

GENERATIVE MODELING FOR FLAP KINEMATICS: ENHANCING DESIGN PIPELINES WITH VARIATIONAL AUTOENCODERS

A. Schreiber*, R. Y. Yanev*, I. Staack*, U. Scholz†

* TU Braunschweig, Institute of Aircraft Design and Lightweight Structures, Hermann-Blenk-Straße 35, Braunschweig

† Airbus Operations GmbH, Future Projects Office, Airbus-Allee 1, Bremen

Abstract

The preliminary design of high-lift configurations for fixed-wing aircraft is a complex, multi-disciplinary task usually relying on extensive Design of Experiment (DoE) studies. This work presents a Variational Autoencoder (VAE)-based pipeline for generating Adaptive Dropped-Hinge Flap (ADHF) concepts that satisfy engineering constraints on cruise drag, separation behavior, and actuation efficiency. Trained on a curated set of desirable designs, the VAE produces novel configurations with characteristics similar to proven solutions, enabling efficient exploration of the design space. A post-processing framework evaluates these candidates and relates them to the original user input. The ADHF-VAE pipeline delivers a broader and higher-quality range of designs in significantly fewer iterations than standard DoE, providing improved starting points for downstream aerodynamic, kinematic, and integration analyses. This framework represents a first step towards automated, data-driven synthesis of high-lift systems for future aircraft design.

Keywords

data-driven design; kinematics; generative modeling; design space exploration; variational autoencoder

1. INTRODUCTION

Preliminary aircraft design is the phase in which market requirements are translated into feasible concepts. This process involves refining the overall aircraft characteristics and architecture, including the definition of flight control devices on the wing. A key element in this context is the design of trailing edge flaps as part of the high-lift system. The mechanism, or kinematics, has a strong influence on performance, integration, and downstream design choices [1, 2]; however, its design typically requires several months of iterative exchanges among aerodynamics, kinematics, and integration disciplines.

Preliminary flap kinematics design relies on multi-disciplinary optimization processes and Designs of Experiment (DoE) for exploration [2–4]. While these conventional methods are established, they struggle with the vast, high-dimensional design spaces, which can potentially lead to inefficient exploration and the overlooking of favorable solutions. This has motivated the exploration of data-driven approaches capable of capturing complex design dependencies, thereby accelerating and reducing iterations.

The Variational Autoencoder (VAE) is a neural-network-based generative model, introduced in 2013 by KINGMA and WELING [5], and remains a versatile, widely used algorithm with a wide range of applications. Its stochastic properties in conjunction with the structured latent space, which is agnostic to downstream processing, have been exploited in engineering design as they allow the efficient exploration of large design spaces and the accelerated assessment of design performance through dimensionality reduction.

LEW and BUEHLER [6] proposed a reduced-order model comprising a VAE for reducing the dimensionality of a cantilever design problem, and a long short-term memory neural network for learning topology optimization trajectories within the latent design space. In a series of studies [7–10], SAHA et al. investigated the use of VAEs for conceptual

car design by learning latent representations of 3D point clouds. They developed an interactive design support system, providing guidance for design space exploration and real-time performance feedback. In a study conducted by WANG et al. [11], a conditional generative design method for airfoils based on Conditional Variational Autoencoders (CVAEs) was proposed. This approach allowed the airfoil design generation to be shape-conditioned (thickness and camber), aerodynamics-conditioned (pressure distributions, lift-to-drag ratio, and lift coefficient), as well as a combination of both.

DESHPANDE and PURWAR [12] proposed the variational synthesis approach involving VAEs as intermediaries between designer and computational kinematics solver for linkages. Their fundamental idea revolved around exploiting the input sensitivity of such solvers by generating a diverse set of novel inputs conditioned on the inherent knowledge of the model, starting with a single, uncertain user input.

In this paper, a novel design pipeline integrating VAEs into preliminary flap kinematics design is proposed. As probabilistic generative models, VAEs can learn compact lower-dimensional representations of design data and generate new candidates reflecting similar properties. This algorithm is employed as an input recognition and conditioning module in our work, preceding a processing pipeline for kinematics synthesis and evaluation, aiding designers in navigating complex design spaces while retaining control. By consolidating expert knowledge in such a framework, diverse sets of favorable candidate designs can be generated, thereby enriching design proposals in preliminary design. The results demonstrate the potential and challenges of data-driven design workflows and lay the groundwork for future research.

2. METHODOLOGY

To realize such a framework, multiple preliminary steps are required. To allow for an efficient representation of designs,

an effective form of parameterization is required. Furthermore, to consolidate and exploit expert knowledge with a generative model, a dataset is required that reflects such knowledge. As the utilization of data from operational aircraft is insufficient in terms of the number of data points, the dataset was synthetically generated. Finally, VAEs were implemented, trained, and evaluated.

2.1. Design Parameterization

For this study, the simplest state-of-the-art mechanism was selected, featuring a hinged flap with downward deflecting spoiler, as implemented on the *Boeing 787* and *Airbus A350*. On the A350, this mechanism is referred to as Adaptive Dropped Hinge Flap (ADHF), featuring multi-functional spoilers that are integrated for downward deflection, also known as droop, in addition to their conventional functionality as roll-control or lift-dumping devices [13].

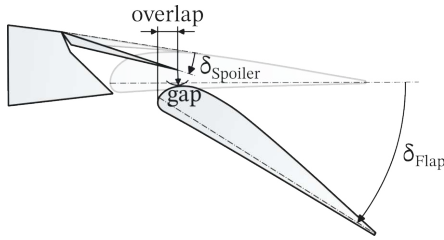


FIG 1. Aerodynamic parameterization of ADHF designs.

Fig. 1 depicts a common form of high-lift parameterization that was adopted for this work to represent ADHF designs in clean and target settings. The gap g is defined as the radius of a circle centered on the spoiler's trailing edge that tangentially intersects with the flap. Conversely, ovl is the overlap of flap and spoiler, measured between the spoiler trailing edge and the flap leading edge. Flap and spoiler deployment angles δ_F and δ_{Sp} , respectively, are measured relative to the retracted position. Flap and spoiler chords were kept constant for this study, with $c_F = 0.23 c_W$ and $c_{Sp} = 0.12 c_W$, respectively, relative to local wing chord c_W . A representative wing section at the inboard flap field support station of a medium-range aircraft was used.

In this work, the term *design option* refers to a five-dimensional feature vector that defines an ADHF design:

$$\mathbf{x} = \begin{bmatrix} ovl_{clean} \\ ovl_{target} \\ g_{target} \\ \delta_{Sp} \\ \delta_F \end{bmatrix}.$$

The indices *clean* and *target* represent the retracted and the fully deployed setting of an ADHF design, respectively. Note that δ_F and δ_{Sp} both represent the target setting - the indices were omitted for clarity. Note that due to the simplicity of this mechanism, a design is fully defined based on the mentioned two settings and does not allow for the definition of additional intermediate settings, e.g., for take-off.

2.2. Dataset Generation with Engineering Constraints

The data generation comprised several steps, as illustrated in fig. 2. First, a DoE study was carried out to gather a large initial design population, uniformly distributed across the design space. Subsequently, each design option was synthesized into an ADHF design and optimized regarding the actuator position. Key Performance Indicators (KPIs) were then derived from each design and used to filter the design population according to three defined engineering constraints. The final population comprises exclusively favorable ADHF designs, according to the defined objectives.

2.2.1. Design of Experiment

A total of $32768 = 2^{15}$ design options were sampled utilizing Sobol' sequences [14]. Due to its low-discrepancy property, this method quickly converges to a uniform distribution of samples in the design space, which is a desirable property for the generative modeling task, in which the sample distributions is learned. The employed parameter ranges for each feature are depicted in tab. 1.

feature	unit	min. value	max. value
ovl_{clean}	[% c_W]	7	9
ovl_{target}	[% c_W]	1	2
g_{target}	[% c_W]	1	2
δ_{Sp}	[°]	6	12
δ_F	[°]	30	45

TAB 1. Selected parameter range for each design feature.

2.2.2. Kinematics Synthesis and Optimization

The general objective of kinematics synthesis is to identify one or more linkage mechanisms that fulfill a certain task [15]. In the context of flap kinematics, the task is that of motion generation, also known as rigid-body guidance, which involves identifying link lengths and joint positions. For the ADHF mechanism, there are two unknown joints: the actuator and the hinge. While the hinge point can be uniquely determined from a design option $\mathbf{x}$ with methods

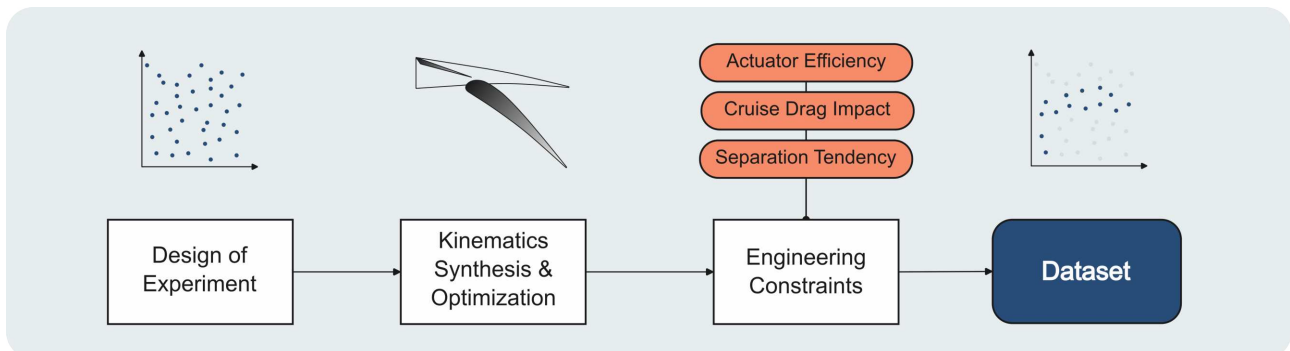


FIG 2. Schematic of data generation pipeline.

such as two-position synthesis [16], the actuator position is driven by design constraints such as the available installation space and the actuator load. Therefore, identification of the actuator position was formulated as an optimization problem.

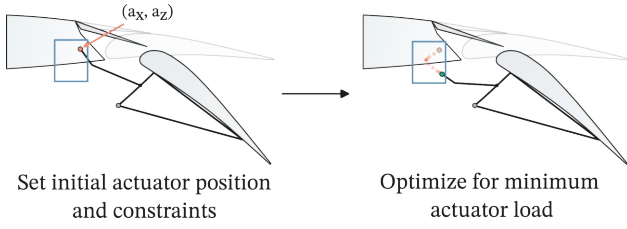


FIG 3. Actuator position optimization within the available wing trailing edge integration space.

The optimization design variables were the x and z-coordinates of the actuator joint, a_x and a_z , respectively. As illustrated in fig. 3, constraints in the form of upper and lower bounds on the design variables were imposed. The objective to be minimized was the actuator load, computed based on the aerodynamic flap loads. The actuator lever arm was kept constant, resulting in a fixed ratio between the perpendicular load on the lever T_{act} and the resulting actuator moment M_{act} . The drive strut length, on the other hand, which is the link attaching the lever arm to the flap truss structure, was variable and determined by a dedicated computational ADHF solver taking into account the displacement of all elements in the kinematic chain. The sectional flap load components are defined as:

$$(1) \quad N_F = c_N \cdot q_\infty \cdot A_F$$

$$(2) \quad T_F = c_T \cdot q_\infty \cdot A_F$$

in which N_F and T_F correspond to normal and tangential forces, respectively, acting in the center of pressure relative to the chord line. The corresponding force coefficients are denoted c_N and c_T . A unit width was utilized to compute the flap area $A_F = c_F \cdot 1 m$. The ambient dynamic pressure q_∞ was set according to flap design speeds of a reference aircraft (A350-1000), as per airworthiness requirements §25.335 (e) in the EASA CS-25 [17]. Regression relationships to compute the load coefficients and center of pressure over δ_F were kindly provided by Airbus engineers. Altogether, this enabled the simulation of resulting flap loads F_F over δ_F .

The objective function was defined as follows: Given an actuator position (a_x, a_z) , the ADHF solver synthesizes the lever arm and drive strut. A load solver subsequently com-

putes the actuator loads based on flap loads at deflection angles $\delta_F \in [19^\circ, \max(40^\circ, \delta_{F,target})]$ with a step size $\Delta\delta_F = 3^\circ$. The objective function value then is the maximum absolute actuator load $\max(|T_{act}|)$ within the evaluated deflection angle range.

2.2.3. Actuator Efficiency

Based on the optimized actuator position, the first metric used to constrain ADHF designs was computed: the ratio of actuator load to flap load R_{act} . It was defined as:

$$(3) \quad R_{act} = \frac{T_{act}}{F_F} \Big|_{\delta_{F,max}} \quad \text{with } \delta_{F,max} = \underset{\delta_F}{\operatorname{argmax}} |T_{act}(\delta_F)|$$

R_{act} was used to characterize the efficiency of load transfer, as it represents the multiples of the flap load the actuator must generate to move the mechanism. A larger R_{act} indicates less efficient actuation, as it requires a more powerful actuator to generate the required load. This increase in actuator size results in higher weight and requires more installation space, which imposes additional design constraints. Engineering experience suggests that actuators start to become ineffective beyond ratios of 2, yielding the actuator efficiency constraint:

$$R_{act} < 2.0$$

2.2.4. Fairing Drag

The second applied constraint limits the cruise drag increment due to the fairing streamlining the mechanism. Fairing shapes were generated based on standard design principles followed at Airbus, from which the wetted area could be inferred. A relationship between wetted area, wing area, and skin friction coefficient was employed to compute the cruise drag increment. The constraint value was determined based on its effect on the feature distribution. As a much larger portion of designs were discarded through this constraint, certain feature value ranges were fully neglected. To this end, an initial reference threshold was relaxed until this behavior was no longer observed, yielding the fairing drag constraint as:

$$\Delta C_{D, \text{fairing}} < 0.9$$

With this threshold, the sample density regarding δ_F plateaued between 42° and 45° (see fig. 4). While solutions with larger δ_F are desirable with respect to fairing size, there are trade-offs, such as for take-off performance, that must be considered. To simulate an adversarial, a penalty was introduced by amplification of the cruise drag increment of design options with $\delta_F \geq 40^\circ$ with a penalty factor λ . The

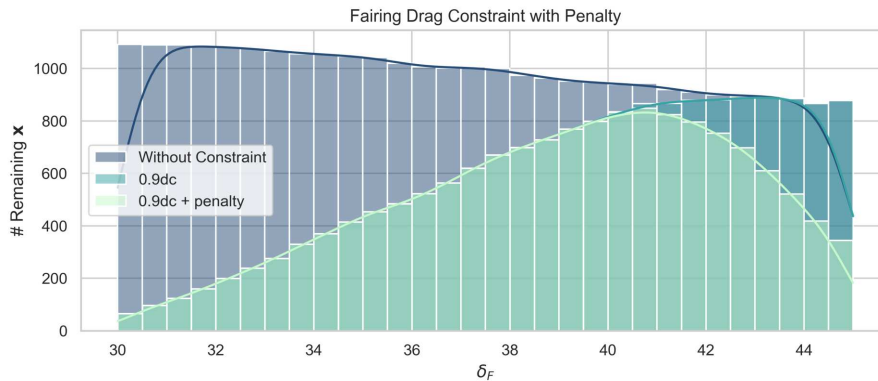


FIG 4. Effect of the fairing drag constraint with and without flap angle penalty on the distribution of flap target angles δ_F .

underlying function is of the form:

$$(4) \quad \lambda(\delta_F) = \begin{cases} 1 & , \text{if } \delta_F < 40^\circ \\ 0.02 (\delta_F - 40)^2 & , \text{otherwise} \end{cases}$$

As can be observed in fig. 4, the δ_F -distribution with penalty reflects that smaller δ_F are not desirable due to large fairing drag increments, and larger δ_F are not desirable due to compromised take-off performance.

2.2.5. Separation Tendency

The third engineering constraint was based on two metrics characterizing the flow separation tendency to evaluate the slot geometry: suction peak $c_{p,\min}$ and maximum boundary layer form factor $H_{12,\max}$. A numerical solver was employed, specifically developed to compute the incompressible flow around high-lift configurations by the iterative coupling of an inviscid panel code solution with a viscous integral method solution for boundary layer and wake.

A design option $\mathbf{x}$ in target configuration comprised the following three elements:

- 1) A slat in target setting (constant for all $\mathbf{x}$): $g_{\text{target}} = 1.5\% c_W$, $ovl_{\text{target}} = 0.0\% c_W$, $\delta_S = 28^\circ$
- 2) A flap in target setting (given by $\mathbf{x}$)
- 3) The effective main airfoil shape resulting from the deployment of all devices, including spoiler droop

The approach was to determine for each ADHF design $\mathbf{x}$ in high-lift configuration, whether it offers stable aerodynamic performance throughout multiple operating points. For that, a range of angles of attack $\alpha = [0^\circ, 3^\circ, 6^\circ, 9^\circ, 12^\circ]$ was computed for a Reynolds number $Re = 2.1 \cdot 10^7$ and a Mach number $Ma = 0.20$. Based on literature [2, 18] and engineering knowledge, the computed pressure and form factor distributions over the devices c_p and H_{12} , respectively, were used to constrain the remaining design options, after applying the preceding two constraints:

$$\begin{aligned} c_{p,\min} &> -10.0 \\ H_{12,\max} &< 4.0. \end{aligned}$$

If an ADHF design $\mathbf{x}$ violated at least one of the constraints for any operating point (α , Re , Ma), it was discarded from the dataset. The $c_{p,\min}$ constraint, which is primarily a separation criterion relevant at maximum lift [2], did not become active since the maximum angle of attack of 12° remained below this condition, but was nevertheless retained as a lower bound.

2.2.6. Post-Processing

After applying each engineering constraint, 6648 ADHF designs, 20.3 % of the initial population, remained in the filtered dataset. Fig. 5 displays individual feature distributions compared to the initial uniform distribution. The actuator

efficiency constraint primarily discarded designs yielding larger lever arms between flap load and hinge point, requiring more powerful actuation. The fairing drag constraint predominantly discarded designs with larger ovl_{clean} and smaller δ_F , as they drive hinge point depth and therefore increase fairing size. The form factor constraint uniformly shaped the features with respect to their prior distribution.

The dataset was finally standardized and split into three parts: 70 % (4653 samples) for training, 15 % (997 samples) for validation, 15 % (998 samples) for testing.

2.3. Variational Autoencoder of Design Options

VAEs are machine learning algorithms based on neural networks that can be trained with gradient-based optimizers. In our work, the encoder compresses ADHF designs into a reduced, or latent, representation, while the decoder reconstructs the original design from this representation, as illustrated in fig. 6. The bottleneck is denoted the latent space and resembles a lower-dimensional proxy of the original design space learned during training.

VAEs are trained by maximizing the Evidence Lower Bound (ELBO), which is typically expressed as a loss function $\mathcal{L}$ to be minimized during training [19, 20]:

$$(5) \quad \mathcal{L}_{\text{VAE}}(\theta, \phi; \mathbf{x}) = -\text{ELBO} = \underbrace{\beta \cdot D_{\text{KL}}(q(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z}))}_{\text{regularization term}} - \underbrace{\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction term}}$$

Eq. 5 represents the loss with respect to the encoder weights ϕ and the decoder weights θ for a training sample, i.e., ADHF design option, $\mathbf{x}$ and its latent representation $\mathbf{z}$. The first term on the right side of eq. 5 represents the Kullback-Leibler (KL) divergence, quantifying the difference in information content of two probability distributions [21]. It ensures that the learned posterior distribution $q(\mathbf{z}|\mathbf{x})$ is similar to the prior distribution $p(\mathbf{z})$, which in our case was chosen to be unit Gaussian. To control the capacity of the latent space, a hyperparameter β can be multiplied with the KL term, as originally proposed by HIGGINS et al. [22]. The second term on the right side of eq. 5 represents the reconstruction loss, which was implemented as the Mean Squared Error (MSE) between input $\mathbf{x}$ and reconstruction $\hat{\mathbf{x}}$.

A VAE with a fully-connected neural network architecture was implemented in *Python*, based on the open-source deep learning library *PyTorch* [23]. After an initial study seeking to find a suitable architecture, early-stopping and plateau-learning-rate-scheduling were implemented with a moving loss average over 20 epochs to increase the efficiency and robustness of training. Finally, a Bayesian hyperparameter optimization was carried out with *Optuna* [24] to identify optimal settings for initial learning rate and batch

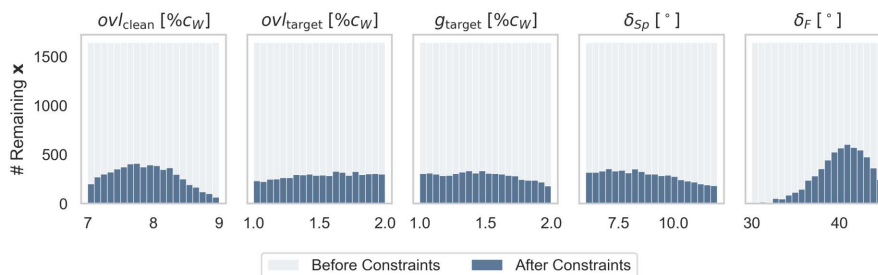


FIG 5. Feature distribution in the dataset after applying all constraints.

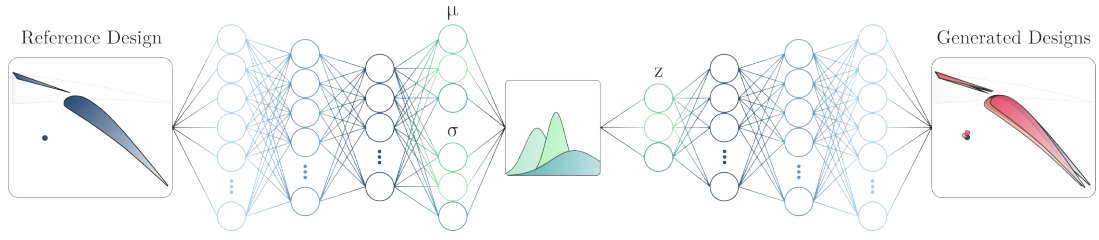


FIG 6. Adaptive Dropped Hinge Flap Variational Autoencoder: Based on a reference design, designs with similar properties can be generated.

size. Details about hyperparameters are provided in tab. 2. The latent dimensionality was fixed to 3 with $\beta = 10^{-2}$ throughout the experiments.

optimizer	AdamW
initial learning rate	$6 \cdot 10^{-3}$
batch size	128
maximum epochs	30,000
encoder neurons	(64, 32)
decoder neurons	(32, 64)
activation	Leaky ReLU
encoder dropout probability	0.05
early-stopping patience	1500 epochs
scheduler patience	1500 epochs
moving average window size	20 epochs

TAB 2. VAE hyperparameter settings.

2.4. Latent Space Analysis

While reconstruction quality is quantifiable via metrics such as MSE and R^2 , ensuring a high quality of the latent design space requires assessment of the latent space structure. To this end, a parameter sweep over latent space dimensionality $d = [2, 3, 4]$ and regularization strength $\beta = [0, 10^i]$ with $i = -7, \dots, 0$ was carried out.

All trained models were first evaluated using MSE and R^2 on the test set, as illustrated in fig. 7. For $\beta \leq 10^{-3}$, reconstruction performance is consistently high. From $\beta = 10^{-2}$, a steep performance drop can be observed towards $\beta = 1$, at which reconstruction performance reaches a minimum of $R^2 = 0\%$. Increasing d improves performance due to the reduced information bottleneck.

The second step of this analysis was the qualitative evaluation of the latent space structures. To this end, the dataset was encoded with each configuration, and the resulting la-

tent representations were visualized in 2D scatter plots. For latent dimensions $d > 2$, the two most active latent dimensions were selected, based on their contribution to reconstruction accuracy R^2 . Fig. 8 displays latent space structures for a subset of model configurations. The structural characteristics vary significantly with β , as lower values impose less regularization on the encoder for shaping the latent distribution, while higher values increasingly force it to learn a unit Gaussian latent design space. For $\beta = 1$, the posterior collapse phenomenon was observed, i.e., the posterior collapses onto the unit Gaussian prior distribution. In that case, sample locations are invariant with respect to the input, and models tend to produce generic outputs such as the dataset mean [25].

Although superior reconstruction performance was achieved with the largest bottleneck $d = 4$ and weak encoder regularization $0 \leq \beta \leq 10^{-4}$, corresponding latent representations lack completeness. Gaps or "blank" regions may yield unpredictable results, particularly for unseen inputs projected into these areas. For $d = 2$, no configuration yielded fully satisfactory results, with R^2 around 81% - our objective was 90%. With $d = 3$, most configurations achieve $R^2 = 90\%$, however, their latent spaces are not regular. For $d = 4$, configurations with $\beta = 10^{-3}$ and $\beta = 10^{-2}$ stand out.

To conclude the analysis, the three most promising configurations, $d = 4, \beta = 10^{-3}$; $d = 4, \beta = 10^{-2}$, and $d = 3, \beta = 10^{-2}$, were examined in detail. The two most active dimensions of their latent spaces per feature are displayed in fig. 9. For the $d = 4, \beta = 10^{-3}$ model (fig. 9a), which achieved near perfect reconstructions with $R^2 = 97.3\%$, encodings are mostly regular, except a discontinuity for g_{target} that may cause larger spread for this feature when sampling in its vicinity. The $d = 4, \beta = 10^{-2}$ model with $R^2 = 95.0\%$ (fig. 9b) shows good structure overall, but there is also a vertical discontinuity apparent, particularly for δ_F . The $d = 3, \beta = 10^{-2}$ model with $R^2 = 90.7\%$ (fig. 9c) exhibits an incomplete latent structure with three separate regions, although the feature values vary continuously over the regions. The

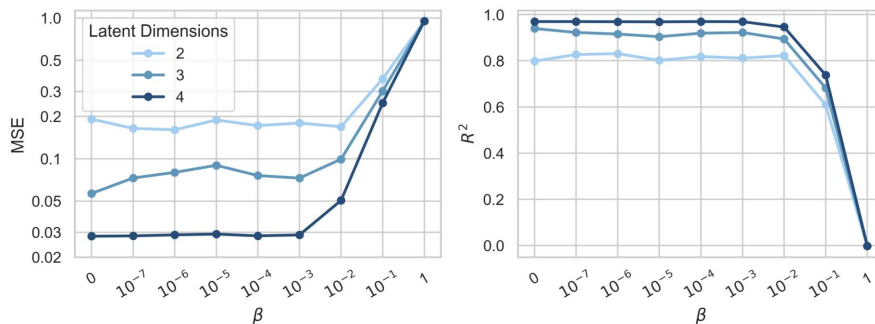


FIG 7. Reconstruction performance of different model configurations, varying the latent space dimensionality d and regularization strength β .

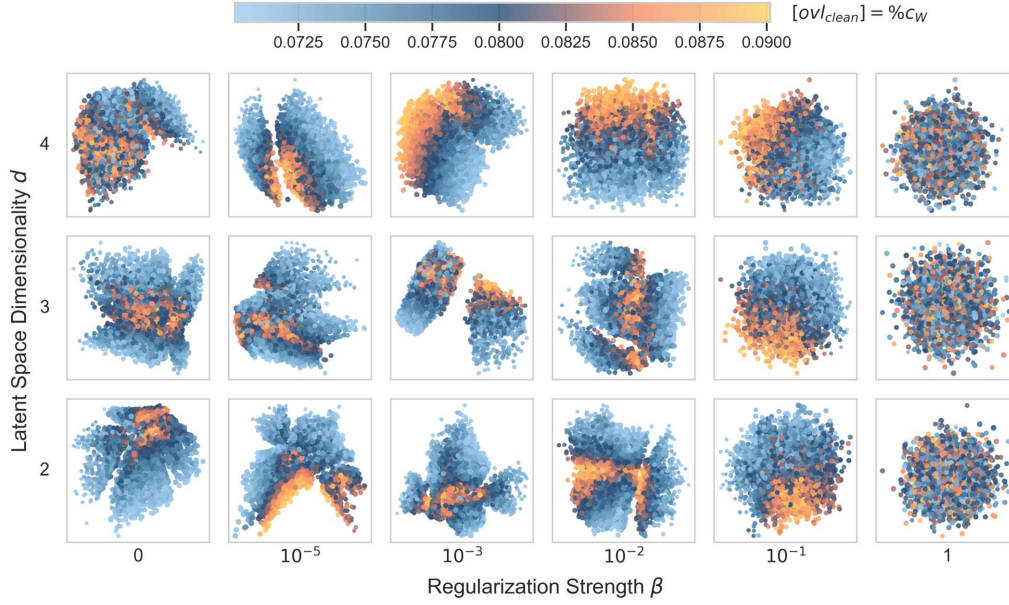


FIG 8. Dataset encoded into the two most active latent variables of different model configurations, varying the latent space dimensionality d and regularization strength β . The color of each latent vector corresponds to the clean overlap ovl_{clean} (a) and flap angle δ_F (b). Similarly, the marker size was adjusted according to the target gap g_{target} (a) and target overlap ovl_{target} (b). The value ranges per plot can differ significantly.

encoding for ovl_{target} is noisy, suggesting this feature may not be represented well by the displayed dimensions.

In summary, the $d = 4$ models are likely to perform well in terms of reconstructing known inputs, but may lack diversity when generating new samples. In contrast, the $d = 3$ model may lead to more diverse outputs, though less accurate in generating designs similar to the reference design. Investigating the physical structure of each latent dimension and the encoding of each feature can be a time-consuming process and is only one aspect of deciding on a suitable set of hyperparameters. For subsequent analysis, the $d = 4$, $\beta = 10^{-3}$ model was excluded due to its large R^2 , which offered too little design diversity.

3. RESULTS

3.1. Design Generation

A selection of superior model configurations was tested on the basis of two distinct ADHF design scenarios. The first simulates a user input in form of a reference design, corresponding to the starting point for design space exploration. The second scenario, conversely, involves providing explicit performance targets to assess the models' abilities to generate diverse designs that satisfy these prescribed objectives.

3.1.1. Design Target

The design target scenario is represented by an unfavorable reference design. This scenario is specifically characterized by a larger ovl_{clean} and a smaller δ_F , which were particularly penalized during data generation due to their contribution to large fairing drag increments. The representative design

option for this low-density feature region was defined as:

$$\mathbf{x} = \begin{bmatrix} ovl_{clean} \\ ovl_{target} \\ g_{target} \\ \delta_{Sp} \\ \delta_F \end{bmatrix} = \begin{bmatrix} 8.5\% c_W \\ 1.3\% c_W \\ 1.8\% c_W \\ 6.5^\circ \\ 36.0^\circ \end{bmatrix}$$

Based on this reference design, 20 novel designs were generated. Fig. 10 displays ADHF designs with their respective feature values and KPIs generated by both models. Both models perform a correction of the reference design, for instance, by generating designs with larger flap angles. This is because the input is not part of the learned data manifold of favorable ADHF designs, as the reference design does not adhere to two of the constraints. While the $d = 4$ model (fig. 10a) generates designs with less deviation to the reference and more diversity, the $d = 3$ model (fig. 10b) consistently generates favorable designs, however, with only little diversity.

3.1.2. Performance Target

The second scenario explores design generation based on specific target performance objectives, reflecting a designer's direct preference. Unlike the previous scenario that relied on defining a reference design as input, in this case, KPIs serve as the basis for design generation. The representative performance target was defined as:

$$\begin{aligned} R_{act} &= 1.5 \\ \Delta C_{D, fairing} &= 0.7 \\ H_{12, max} &= 3.5 \end{aligned}$$

To identify a suitable design option corresponding to this target, a nearest neighbor search was performed using the KPIs of each design in the dataset. To ensure equal weighting, each KPI was first standardized. The Euclidean distance to the query point (representing the performance tar-

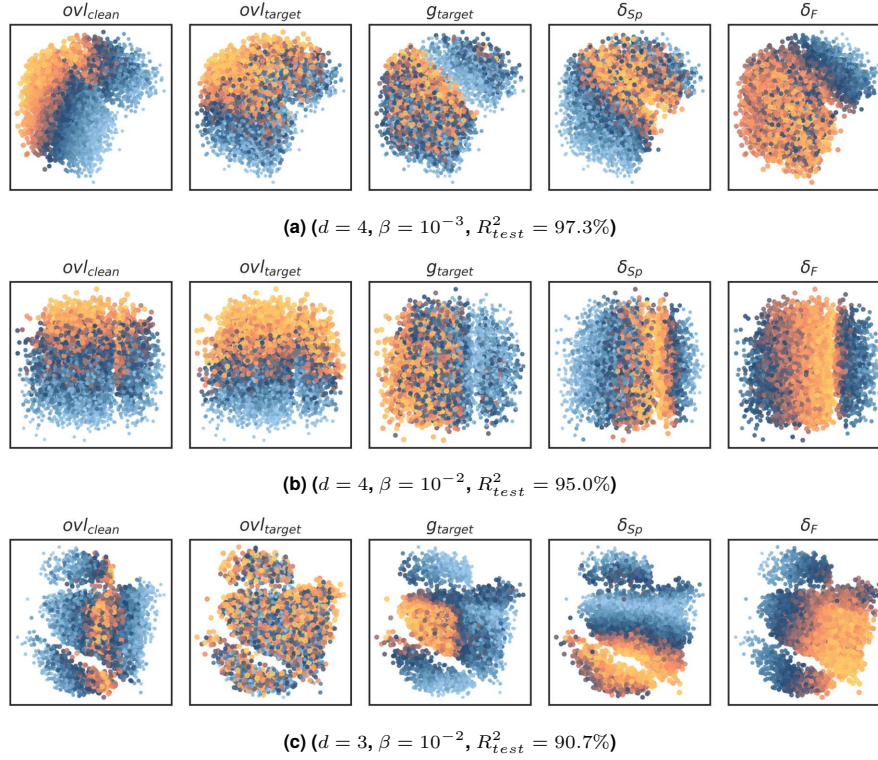


FIG 9. Dataset encoded into the two most active latent variables of the three best-performing model configurations.

get) was then computed. The design option associated with the smallest Euclidean distance to this query point served as the reference input, thereby indirectly representing the user's desired performance. For this scenario, the identified reference design was:

$$\mathbf{x} = \begin{bmatrix} \text{ovl}_{\text{clean}} \\ \text{ovl}_{\text{target}} \\ \text{g}_{\text{target}} \\ \delta_{\text{Sp}} \\ \delta_{\text{F}} \end{bmatrix} = \begin{bmatrix} 7.8\% c_W \\ 1.3\% c_W \\ 1.1\% c_W \\ 6.1^\circ \\ 40.8^\circ \end{bmatrix}.$$

Fig. 11 displays 20 ADHF designs generated with both models based on the performance target. In contrast to the previous scenario, there is significantly less deviation apparent, since the input is always part of the learned data manifold with this approach. Both models reflect the performance target well and offer a range of similar-performing designs with feature values spread around the reference. Although the design diversity is similar in this case, the $d = 3$ model outputs (fig. 11b) exhibit larger offset to the reference design and performance, and more often violate a constraint - H_{12} in this case.

Testing model candidates on various design generation scenarios offered valuable insights into their characteristics. The $d = 4$ model outperformed the $d = 3$ model in terms of reconstruction accuracy and generative diversity. Although $d = 4$ represents a comparably large bottleneck compared to the $d = 5$ original design space, the intricate relationships within the dataset and the specific test case require such a configuration. For the remaining part of the analysis, the $d = 4$ model will therefore be used for further investigations and experiments.

3.2. Latent Performance Maps

Apart from the generative use of VAEs in design pipelines, the learned latent design space can be used as a lower-dimensional proxy of the higher-dimensional, original design space. Fig. 12 displays the latent representation color-coded by each of the KPIs used during dataset generation. Again, utilizing the two most active latent dimensions for visualization, this offers a natural interface for humans to understand and explore trade-offs.

The performance map displayed in fig. 12, learned by the $d = 4, \beta = 10^{-2}$ VAE, reveals, for instance, that actuation efficiency R_{act} and fairing drag $\Delta C_{D,\text{fairing}}$ act adversely. Designs with lower $\Delta C_{D,\text{fairing}}$ yield shallower hinge points, resulting in configurations with less efficient load transfer to the actuator in the specific use case of our work.

3.3. Model Temperature

To increase the flexibility of VAEs as intermediaries between designer and kinematics pipeline, the model temperature is proposed, inspired by large language models. It is intended to enable designers to request either a larger variance in model outputs, i.e., the model is more "creative", or lower variance, i.e., the user is more certain about the provided reference input and desires only little diversity. To incorporate the temperature T into the generative process, the reparameterization trick proposed by KINGMA and WELLING [5] was extended to scale posterior standard deviation by T .

Low temperature $T \in [0, 1)$ can be used to enforce more deterministic model behavior, whereas high temperature $T \in (1, \infty)$ can be used to enforce higher variance in generated designs. The parameter was set to $T = 1.0$ during model training.

To investigate its effect, a temperature sweep was conducted across values $T = [0.5, 1.0, 2.0, 3.0, 5.0, 8.0]$. The performance target from section 3.1.2 served as the reference input, based on which the posterior was inferred, and

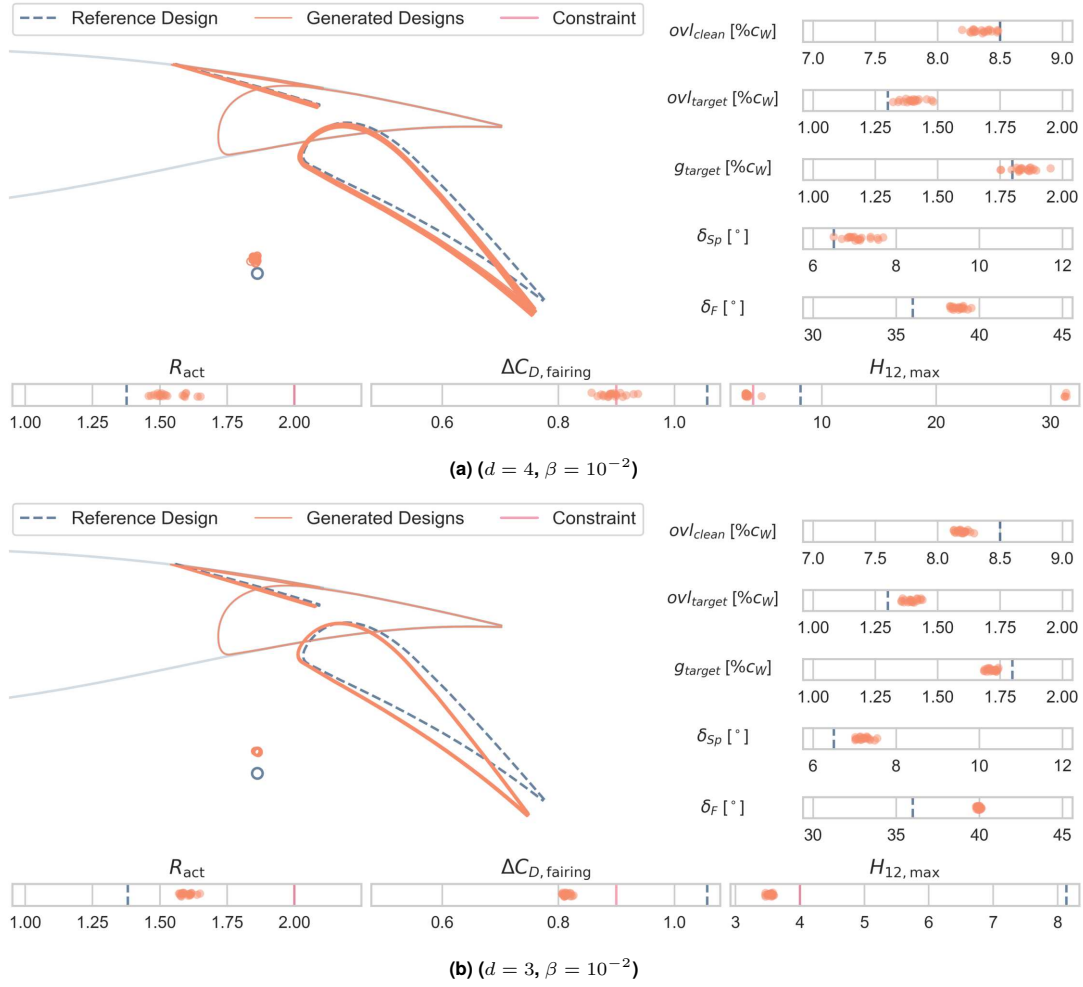


FIG 10. ADHF designs generated with each configuration. A reference design option $\times$ was fed to each model, representing low-density feature values.

1000 samples were drawn. Fig. 13 displays kernel density estimates of the sample distributions and corresponding ADHF designs for a subset of temperatures.

For $T = 0.5$, the posterior standard deviation is halved with respect to the default temperature $T = 1.0$, causing a concentration of samples within a smaller area. Synthesized designs show high similarity with minimal variance. The temperature can be arbitrarily increased, corresponding to a broader exploration of the latent design space. For instance, at $T = 5.0$, the samples span approximately one-third of the latent space, yielding a large variety of designs. It is important to note that the effect of model temperature is not universal, but may vary between model configurations and inputs. However, obtaining a quick overview of the temperature effect, as shown in fig. 13, is computationally inexpensive for this model size and can be computed in a fraction of a second to set the desired level of design diversity before the computationally expensive post-processing steps.

3.4. VAE vs. Conventional DoE

The final step of this work was a comparative analysis between the proposed VAE-based design pipeline and a conventional DoE-based pipeline. The conventional approach is based on Sobol' sampling, rather than a generative model. A global and a local design space exploration scenario were considered for the comparison.

3.4.1. Global Exploration

In the first scenario, 128 design samples were generated within the global design space (see tab. 1) with both approaches. The VAE pipeline was provided with a representative design containing the mean value of each feature's range and a temperature setting of $T = 6.0$.

Fig. 14 shows the distributions of feature values explored with either method, as well as the KPIs of corresponding ADHF designs. While the conventional DoE method yields a uniform distribution across the design space, the VAE-generated samples cluster in high-interest regions. Regarding KPIs, most VAE-generated designs exhibit acceptable R_{act} and $\Delta C_{D, fairing}$ below their respective constraint value. However, a total of 45.3 % of the designs violate any one constraint, with 42.2 % of these violations attributed to $H_{12, max}$. As for the conventional pipeline, a substantial 82.0 % of designs violate any one constraint, rendering the majority of designs invalid. While violations of the actuator efficiency constraint constitute a small fraction (8.6 %), the majority of violations are attributable to the fairing drag constraint (39.8 %) and the separation tendency (72.7 %).

3.4.2. Local Exploration

In the second scenario, the quality of designs produced through local design space exploration was compared. The target performance design scenario (see section 3.1.2) served as the reference design for both approaches. VAE

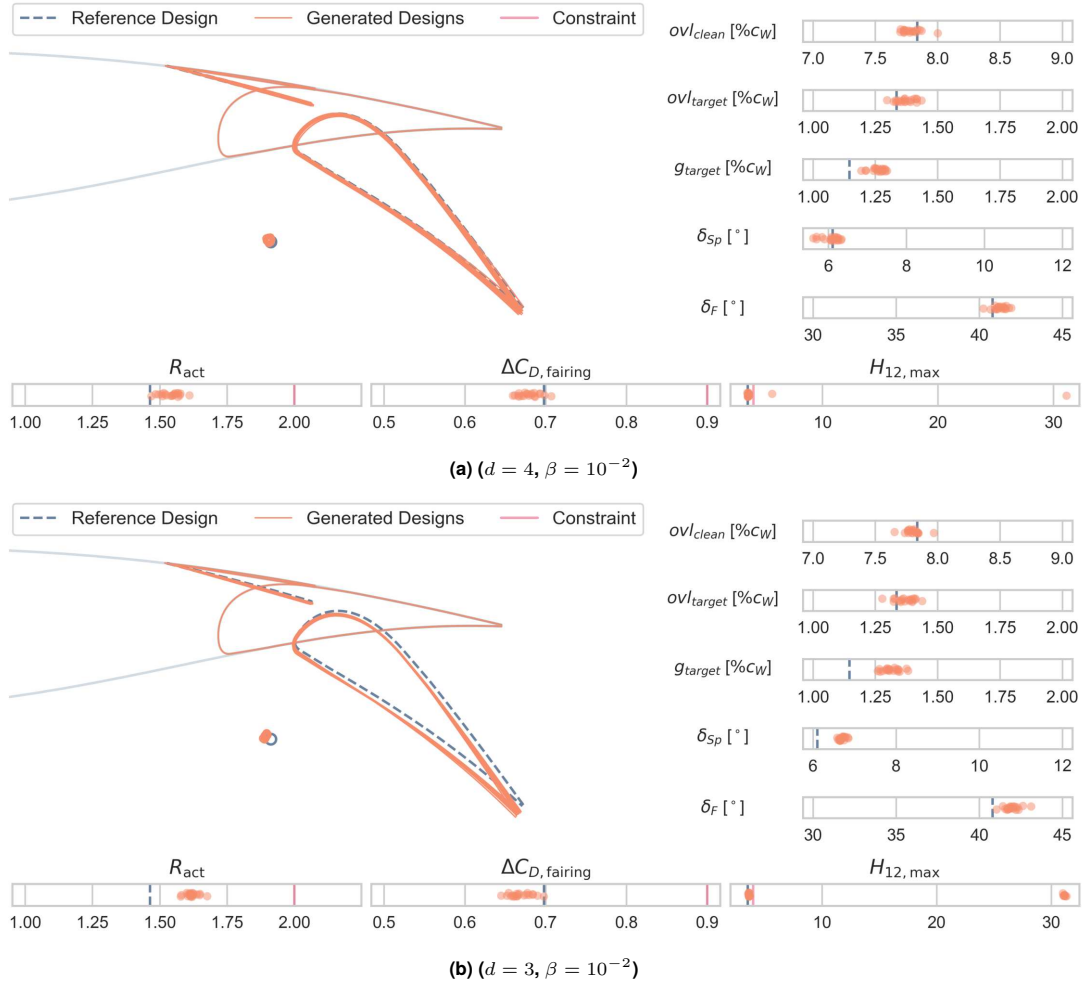


FIG 11. ADHF designs generated with each configuration. A reference design option $\times$ was fed to each model, representing a performance target.

temperature was set to $T = 2.0$ in this case and the design space for the conventional DoE was defined as $\pm 10\%$ of the full feature range (see tab. 1) relative to the reference input, resulting in the feature distributions shown in fig. 15. While the conventional DoE draws samples uniformly around the reference value, the VAE tends to generate designs with larger target gaps and flap angles. The performances of ADHF designs generated with both approaches differ specifically for H_{12} , for which all violations occurred. A total of 31.3% of designs generated with Sobol' sampling and 8.6% of the VAE-generated designs violate the H_{12} constraint. This underlines that these two KPIs are considerably more predictable and change smoothly across feature values. Conversely, the underlying processes determining separation behavior are far more complex and depend on feature interactions. In such cases, the VAE's advantage lies in its ability to learn these relationships from the training data distribution, thereby producing fewer invalid designs.

The previous comparison demonstrates the fundamental difference between conventional and data-driven design space exploration. The VAE performs exploration within a latent space that represents the data manifold of the training data, inherently focusing on favorable designs. Consequently, it is significantly less prone to generating invalid designs, although it requires time-consuming training and evaluation.

4. CONCLUSIONS

This work presented a generative, data-driven pipeline for aircraft flap kinematics design, demonstrating the applicability of VAEs in capturing complex design dependencies and efficiently generating diverse candidate designs. Compared to conventional DoE, the VAE-based approach enables more efficient exploration of the design space, supports intuitive visualization through latent performance maps, and allows designers to tune solution diversity with model temperature. These contributions highlight the potential of generative models to accelerate preliminary design and improve cross-disciplinary communication in early stages. Future research may focus on scaling and adapting the framework to increase fidelity and design freedom. This could include incorporating additional KPIs (e.g. take-off performance), adding shape parameters (e.g. flap and spoiler chords), and expanding the approach to other relevant kinematics types (e.g. track mechanisms). Conditional generative models (e.g., CVAEs) could be employed to guide sampling based on wing section characteristics (e.g. spanwise position), while surrogate models may address the current KPI evaluation bottleneck and enable real-time exploration.

The long-term vision is to move beyond sectional flap kinematics towards full 3D layouts, integrating multiple flaps and leading-edge devices within a conditional generative framework. Although this raises challenges related to data re-

quirements and computational effort, the efficiency gains of generative models are expected to become increasingly decisive as design spaces grow. Ultimately, this work provides a first step for generative modeling in flap kinematics design, laying the foundation for broader adoption of machine learning in overall aircraft design.

Contact address:

antonschreiber99@gmail.com

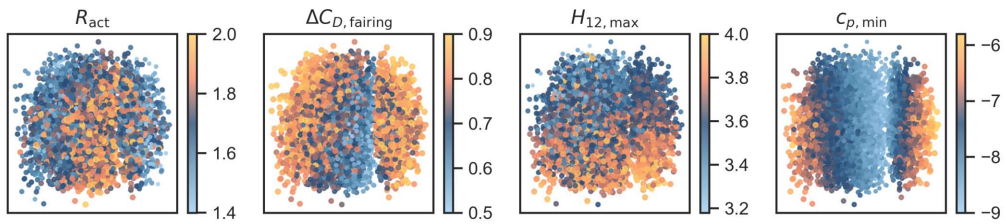


FIG 12. Latent performance maps: Dataset encoded into the two most active latent variables, color-coded by each of the KPIs.

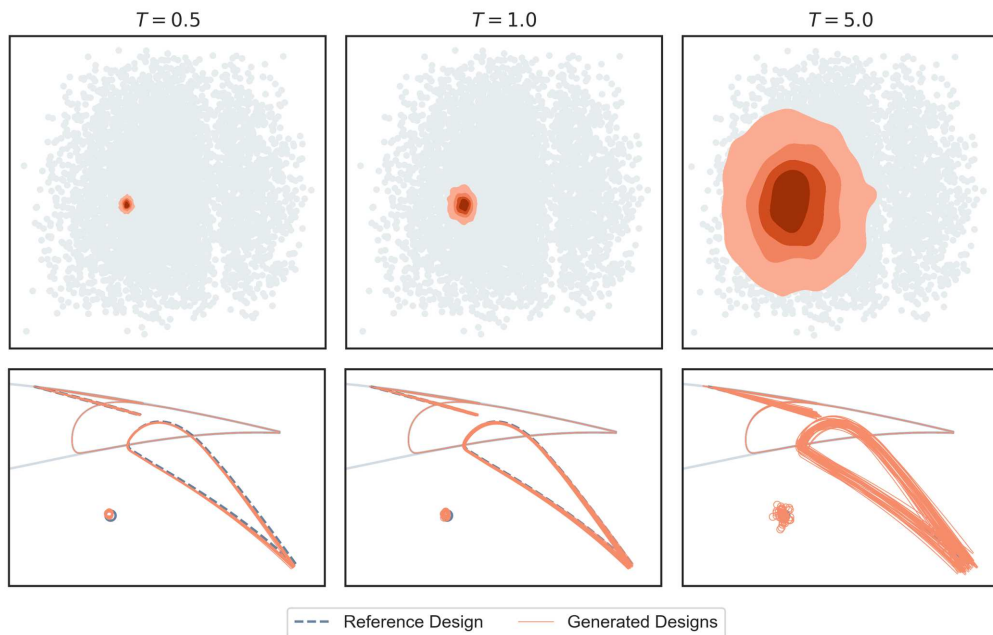


FIG 13. Effect of the model temperature on sampled representations and generated designs.

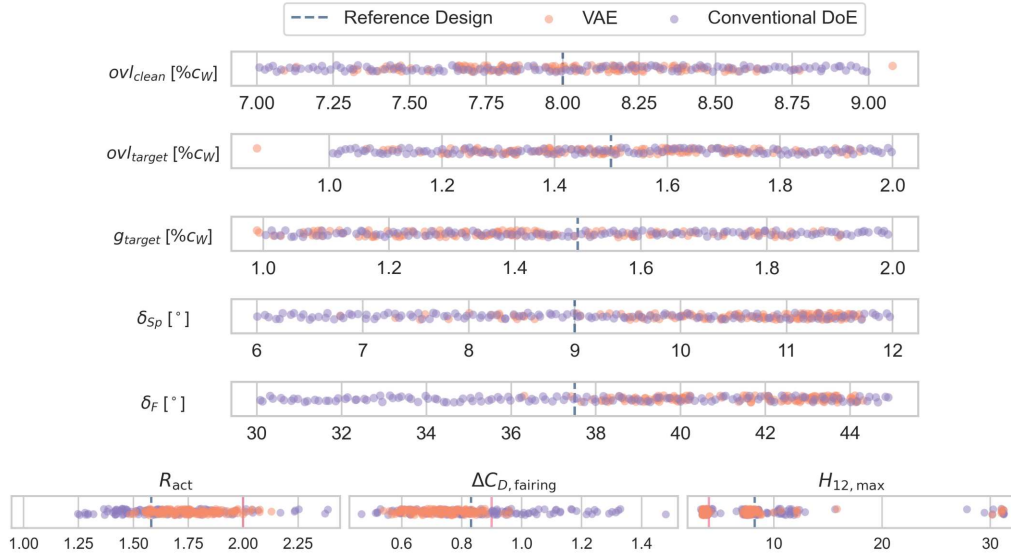


FIG 14. Comparison of the feature and KPI distribution produced with a conventional DoE (Sobol' sampling) and with a VAE for global exploration.

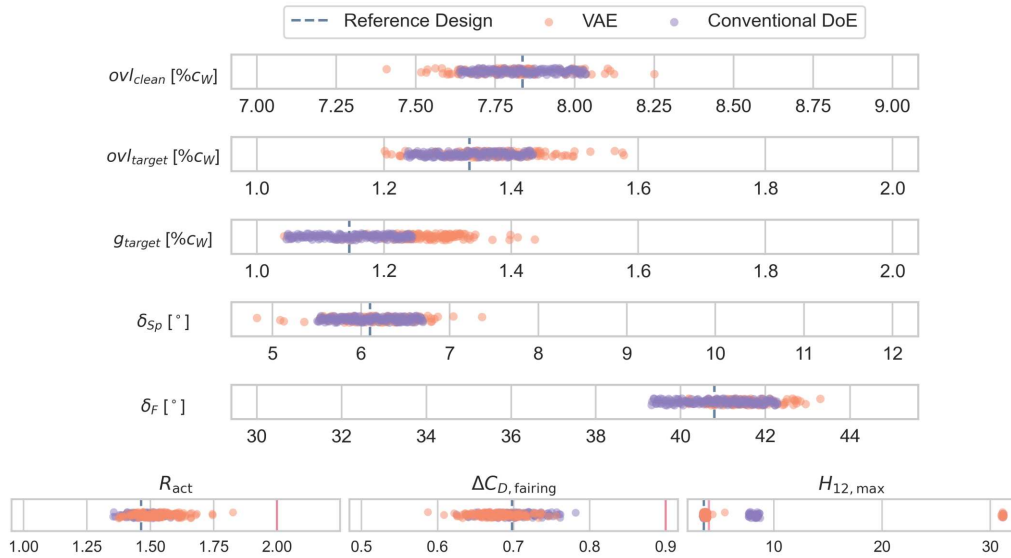


FIG 15. Comparison of the feature and KPI distribution produced with a conventional DoE (Sobol' sampling) and with a VAE for local exploration.

References

- [1] Peter K. C. Rudolph. High-lift systems on commercial subsonic airliners. Technical report, National Aeronautics and Space Administration, 1996.
- [2] Jochen Wild. *High-Lift Aerodynamics*. CRC Press, 2022.
- [3] Dirk Franke. *Multidisciplinary Design of High-Lift Systems*. PhD thesis, 2019.
- [4] Parth Kumar and Brian J. German. Optimization of high-lift systems at takeoff conditions subject to kinematic constraints. *Journal of Aircraft*, 59(4):1098–1103, 2022.
- [5] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes, 2013.
- [6] Andrew J. Lew and Markus J. Buehler. Encoding and exploring latent design space of optimal material structures via a vae-istm model. *Forces in Mechanics*, 5:100054, 2021.
- [7] Sneha Saha, Stefan Menzel, Leandro L. Minku, Xin Yao, Bernhard Sendhoff, and Patricia Wollstadt. Quantifying the generative capabilities of variational autoencoders for 3d car point clouds. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1469–1477, 2020.
- [8] Sneha Saha, Leandro L. Minku, Xin Yao, Bernhard Senhoff, and Stefan Menzel. Exploiting linear interpolation of variational autoencoders for satisfying preferences in evolutionary design optimization. In *2021 IEEE Congress on Evolutionary Computation (CEC)*, pages 1767–1776, 2021.
- [9] Sneha Saha, Thiago Rios, Leandro L. Minku, Bas Vas Stein, Patricia Wollstadt, Xin Yao, Thomas Back, Bernhard Sendhoff, and Stefan Menzel. Exploiting generative models for performance predictions of 3d car designs. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–9, 2021.
- [10] S. Saha, L. L. Minku, X. Yao, B. Sendhoff, and S. Menzel. Exploiting 3d variational autoencoders for interactive vehicle design. *Proceedings of the Design Society*, 2:1747–1756, 2022.
- [11] Xu Wang, Weiqi Qian, Tun Zhao, Hai Chen, Lei He, Haisheng Sun, and Yuan Tian. A generative design method of airfoil based on conditional variational autoencoder. *Engineering Applications of Artificial Intelligence*, 139:109461, 2025.
- [12] Shrinath Deshpande and Anurag Purwar. Computational Creativity Via Assisted Variational Synthesis of Mechanisms Using Deep Generative Models. *Journal of Mechanical Design*, 141(12):121402, 2019.
- [13] Daniel Reckzeh. Multifunctional wing moveables: Design of the a350xwb and the way to future concepts. In *Proceedings of the 29th Congress of the International Council of the Aeronautical Sciences*, 2014.
- [14] I.M Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967. ISSN: 0041-5553.
- [15] J. Angeles and S. Bai. *Kinematics of Mechanical Systems: Fundamentals, Analysis and Synthesis*. Mathematical Engineering. Springer International Publishing, 2022.
- [16] Hanfried Kerle, Burkhard Corves, and Mathias Huesing. *Getriebetechnik*. Springer Vieweg Wiesbaden, 2015.
- [17] European Union Aviation Safety Agency. Certification specifications and acceptable means of compliance for large aeroplanes (cs-25). European Union Aviation Safety Agency, 2023.
- [18] J. Gordon Leishman. *Introduction to Aerospace Flight Vehicles*. Creative Commons, 2024.
- [19] Carl Doersch. Tutorial on variational autoencoders, 2021.
- [20] Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023.
- [21] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 edition, 2006.
- [22] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR (Poster)*. OpenReview.net, 2017.
- [23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
- [24] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.
- [25] James Lucas, George Tucker, Roger Grosse, and Mohammad Norouzi. Don't blame the elbow! a linear vae perspective on posterior collapse, 2019.