greatly speeding up an iterative design optimization process. The objective is to identify the magnitude of applied loads on an airplane wing to match the target bending moment diagram (BMD) in a laboratory simulation of in-flight conditions. Working in tandem with our in-house optimization tool Altair® HyperStudy®, a seamless solution for developing an optimized design is created. Our results show that physicsAI can make predictions 10-100x quicker and achieve an accuracy higher than 90% relative to the finite element analysis (FEA) results in the early stages of design.

Challenges

Effectively exploring the design space to identify the best design variant is an important goal for engineers. Currently, the most popular strategies include heuristic and systematically generated DOEs. The precursor to this involves parameterizing the designs through design variables. These input parameters are then varied per the DOE scheme and the results are used to identify the effect of parameters on the outputs by fitting mathematical relationships between them. The variation of parameters must be carefully monitored as it can sometimes result in infeasible design variants. Moreover, parameterization restricts variety and extensibility, usually requiring new evaluations for each parameter set.

physicsAI uses an alternative methodology in which geometric deep learning is used to capture the effect of geometric variation on the result values (Figure 1). In addition to geometric variability, custom inputs such as thicknesses, materials, loads, or boundary conditions can also be included. Furthermore, physicsAI machine learning models can be trained to predict custom outputs such as key performance indicators (KPIs) or time-dependent response curves in addition to full 3D field predictions.

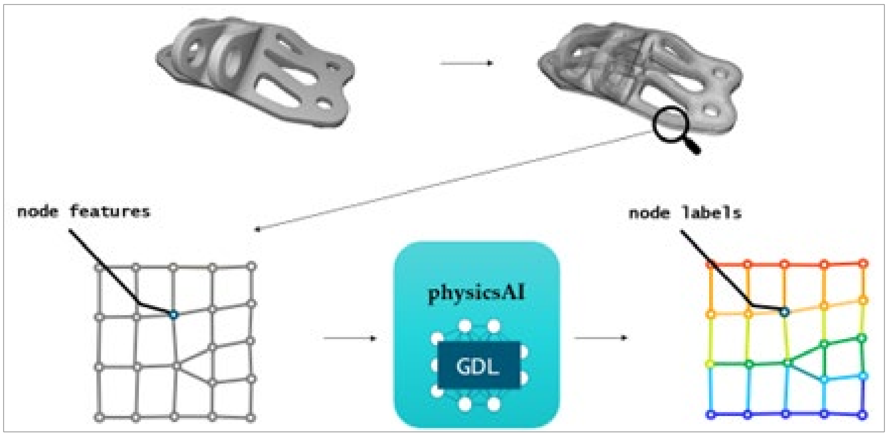

Figure 1. Geometric deep learning in physicsAI

Problem Formulation

In this case study, we predict the BMD of an airplane wing, which can be considered to be a vector KPI. The wing is approximately 24 meters long, constrained at the base, and loaded as a cantilever structure. It is meshed using a combination of 5,000 shell and beam elements. Loads of varying magnitudes are applied at nine different locations on the wing along the span.

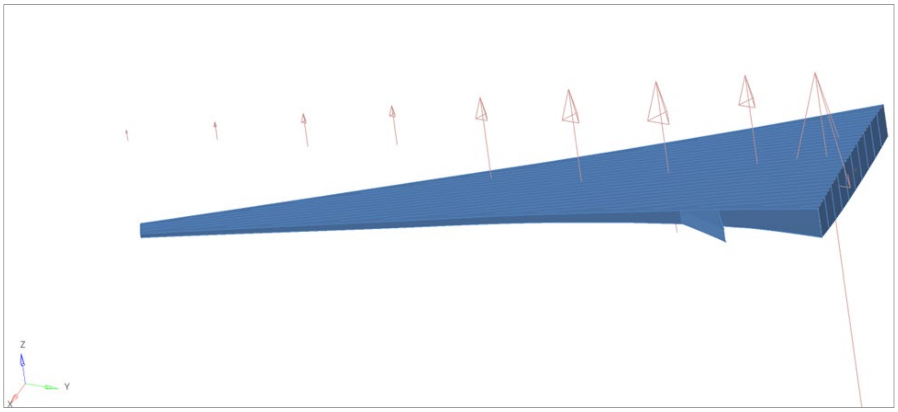

Figure 2. 3D wing geometry with applied loads

The optimization problem is set up in HyperStudy by parameterizing the magnitude of the applied forces as the input variables. The output responses are the bending moments in seven sections along the length of the wing (Figure 3) manifested in the form of a BMD. The target BMD is supplied in vector form and is used as the datum or reference for the output responses. The difference between the target and predicted BMD is integrated into a single scalar value using HyperStudy's Area tool (Figure 4). The objective is to reduce this value to as close to 0 as possible for matching the reference curve to the target by adjusting the input variables within ±33% of the initial value.

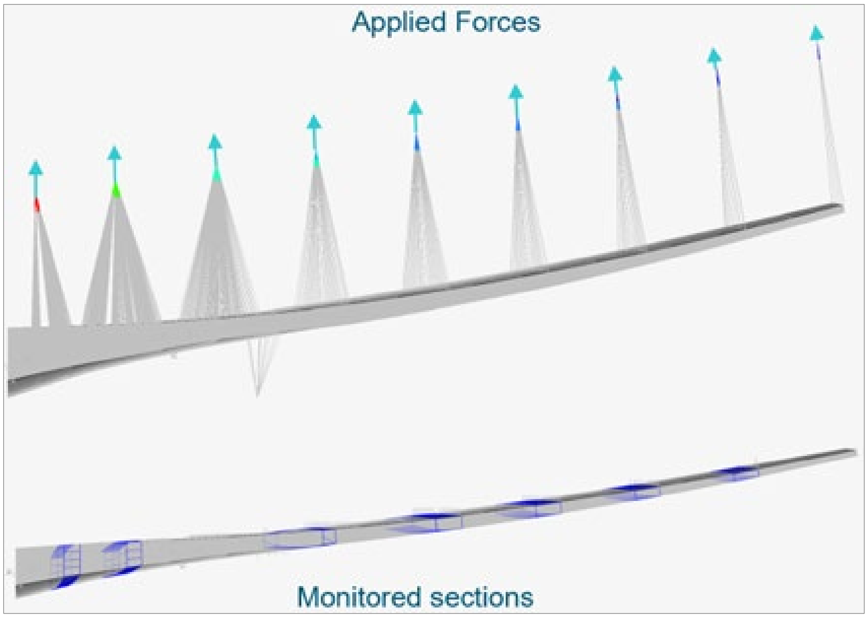

Figure 3. Force application locations and monitored sections for bending moments

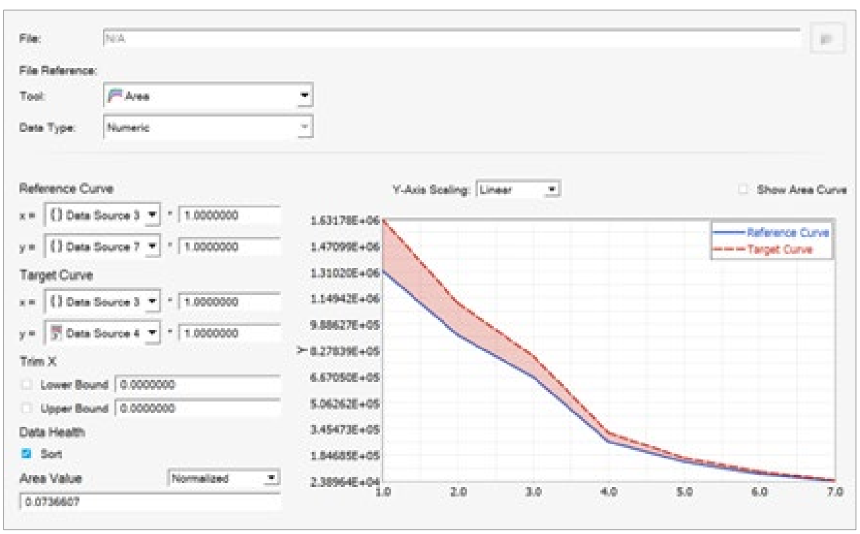

Figure 4. Area tool for calculating the difference between curves

physicsAI Workflow

Figure 1. illustrates the process of training and deploying a physicsAI model. The first step involves using historical or synthetically generated 3D simulation models for training. Consistent with data science practices, a sufficient number of training data points (typically in the hundreds) should be used for training; a fraction of the training data points should be reserved for testing the model quality. These are referred to as validation data points and are not used for training, but rather to ensure that the model is not overfit. In this study, training data was synthetically generated by running a DOE of Altair® OptiStruct® finite element simulations. Of the 1,129 usable DOE datapoints, 80% (903) points were used for training and 20% (226) were reserved for testing.

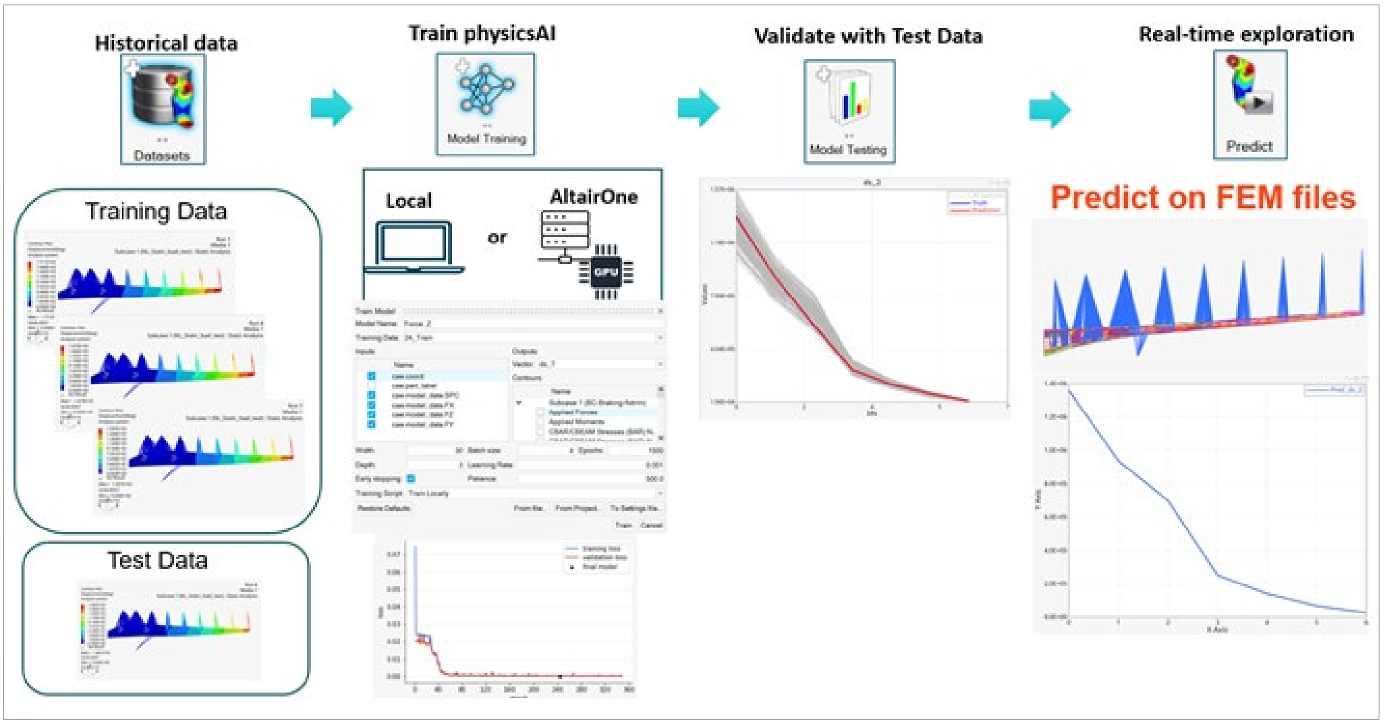

Figure 5. A typical physicsAI workflow

During the second step, i.e., training, nodal coordinates and load magnitudes are used as inputs and the BMD vector is used as the true outputs that the model has to learn to predict.

A fraction of the training files (15%) is reserved for validation and the loss function is tracked on these validation datapoints. A well-trained model will have both the training and validation curves approaching zero. The training was run on a desktop with an Intel i7 CPU and 32 GB RAM. The training was stopped after 349 episodes since no improvement was observed for 100 epochs and the configuration for minimum loss function value, i.e., epoch 244 was identified as the optimal (Figure 2). The training time was 43 minutes.

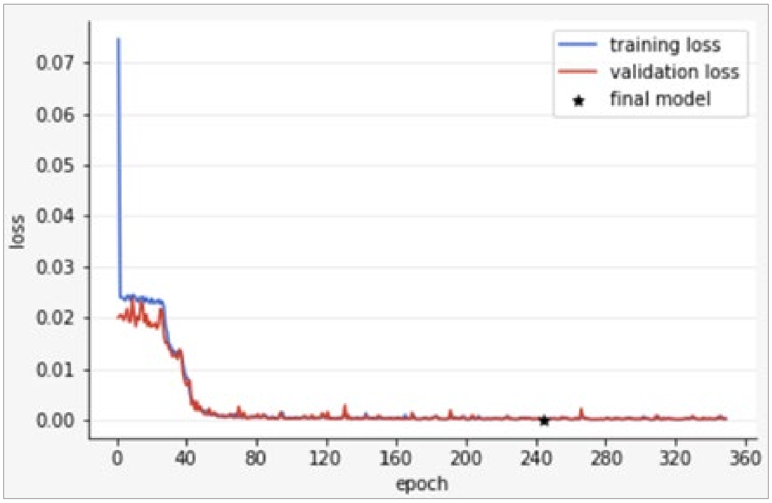

Figure 6. Loss curves during physicsAI machine learning model training

Next, the trained model was tested on a dataset unseen during training, i.e., the remaining 226 files generated by the DOE. The mean absolute error was found to be 1540 N.

The spread of the true and predicted KPI vectors for the test datapoints can be visualized as a distribution to get a visual representation of the phyicsAI model accuracy (Figure 7). Considering the magnitude of the predicted data, the average percent error is calculated to be 1.5%, satisfying the accuracy requirement of <10%.

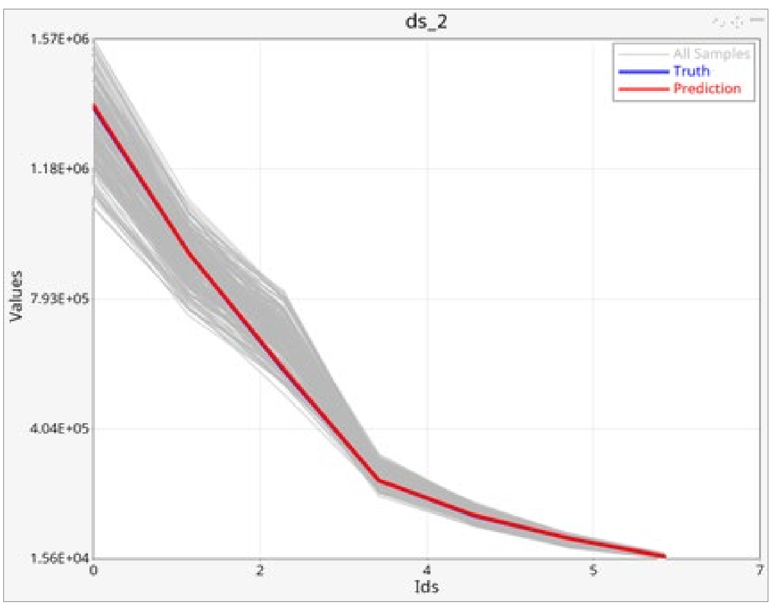

Figure 7. Distribution of the true and predicted KPIs

After establishing the accuracy of the physicsAI trained model, it is used as a surrogate model for a HyperStudy-driven optimization study (Figure 8). While traditional solver-driven optimization can be prohibitive for computationally heavy simulations, a physicsAI, machine-learning, surrogate-driven optimization is significantly quicker, making the optimization process possible.

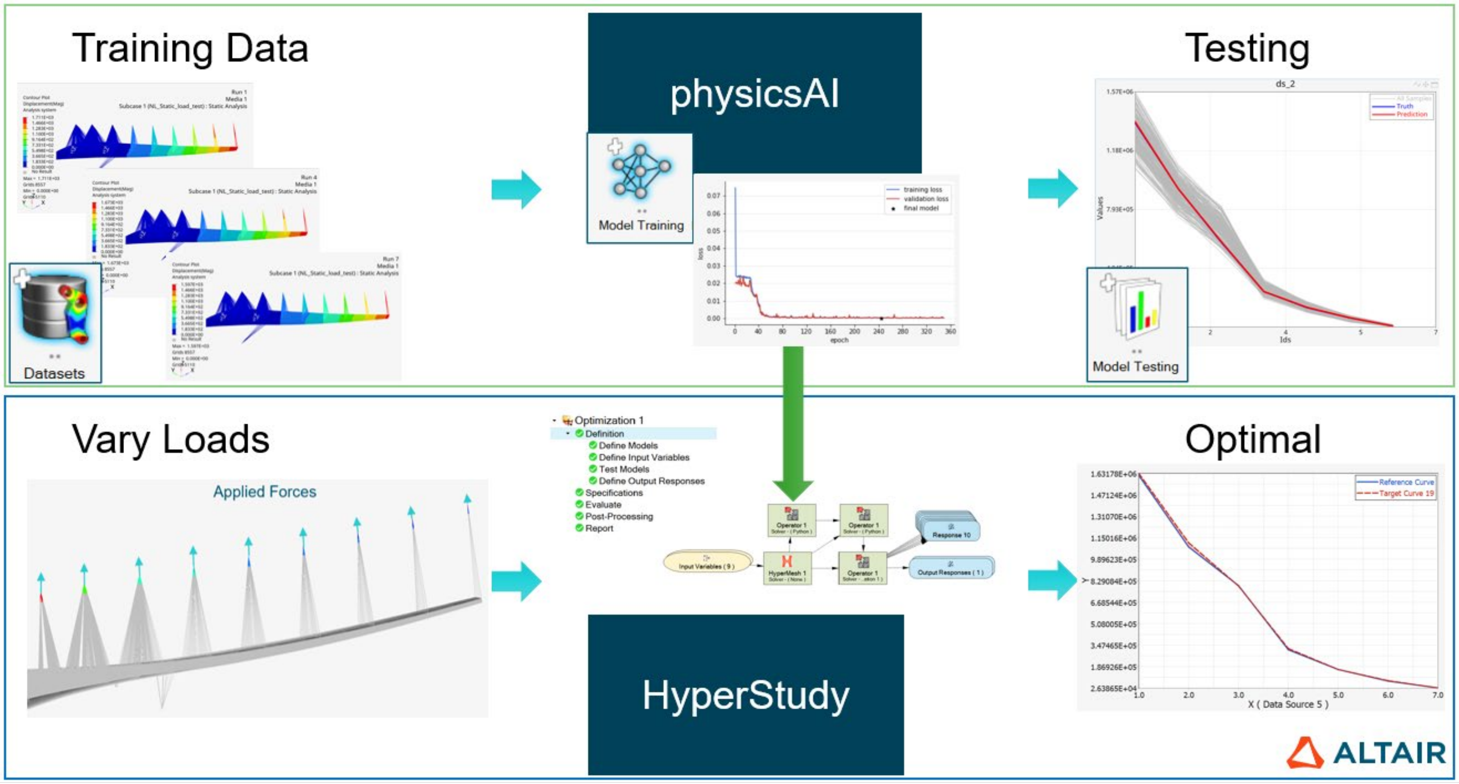

Figure 8. Optimization using a pre-trained physicsAI surrogate

Optimization Results

The total run time for this process was approximately 100 minutes with dataset creation being the longest segment (Figure 9), closely followed by training. physicsAI prediction time per iteration (~3 seconds) is found to be ~50x faster than OptiStruct (~150 seconds), greatly speeding up the optimization.

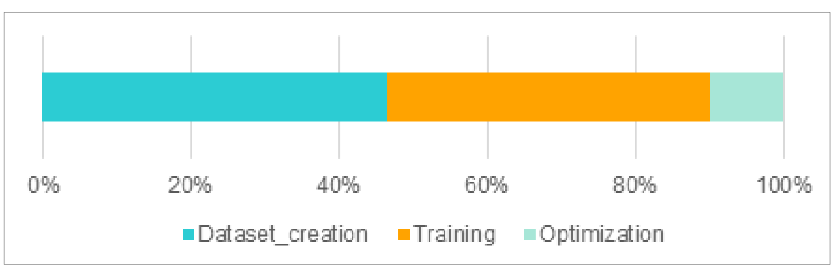

Figure 9. Time requirement for physicsAI tasks

The optimal solution was achieved in 19 iterations. The predicted load configuration was manually validated through an OptiStruct run and the results show a maximum deviation of 11%.

As seen in Figures 3. and 4., the highest percentage deviation corresponds to the lowest magnitude value, i.e., the lowest bending moment measurement location (Location 7) and the overall error is within acceptable limits.

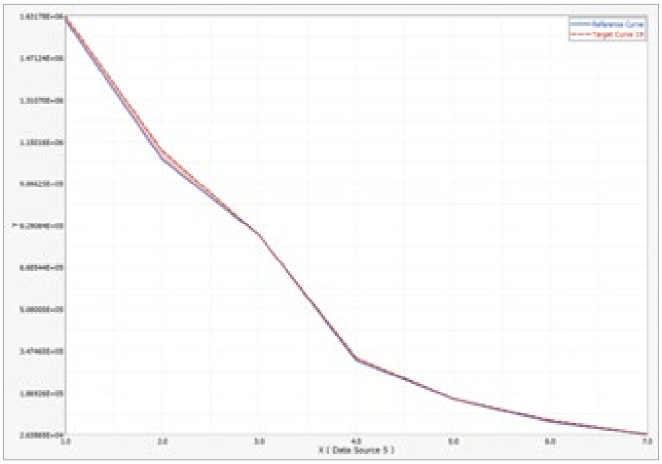

Figure 10. Optimal solution predicted using physicsAI driven optimization

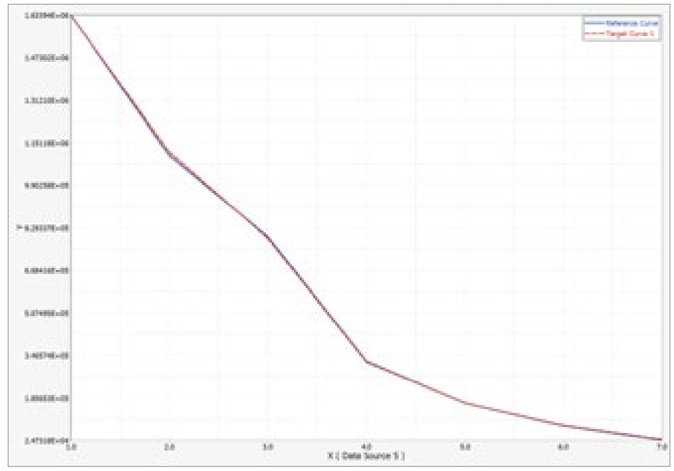

Figure 11. Optimal solution validated through an OptiStruct run

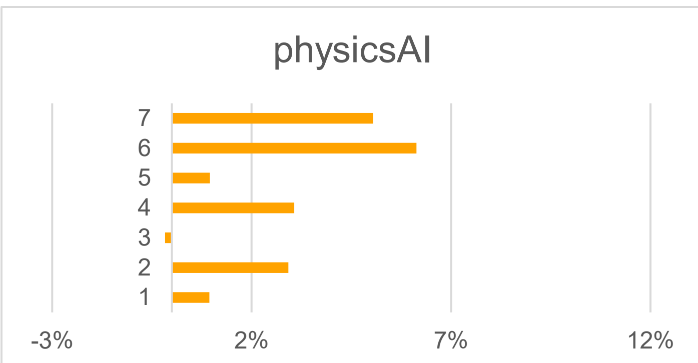

Figure 12. Deviation from the target curve: physicsAI predicted

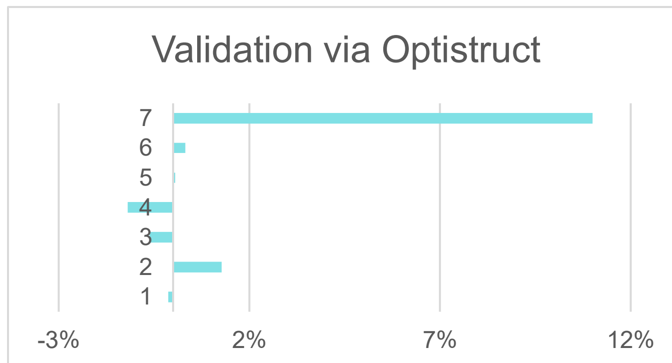

Figure 13. Deviation from the target curve: OptiStruct determined

Conclusion

In this paper, we successfully demonstrate the use of a physicsAI machine learning model as a surrogate in an optimization study to identify a configuration of input loads that put a wing into the desired state of BMD loading in simulated flight conditions. The physicsAI machine learning model is trained using synthetically generated 3D simulation models and is found to be 50x faster than the comparable FEA solver (OptiStruct). The optimal load configuration predicted by the physicsAI machine learning model closely matches the solution produced by OptiStruct, showcasing the former's accuracy.

Overall, this paper demonstrates the effectiveness of physicsAI machine learning models for accurately evaluating design load configurations in a fraction of the time required by a traditional FEA solver. This advantage can be leveraged for evaluating a large number of early-stage design variants, resulting in a much more comprehensive exploration of the design space than would be possible using a purely FEA solver-driven process. Additionally, physicsAI machine learning surrogates in iterative numerical optimization can greatly speed the process of arriving at the optimal solution. This is especially advantageous in the case of computationally heavy simulations where multiple evaluations would be prohibitive. Moreover, physicsAI machine learning models can be trained on having geometric and non-geometric variations, a major advantage over response surface models.

https://altair.com/physicsai