

Stochastic Exploration of Transition Pathways for Aviation Using Temporal Convolutional Networks

Ilias Baali*, Burak Bagdatli†, and Dimitri Mavris‡

Aerospace Systems Design Laboratory, Georgia Institute of Technology, Atlanta, Georgia, 30332

There is a growing sense of urgency when it comes to reducing aviation’s environmental impact. Several comprehensive studies have been published in the recent years, setting out goals for the industry and providing recommendations on what actions key stakeholders can take to gradually transition toward a net-zero future. However, a lot of uncertainty remains in the actual impacts that the stakeholders’ decisions will have on the industry. In order to analyze how this uncertainty translates to uncertainty on CO₂ emissions, this study sets out to develop a methodology enabling a probabilistic evaluation of specific decision scenarios. First, a framework to construct a surrogate model linking a decision scenario timeline to its corresponding evolution of CO₂ emissions over time is introduced. Then, this model is used to analyze the sensitivity of a few scenarios to variations in their respective decisions’ impacts.

I. Nomenclature

β	=	Regression coefficients
ε	=	Regression error term
I	=	Information matrix
I_k	=	Impact of decision k
K	=	Number of impact variables per case
M	=	Number of curves generated
N	=	Size of the final DoE
$\mathcal{N}(\mu, \sigma^2)$	=	Gaussian distribution with mean μ and variance σ^2
T	=	Number of time steps
X	=	Design matrix
Y	=	Calculated response

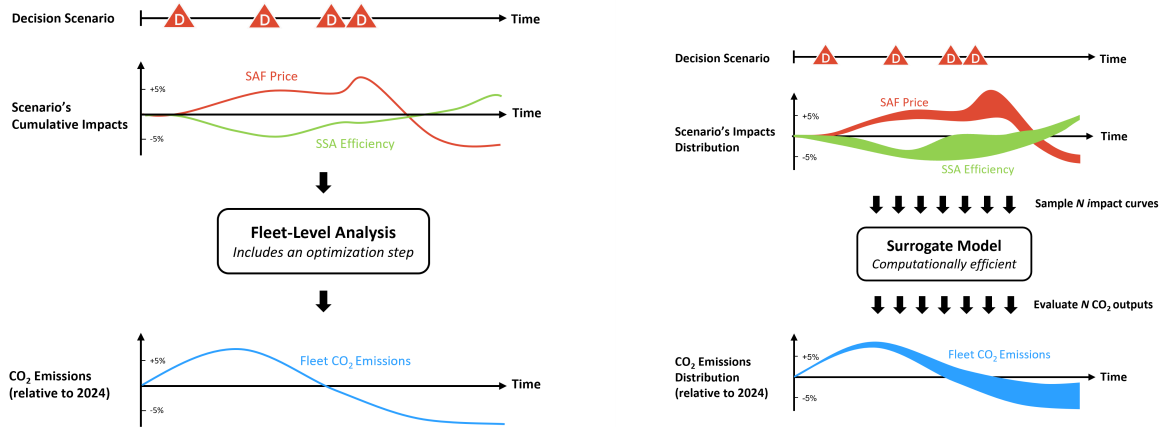
II. Introduction

If aviation is one of the major contributors to the United-States economy, it is also deemed responsible for about 2.5% of global greenhouse gas emissions. Despite significant improvements on aircraft efficiencies, the rapidly growing air traffic exceeds these improvements and without a major shift of the industry, aviation’s CO₂ emissions are expected to almost triple by 2050 [1]. In an effort to reduce aviation’s impact on the environment, comprehensive studies aiming at analyzing scenarios describing the potential evolution of the aviation sector have recently been conducted. These studies include ICAO’s Long-Term Aviation Goals [1], IATA’s Net-Zero Roadmap [2], and ATAG’s Waypoint 2050 [3], which all have a common goal of better understanding the impacts of future technologies and shifts in airline operations, to progressively guide the aviation sector towards a net-zero emissions future. However, a common limitation of these studies is the very limited number of scenarios that they provide, often restricting their scope to a handful of scenarios due to time constraints and a limited availability of subject-matter experts. Moreover, these studies often overlook the intricate decision-making process of key industry stakeholders, a determining factor in turning the proposed scenarios into a reality. With these limitations in mind, the *Scenario Generation for Strategic Decision-Making towards Net-Zero Aviation* [4] and *Scenario Modeling and Analysis for Strategic Decision-Making towards Net-Zero Aviation* [5] papers

*Graduate Research Associate, Aerospace Systems Design Laboratory, School of Aerospace Engineering, AIAA Student Member

†Research Engineer II, Aerospace Systems Design Laboratory, School of Aerospace Engineering, AIAA Member

‡S.P. Langley Distinguished Regents Professor, Aerospace Systems Design Laboratory, School of Aerospace Engineering, AIAA Fellow



(a) **Single scenario evaluation:** as decisions are made over time, they affect the value of different impact variables (e.g., SAF prices). The evolution of these impact variables serves as an input to the fleet-level analysis, which outputs the corresponding evolution of CO₂ emissions.

(b) **Probabilistic Scenario Exploration:** the fleet-level analysis is replaced by a surrogate model that quickly evaluates impact inputs sampled from the scenario's impact distribution, yielding a distribution of possible CO₂ emissions for that decision scenario.

Fig. 1 Comparison of the single scenario evaluation and probabilistic scenario exploration

provide a new methodology that enables the quick generation of realistic decision-based scenarios, as well as the analysis of the impacts that these decision scenarios would have on the sector.

The first paper focuses on generating realistic scenarios composed of some decisions available to the industry's stakeholders. The paper starts by identifying six key stakeholder categories of the aviation industry, namely, the policymakers, the technology developers (such as NASA), the airports, the energy producers, the OEMs, and the airlines. These stakeholders are the most likely to shape the future of aviation as they make specific operational decisions throughout the years. In an effort to encompass most of the scenarios defined in prior studies, the authors of the paper laid out a list of decisions available to each stakeholder, mainly extracted from the transition paths suggested in studies like ICAO's Long-Term Aviation Goal. To each of these decisions, impact metrics were associated to model the effect of making a decision. For example, the energy producers' decision of increasing power-to-liquid production of Sustainable Aviation Fuels (SAF) was associated to an increase in average SAF price, water and land use, as well as hydrogen demand. In order to account for uncertainty as well as time fluctuations of the impacts of a decision, a range of impact values is given to each decision, and the actual value used to evaluate a scenario varies randomly depending on the year the decision is made. These impact values are also used to evaluate the value of making a decision, to mimic the decision-making process of aviation stakeholders using a real-options analysis. For more information on the exact methodology followed to generate realistic scenarios, the reader is invited to refer to the original paper [4].

Once a decision scenario is generated, the impacts of its decisions are added together to build a cumulative impact time series. Therefore, each scenario is associated to the time evolution of 18 impact variables resulting from the decisions it contains. In order to analyze how that translates to an evolution of the fleet-level CO₂ emissions, the second paper *Scenario Modeling and Analysis for Strategic Decision-Making towards Net-Zero Aviation* [5] builds up on previous work [6] to provide a fleet-level analysis framework that takes the scenario's cumulative impacts time series as inputs, and outputs the time evolution of key metrics for the United States aviation fleet, such as fuel demand (including alternative fuels), evolution of the airline ticket price, composition of the fleet, and CO₂ emissions. In this framework, the impact inputs are used to rescale forecast data from the FAA [7] and other agencies, prior to an optimization step that matches the air traffic supply with the demand. The result of this optimization is then combined with estimated future aircraft efficiencies, potentially affected by the decisions' impacts, to calculate output metrics like CO₂ emissions. A general overview of this evaluation process is depicted in Fig. 1a. The exact assumptions used for this modeling are specified by Baali et al. and Hassan, Pfaender, and Mavris [5, 6].

This scenario generation and analysis framework was used to generate and analyze 20,000 decision scenarios, and a visualization environment was developed to display the results of the simulations. Due to the complexity of the

optimization step in the fleet-level analysis, the different cases needed to be pre-computed before being visualized, which limits the number of scenarios that can be analyzed as well as the scalability of the approach, since adding new available decisions would necessarily require running all the simulations again. That visualization environment also provides the ability to manually build decision scenarios and evaluate them at the fleet-level, enabling a better understanding of how specific decisions would impact the future of the industry. Ideally, this environment would be used as a tabletop exercise, with the actual stakeholders making decisions together to identify realistic and satisfying transition paths towards a net-zero aviation future. However, due to the stochastic nature of the impact values associated to each decision, a single-point evaluation is not necessarily representative of the goodness of a scenario. Indeed, a good-performing scenario might have performed well during the analysis not because it is intrinsically good but because, by chance, all its decisions had their impact at their optimal values for that specific simulation. Therefore, to enable reliable and informed decisions that are robust to uncertainty in the future value of each decision's impact, a more probabilistic approach to the scenario evaluation is needed. The rest of this paper focuses on developing this stochastic evaluation process.

III. Proposed Approach

As discussed in the previous section, robust decision-making towards a net-zero future can only be achieved through a probabilistic analysis of potential transition scenarios. A similar problem was encountered by Kirby in her dissertation [8], in which she developed a methodology to evaluate and select future aircraft technologies based on uncertain "k-factors", that represent the impact of a future technology on aircraft parameters. In order to evaluate the uncertainty in the final design performance from the uncertainty in individual k-factors, Kirby makes use of a Monte-Carlo approach coupled with a surrogate model of the simulation software used to analyze a design's performance. A similar process can be followed to stochastically evaluate transition scenarios. For a specific timeline of decisions, the uncertainty in each decision's impact results in a distribution of potential impact time series for that scenario. The general idea is that repeating the fleet-level analysis step shown in Fig. 1a with different inputs sampled from the impact distribution would, in turn, generate a distribution of CO₂ emissions over time. Analyzing this distribution will provide a deeper insight into the scenario's robustness to variations in its decisions' impacts. However, obtaining accurate output distributions using Monte-Carlo simulations often requires analyzing thousands of input samples, a task that would be computationally expensive using the fleet-level analysis as described in [5]. To address this challenge, a surrogate model can be created to perform the evaluation step and quickly estimate the CO₂ outputs for different impact inputs. An overview of this approach is depicted in Fig. 1b. Building a surrogate model usually involves 3 steps given below.

- 1. Design of Experiments (DoE)** The first step of the process is to acquire data that will be used to build the surrogate model. Instead of generating random input combinations, collecting the corresponding outputs, and using that data to create a surrogate model, experimental designs [9] provide efficient methods to design a set of experiments that will maximize the information that can be extracted from the results of the simulations, often yielding better prediction accuracy when used for predictive modeling. For a fixed number of inputs and experiments, DoE methods aim at maximizing the coverage of the input space, while minimizing the correlation between the experiments to avoid redundancy. For this work, a case in the DoE specifies the evolution of 18 impact time series over 35 years, corresponding to the cumulative impacts of a scenario's decisions.
- 2. Simulations** Once the cases defined, simulations are run for each experiment in the DoE to calculate the output(s) of interest. Here, the simulations are run using the modified simulation environment from [5], and the evolution of CO₂ emissions is recorded for each case.
- 3. Surrogate Modeling** Finally, a surrogate function is created based on the data accumulated in the simulations. Multiple models exist to create this function, ranging from the simple linear regression model to artificial neural networks (ANN), more suited to non-linear applications. Multiple methods will be analyzed in the next sections to create the surrogate model of the fleet-level analysis step.

For this work, the difficulty stems from the time-series nature of inputs and outputs in the design of experiments and surrogate modeling processes. Thus, the next sections will focus on adapting the DoE process to time series, before providing a review of a few time-series modeling approaches and evaluating their performance to replace the fleet-level scenario evaluation step. Finally, the selected surrogate model will be coupled to a Monte-Carlo simulation to analyze the distribution of potential CO₂ emissions over time resulting from a specific decision scenario.

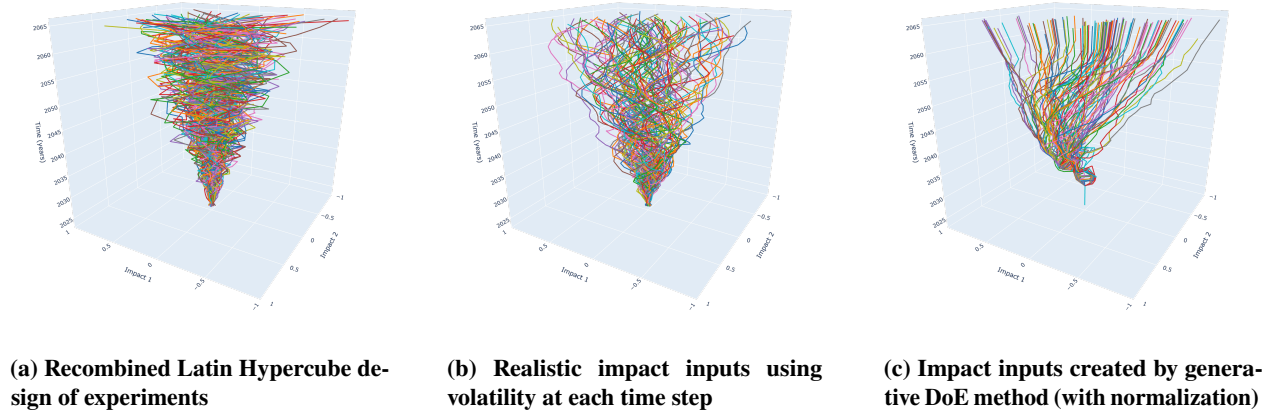


Fig. 2 Different approaches to create decision impact inputs, projected in two impact dimensions

IV. Design of Experiments

A. Input generation

Unlike with scalar data for which it is straightforward to sample an input point by randomly choosing the values on each of its dimensions, there is no immediate way to sample a whole time series. In the next paragraphs, we identify a few options to the time series generation problem.

Baseline approach As previously discussed, a design of experiments is created with the objective of maximizing the information that can be extracted from the experiments. Traditional DoE methods are designed with the assumption that inputs are scalar, independent and uncorrelated variables. The inputs to the fleet-level analysis that needs to be surrogated, however, are time series, which makes the problem more intricate.

A first approach could be to bypass this difficulty by considering the value of each impact variable at each time step as a separate input variable, and applying a traditional DoE method to generate the values of these variables. An impact time series can then be rebuilt by concatenating the different time steps for each impact variable to run the simulations. This first approach was implemented as a baseline using a Latin Hypercube design of experiments, which creates N cases by splitting the range of each variable into N intervals, sampling a random uniform value in each of these intervals, and combining the sampled values for each variables to create N combinations of inputs. This is usually an efficient way to create a design of experiments as it ensures that all the input space is being covered. A visualization of the impact time series generated with this approach, projected in two impact dimensions, is shown in Fig. 2a.

Even though this approach is relatively easy to implement and provides a good coverage of the impact space, as one can see, the generated input curves present sharp variations from year to year, which are unrealistic in the context of decision impacts. For instance, the production of Sustainable Aviation Fuel has no reason to jump from +40% to -20% in a single year, before going up again the following year. This problem comes from the assumption that the impact values are independent from one year to the other, an assumption that was implicitly made when it was decided to consider different years as separate variables.

Volatility approach Intuitively, the value of an impact I_k at year $t + 1$ is correlated to its previous value. In a realistic time series with a relatively low volatility, it is not likely to observe high variations in impact values, and one can expect the value of $I_k(t + 1)$ to be close to $I_k(t)$. Therefore, an approach to generate more realistic impact curves could involve starting with an initial distribution of values for the first year $I_k(0)$, and iterating over the time steps to generate the following value using a distribution centered around the previous value: $I_k(t + 1) \sim \mathcal{N}(I_k(t), \sigma^2)$, where the standard deviation is chosen to regulate year-to-year variations. To keep the impacts within the minimum and maximum values, these probability density functions are truncated and rescaled to only allow values in $[I_{k_min}, I_{k_max}]$. By repeating this process for each impact to generate N input samples, another design of experiment was created, shown in Fig. 2b. As one can see on the projected plot, the impact time series generated with this approach are smoother and more realistic than those obtained with the Latin Hypercube method. However, this approach does not necessarily model the gradients

of the input curves accurately. Indeed, year-to-year variations are modeled through a fixed standard deviation parameter which does not allow modeling inputs with fast variations in the initial years and slow variations later in time, or inputs that only increase or decrease, for example.

Generative models Generative models are deep learning models trained to generate new data similar to the data on which they have been trained. A simple example would be a model trained on a dataset of dog pictures, which would learn to generate new artificial pictures of dogs. For our specific problem, a generative model would be trained on a small dataset of true input curves and would learn to generate new sample inputs, similar to the actual data. State of the art generative models include the following methods.

Generative Adversarial Networks (GAN) [10] concurrently train a generator to generate realistic samples, and a discriminator, to recognize real and fake samples. Once the generator is able to consistently fool the discriminator, it can be used to generate new realistic samples.

Variational Autoencoders (VAE) [11] train an encoder to encode the real samples from the training dataset into a normal distribution in a latent space, as well as a decoder that reconstructs a sample from its latent space prior. New samples can be generated by sampling new priors from the latent space distribution and using the decoder to obtain new data.

Denoising Diffusion Probabilistic Models [12] iteratively add gaussian noise to training samples until the pixel values seem normally distributed, and trains a neural network to predict the noise added at each step. New samples can be generated by iteratively denoising the random image using the trained model.

If these generative models are usually used with image data, previous work [13] has shown that variational autoencoders can successfully be applied to the generation of realistic multivariate time-series data. Thus, we trained a TimeVAE model as described in [13] to generate realistic input curves. This model was used to generate the sample shown in Fig. 2c. The advantage of this approach is that the model generates samples that accurately mimic the variations and range of actual data, but also the correlations between the various dimensions of the inputs (e.g., the fuel prices and SAF prices). Moreover, by learning from actual data, the VAE model is able to accurately identify correlations between the different dimensions of the multivariate input time series, which allows the DoE to closely focus on the part of the input space that is of actual interest. The strength here lies in the fact that there is no need for the user to specify explicit linear and/or non-linear constraints between input dimensions, as the deep learning model implicitly learns these constraints from the training data. However, this approach can only be used when a varied sample of input data is already available to train the generative model. Depending on the application, data representative of the whole input space might not always be available at the time the DoE is created.

B. Increasing the covered space

If the inputs generated with the previous approaches are somewhat realistic, there is no guarantee that they form an optimal set of cases when trying to fit a surrogate model. To increase the expected coverage of the impact space, a solution would be to increase the number of random realistic impact curves generated with the previous approaches. However, doing so would also increase the number of simulation to run, which defeats the purpose of using a design of experiments. In order to keep the size of the DoE to the same value while improving the coverage of the input space, we decided to generate a higher number of realistic curves $N' \gg N$, and to sample N trajectories among these curves in a way that maximizes the coverage of the impact space. For that purpose, we identified Farthest Point Sampling (FPS) [14] to be an appropriate sampling method. Originally, the purpose of this greedy algorithm is to downsample from point clouds while maximizing the distance between each sampled point and their closest neighbor. FPS is commonly used in 3-D object detection and point cloud processing applications, and is one of the most popular sampling methods today [15]. By maximizing the distances between the points in the sample, the algorithm ensures that the coverage of the original point cloud is maximum. Therefore, applying the Farthest Point Sampling method to a set of realistic impact curves that contains enough trajectories to cover most of the impact space should reduce the number of cases, minimize the redundancy, while maintaining a good coverage of the impact space.

In order to apply Farthest Point Sampling to time series, we need a distance metric measuring the similarity between impact curves. In traditional implementations on simple point clouds, the Euclidean distance is used. However, when comparing time series, this distance would only measure year-to-year similarity between curves, which means that two identical curves shifted in time could result in a high distance even though they are very similar. To alleviate this problem, we replaced the Euclidean distance by Dynamic Time Warping (DTW) [16], a distance metric specifically

Algorithm 1 DoE Generation

```
1: procedure MAKE_DOE( $N, N', initial\_data, method$ )
2:   if method == 'generative' then
3:     Train a TimeVAE model [13] on  $initial\_data$ 
4:     Use the trained VAE model to generate  $N'$  realistic inputs:  $gen\_curves$ 
5:   else if method == 'volatility' then
6:     Iteratively generate  $gen\_curves$  with the volatility approach
7:   Apply the sampling method to keep  $N$  generated inputs:  $DOE = FPS(gen\_curves, N)$ 
8:   return  $DOE$ 
9:
10: procedure FPS( $curves, N$ )
11:   Randomly pick a first curve in  $curves$  to add to  $sample$ 
12:   for  $1 \leq k \leq N$  do
13:     Calculate DTW distance between remaining curves and  $sample$ 
14:     Add the farthest curve to  $sample$ 
15:   return  $sample$ 
```

designed for time series and widely adopted in machine learning applications. DTW measures the similarity between time series by aligning them in the time dimension, before calculating the Euclidean distance between the aligned series. As a result, two impact curves with similar variations but elongated or shifted in time will be considered close to each other by the FPS algorithm, and only one of them will be kept in the final sample. This is the desired behavior of our DoE method, as such inputs would not provide a lot of information compared to two input curves with very different variations. The proposed algorithm to generate the design of experiments is summarized in Algorithm 1.

C. Evaluation metrics

The performance of a design of experiments is measured by optimality metrics. D-Optimality and Orthogonality are commonly used metrics for this purpose. D-optimality measures the amount of information contained in the design of experiments and Orthogonality ensures that the impact of each input parameter can be estimated independently, minimizing confounding effects between variables.

These metrics are defined using the Fisher information matrix. Strictly speaking, the definition of this matrix should be associated to a specific regression model, but design of experiments are commonly evaluated using the information matrix of a linear regression model. The common form of a linear regression model is given in Eq. 1, where X is the design matrix, Y is the calculated response, and $\varepsilon \sim \mathcal{N}(0, 1)$ is the error term. For this particular model, the information matrix is defined as I given in Eq. 2.

$$Y = X\beta + \varepsilon \quad (1)$$

$$I(\beta) = X^T X \quad (2)$$

Unfortunately, the linear model can no longer be applied in the context of time-series design of experiments, as it does not account for the temporal dimension of the inputs and outputs. However, by defining a new reference model, similar to the linear model but adapted to time-series inputs and outputs, a modified Fisher information matrix can be derived. Eq. 3 defines a model similar to linear regression but accounting for the time dimension (with finite time horizon T), where $X(t_k) \in \mathbb{R}^{N \times K}$ is the design matrix containing the input values at time t_k , $Y(t_k) \in \mathbb{R}^N$ is the corresponding response, and $\beta \in \mathbb{R}^K$ contains the regression coefficient.

$$Y(t) = \begin{cases} X(t_1)\beta + \varepsilon(t_1) & \text{for } t = t_1 \\ X(t_2)\beta + \varepsilon(t_2) & \text{for } t = t_2 \\ \dots & \\ X(t_T)\beta + \varepsilon(t_T) & \text{for } t = t_T \end{cases} \quad (3)$$

This model is essentially a linear model applied to each time step, with a single set of regression coefficients β . It can be rewritten using tensor notations to obtain the Eq. 4 with $\tilde{X} \in \mathbb{R}^{N \times T \times K}$, $\tilde{Y} \in \mathbb{R}^{N \times T}$, $\beta \in \mathbb{R}^K$, and $\tilde{\varepsilon} \in \mathbb{R}^{N \times T}$.

Table 1 Comparison of DoE Methods

Metric	Latin Hypercube	Volatility Approach		VAE Approach	
	Baseline	No FPS	FPS	No FPS	FPS
D-Optimality (log ₁₀ value)	52.61	57.57	58.62	46.02	47.80
Condition number	1.23	1.37	1.29	3736.63	1255.35
Average DTW distance to closest point	6.61	8.51	9.49	3.5	4.15

Here, \tilde{X} corresponds to the 3-dimensional design tensor which contains the values of the K impact time series at each time step $1 \leq t \leq T$, for each one of the N experiments in the DoE.

$$\tilde{Y} = \tilde{X}\beta + \tilde{\varepsilon} \tag{4}$$

Under a few assumptions, the corresponding Fisher information matrix in Eq. 5 can be derived with $\tilde{X}^{(n)} \in \mathbb{R}^{T \times K}$ the input data for case n of the DoE. In the specific case where $T = 1$, Equation 5 yields the same result as Equation 2. The derivation of Equation 5 is provided in the Appendix.

$$\mathcal{I}(\beta) = \sum_{n=1}^N \tilde{X}^{(n)\top} \tilde{X}^{(n)} \tag{5}$$

D. Comparison of DoE methods

Once the information defined for time-series designs of experiments, this result can be used to calculate the different criteria and compare the goodness of different designs. For this specific study, it was decided to compare DoEs based on their D-optimality metric, orthogonality, and the uniformity of the input distribution in the impact space. To provide a fair comparison between the different approaches, the range of value for each input dimension was normalized such that the minimum and maximum impacts linearly increase from 0 in 2024 to ± 1 in 2065, as displayed in Fig. 1. The results are summarized in Table 1.

As mentioned before, the D-optimality measures the amount of information contained in the design. It is defined as the determinant of the information matrix and needs to be maximized to get an optimal design. As Table 1 shows, the volatility-based curve generation coupled with the FPS method has the best D-Optimality. The VAE approach lags behind the other methods, which is not surprising considering that the inputs it generates are less varied but more focused on specific areas of interest within the input space. In both cases, the addition of Farthest Point Sampling improved the D-optimality of the DoE. The condition number measures the orthogonality of the design, with a value of 1 corresponding to a perfectly orthogonal design. Here, the Latin Hypercube method performs slightly better than the volatility-based approach, but both remain quite close to reaching orthogonality, especially when coupled with FPS. However, the high value of the condition number for the VAE method suggests that the generated design lacks in orthogonality. This observation can probably be explained by the specific nature of our data, as most inputs have a similar behavior with the main differences lying in the magnitude of the impact. Therefore, it is likely that the generated DoE also contains some curves close to be linearly dependent. We can see that for that approach especially, the FPS algorithm with the DTW greatly improves the orthogonality of the design by efficiently removing cases that are too similar and might be correlated. Finally, the distribution of the similarity distance between each input curve and the closest one in the DoE is shown on Fig. 3, with the average values reported in Table 1. The volatility approach outperforms the baseline method with higher DTW distance between the DoE samples, which suggests more varied inputs and a better coverage of the impact space, as the generated cases are more spread out. We can see that adding the FPS also improves the diversity of input cases as the distances between samples increase overall. Moreover, the narrow distribution of distances to the closest point suggests a more uniform coverage of the input space with FPS, similar to what can be observed with scalar DoE methods like space-filling designs. The DoE that uses the VAE model for curve

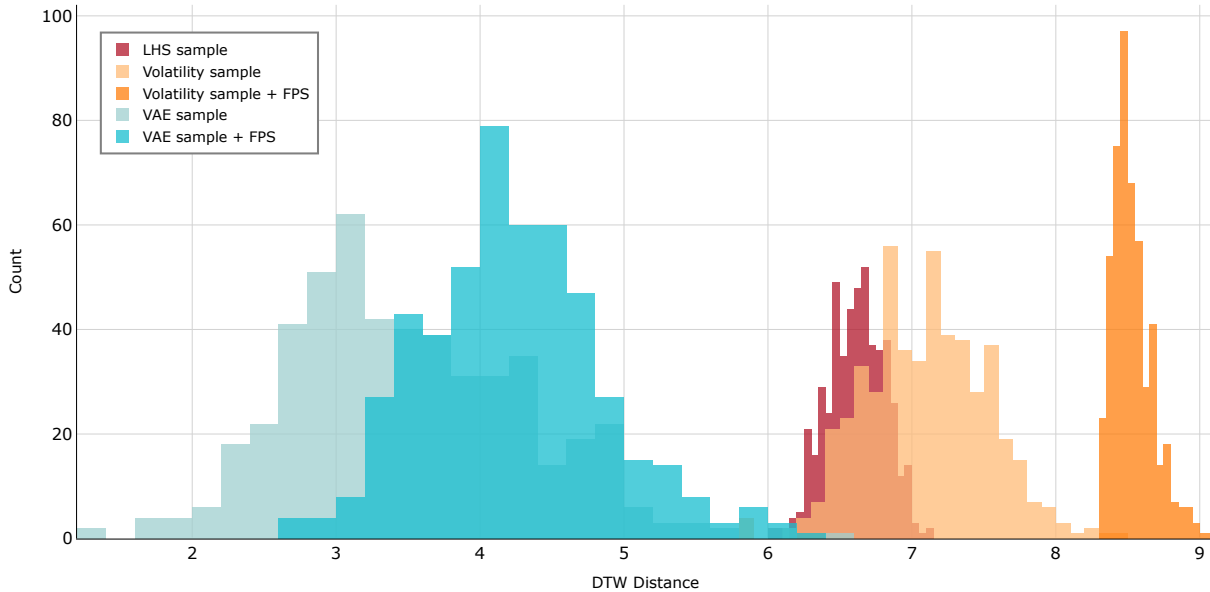


Fig. 3 Distribution of DTW distances to the closest impact curve in the sample.

generation has, in average, lower distances between the input curves, meaning that the inputs are more similar to each other. However, this is not surprising as this approach specifically targets areas of the input space around which the future surrogate model will be applied. Since the same sample size is used to cover a smaller volume, it is expected that the distances between samples are reduced. This will result in better precision of the surrogate model in the area due to less interpolation compared to the volatility approach.

Ultimately, the choice of the appropriate approach will depend on the specific application. When enough realistic data is available to train a generative model capable of creating samples varied enough to cover the region of interest in the impact space, we found that this approach results in surrogate models that generalize a lot better to new data compared to the volatility-based DoEs. However, the latter could still be useful in situations where the knowledge on the input distribution is limited, as it enables the simple generation of realistic time series without any prior data. We could also imagine combining the different methods to improve the variety of cases while keeping a good resolution on the expected input region, for example. In any case, we found that the inclusion of Farthest Point Sampling with Dynamic Time Warping after curve generation greatly improves the quality of time-series DoEs.

For the remainder of this work, we used the VAE approach combined with FPS, as it yielded good results in the surrogate modeling process.

V. Surrogate Modeling

A. Approach

After running the simulations defined by the design of experiments, the next step is to create a surrogate model of the fleet-level analysis simulation environment. This model will predict future CO₂ emissions resulting from specific impact inputs. These inputs consist of 18 continuous variables, as well as two discrete variables corresponding to the first year airlines decide to incorporate hydrogen and electric aircraft into their fleet, respectively. To simplify the modeling task, these two years are transformed into continuous variables using one-hot encoding. They are represented using two time series equal to 0 until their corresponding decision is made, at which point their value changes to 1. Moreover, all the input time series are normalized to be centered around 0.

Iterative ANN Once again, the time-series nature of the inputs and outputs makes the creation of a surrogate model a more complex task, as traditional methods tailored to scalar data, such as Artificial Neural Networks, would consider the value of each impact at each time step to be independent variable. This would result in an increased number of

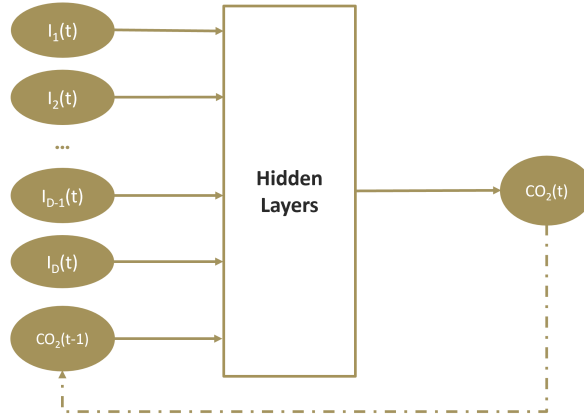


Fig. 4 Iterative ANN approach

inputs which would necessitate bigger models and would require a lot more training data than available. Moreover, this approach would not account for causality, as the CO_2 output for a specific year would also depend on the future impact values. To reduce the number of inputs and recover the temporal causality while leveraging ANN, a different approach consists in predicting the CO_2 output of a year t using only the impact values of a fixed number of prior years, plus the corresponding previous (predicted) CO_2 outputs. An overview of this approach is provided in Fig. 4. However, we observed that despite good single-year prediction capabilities, small errors tend to propagate when applying this approach iteratively to predict the whole time series of CO_2 emissions, resulting in unreasonably high residual errors.

Recurrent Neural Networks Therefore, it is necessary to resort to other approaches more suited to sequential data, such as Recurrent Neural Networks (RNN) [17]. RNNs follow a similar structure as the one depicted in Fig. 4, but they are trained on the whole time-series inputs to predict the time-series outputs. They have been widely used for applications ranging from time-series prediction, to sentiment analysis and Neural Language Processing. However, a common issue with RNNs is that they suffer from vanishing gradient, which makes it difficult for them to “remember” data longer than for a few time steps. To solve this problem, variants like Long Short-Term Memory (LSTM) [18] and Gated Recurrent Unit (GRU) [19] have been introduced and modify the RNN cell architecture to selectively remember or forget information, improving the prediction capability on longer sequences. When training a LSTM and a GRU model to predict the CO_2 time series, we obtained relatively good results over the first 20-25 time steps, but on the later years, the propagated error started to diverge quickly, and remains too high to get any benefits from a sensitivity analysis.

Temporal Convolutional Networks A more recent approach to process sequential data is to adapt Convolutional Neural Network, traditionally used on 2-D matrices and images, to the 1-dimensional problem of time-series forecasting. This specific problem has been extensively described in [20]. In this paper, the authors introduce a new network named Temporal Convolutional Networks (TCN), which consists of multiple 1-dimensional convolutional layers that respect causality. They introduce a dilation process to reduce the number of layers needed to ensure that all the inputs are used to predict the future outputs of the model (*full history coverage*), while keeping the size of the kernel relatively low to allow for finer variations. The authors show that their TCN model can outperform LSTM and GRU on similar predicting tasks. Thus, this model was trained using the DoE data. In order to keep the relative error relatively constant over the predicted time series, we used a *mean absolute percentage error* (MAPE) loss function instead of the more traditional mean squared error (MSE) to train the model. Indeed, the MSE loss tends to train a model that predicts the CO_2 emissions with a relatively constant error over all time steps but since the general trend in the data features decreasing emissions over time, this behavior results in higher relative errors in the later years, which is not desirable. Overall, we obtained better fit of the whole CO_2 time series with the MAPE loss, as defined in Eq.6.

$$MAPE = \frac{1}{N} \sum_{i=1}^N \sum_{t=1}^T \left| \frac{\hat{y}_i(t) - y_i(t)}{y_i(t)} \right| \quad (6)$$

Ensembling A major challenge resulting from the time-series nature of the inputs is preventing extrapolation when using the trained model. In traditional surrogate modeling approaches, we usually ensure that the model is not extrapolating by specifying bounds on each dimension and ensuring that the DoE covers all the range of values within these bounds. Then, the model is only used within these intervals (*in-distribution data*). However, in our case, the infinite dimensionality of the input space blurs the line between *in-distribution* (ID) and *out-of-distribution* (OOD) data, as time series with a similar range of values could still be OOD due to differences in additional features, such as high gradients, for example. This issue must be accounted for when using a time-series surrogate model on new input data, as deep learning models do not usually extrapolate well. OOD detection for time series is an active field of research, but a simple approach consists in training an ensemble of surrogate models and comparing their prediction at inference time. The general idea is that on in-distribution data, the models of the ensemble are expected to predict similar values if trained correctly, while the extrapolation on out-of-distribution data would result in sparser predictions of the ensemble output. Moreover, averaging the predictions over the ensemble for ID data can also improve the accuracy of the predictions. Thus, we created an ensemble of 10 TCN models, using bootstrapping to slightly alter the DoE (training) dataset [21] as well as random weight initialization. The performance of the final ensemble is compared to the performance of a single model in the next paragraph.

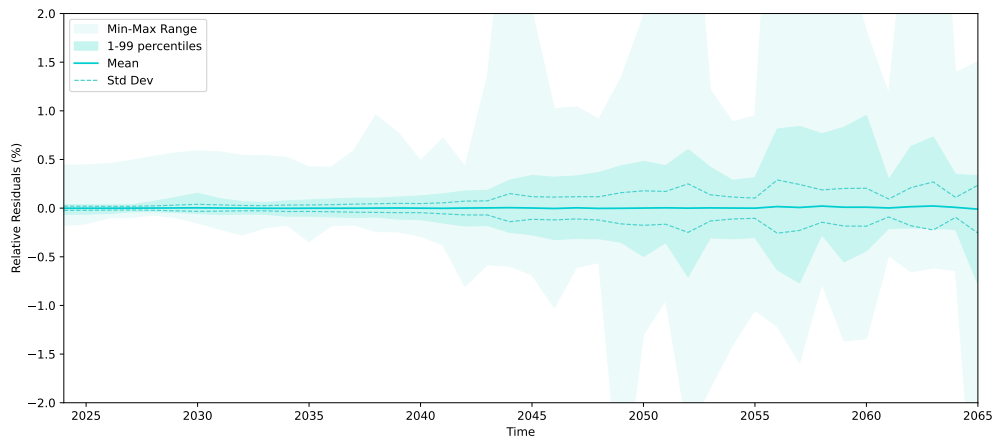
B. Prediction performance

To compare the performance of the single TCN model and the ensemble of 10 models, both were trained using the same data from the DoE dataset. 18,000 cases were used as the training set and 2,000 cases were retained for validation. In addition to the DoE data, a test dataset of actual scenario impact curves, as generated in [4], was used to assess and compare the generalization performance of the single model and ensemble.

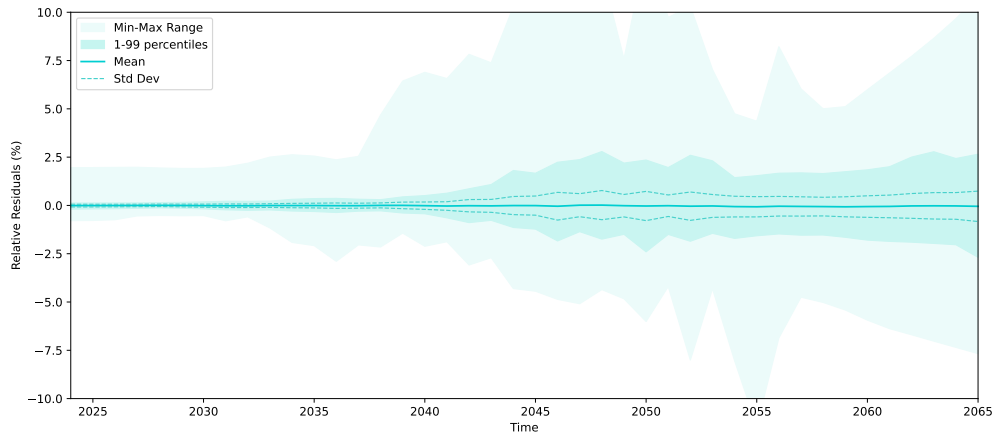
Single model performance Fig. 5a and Fig. 5b show the distribution of the relative residuals for each year of prediction with the trained single model on the validation and test datasets. For the validation data (Fig. 5a), we can see that the model accurately predicts the CO₂ emissions over the whole time horizon, with relative errors under 1% each year. The model is especially good before 2040, which is not surprising as the true CO₂ values to predict do not vary significantly from case to case. Most of the work of the surrogate modeling task is to predict the transition phase from 2040 to 2065, which is done accurately for the validation data. When we apply the trained model to the test dataset (Fig. 5b), we notice that the relative error increases significantly, even though the predictions are still acceptable with errors kept under 3%. We observe that in the transition years, the standard deviation of the relative residuals increases from 0.25% for the validation data to close to 1% for the test data. The analysis of the test cases leading to higher residuals showed that the impact curves fed to the trained model to predict the CO₂ emissions presented features that did not appear within the training and validation data generated with the VAE model, as described in Section IV. For example, some cases exhibited a drop in hydrogen operating costs slightly earlier than all the cases in the training data, making them edge cases for which the model is forced to slightly extrapolate. These edge cases do not exist in the validation data by construction, as this data was generated from the same VAE latent space distribution as the training data, making it *in-distribution* by nature. The distributions of relative residuals shown in Fig. 5b provide a good insight into how well the surrogate model can predict the CO₂ emissions for a specific year. However, this results in the evaluation of the model on a different, more complex task than the one for which it is intended. Indeed, the objective of our surrogate model is to predict the *trajectory* of CO₂ emissions resulting from a set of time-series inputs, and the goodness of fit should therefore be evaluated using case-specific metrics that measures how well the whole CO₂ time series is predicted. Thus, we calculated the average relative residuals over time for each case in the datasets to evaluate the overall performance of each prediction. Additionally, we also considered the relative error on each CO₂ output’s integral, as this is an important metric for transition scenarios due to long-lived and cumulated climate effects of CO₂ in the atmosphere. These two trajectory-specific metrics are shown in Fig. 6 and 7, respectively.

Overall, we observe that the model accurately predicts the whole CO₂ trajectories with an average error consistently under 1%, even on the test data. The prediction of the integral emission is also greatly accurate, similar to what we could expect of a more traditional scalar surrogate model.

Ensemble performance As previously mentioned, a simple way of detecting the edge cases or *out-of-distribution* cases during inference is to use an ensemble of models. Thus, we trained an ensemble of 10 models, each similar to the one analyzed in the previous paragraph. During inference, the median of the 10 model predictions is kept as the ensemble output, each year. Using the median instead of the mean prediction helps mitigate the impact of a single

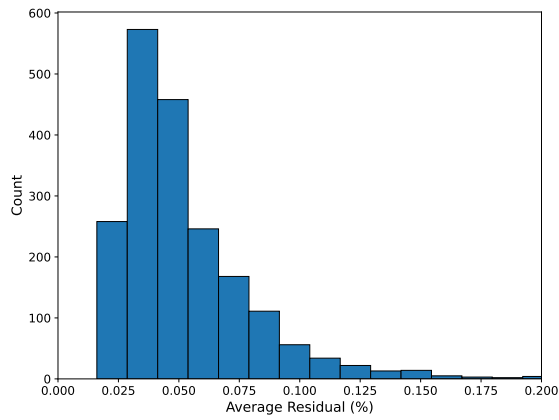


(a) Relative residuals for the validation dataset (%)

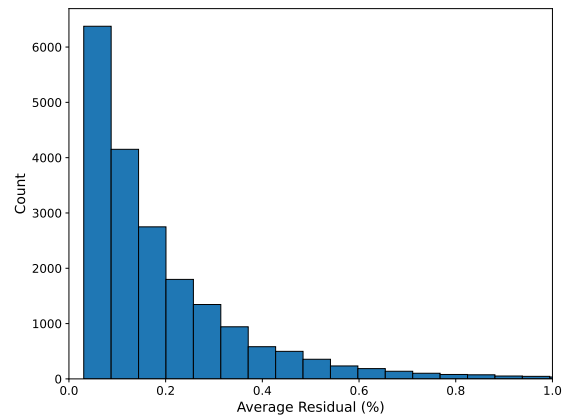


(b) Relative residuals for the test dataset (%)

Fig. 5 Distribution of relative residuals on CO₂ emissions for the single model



(a) Validation set



(b) Test set

Fig. 6 Distributions of per-case average residuals for the single model

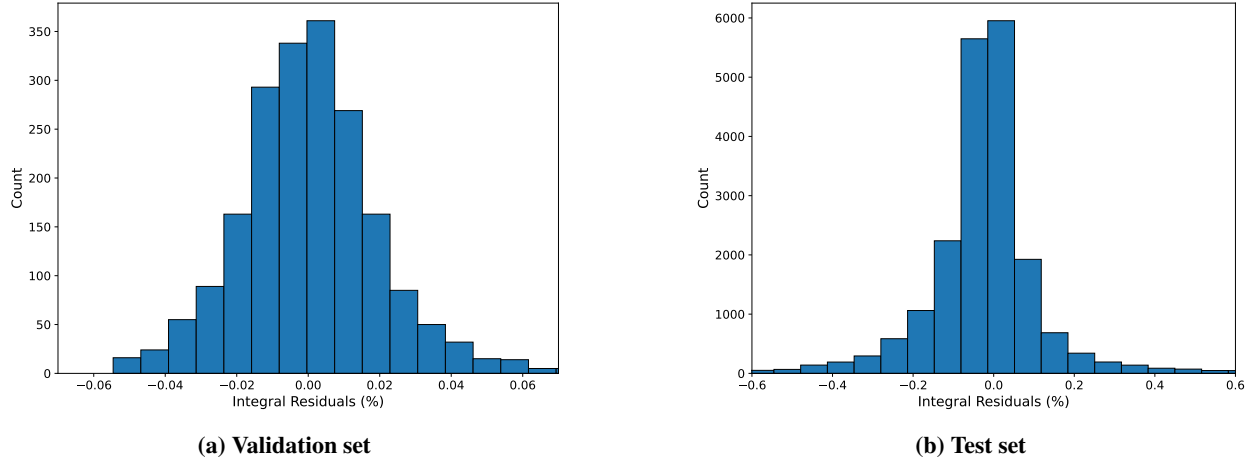


Fig. 7 Distributions of per-case relative residuals on the predicted cumulative CO₂ emissions for the single model

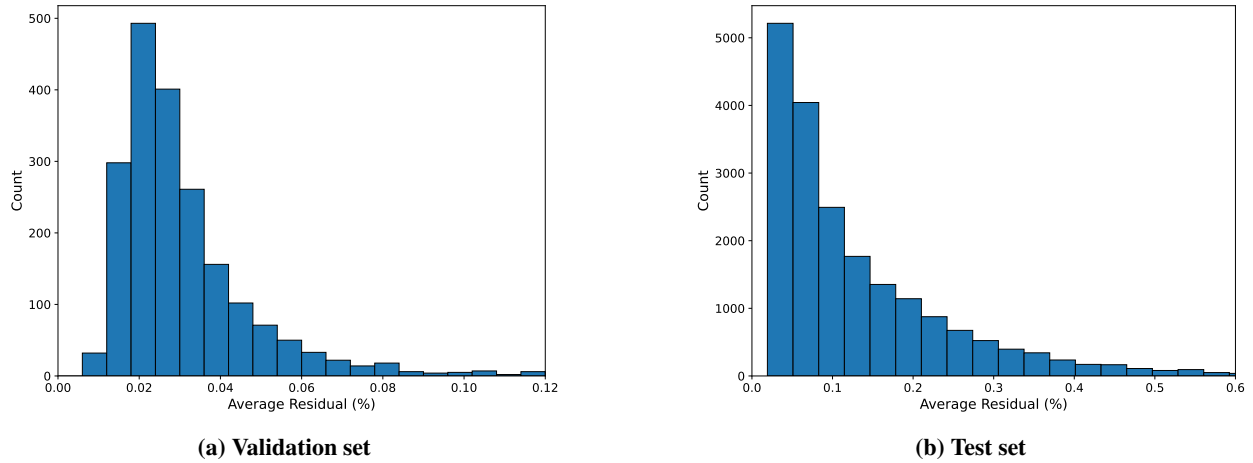


Fig. 8 Distributions of per-case average residuals for the ensemble

erroneous model on the ensemble prediction error. The validation and test prediction performances of the ensemble is shown in Fig. 10. We can see that using an ensemble of 10 models reduces the residual errors on both the validation and test datasets. This improvement in fitting performance is also observed on the average and integral residual distributions, as shown in Fig. 8 and 9.

Interestingly, the ensemble generalizes better than a single model, as the standard deviation of the residuals is halved on the test dataset after 2055. This improvement might be due to edge cases between in-distribution and out-of-distribution data, for which each single model would be extrapolating slightly differently, but after averaging between the 10 models of the ensemble, this overall extrapolation error remains mitigated. However, some cases are still out-of-distribution and need to be identified as such. Fig. 11 shows the maximum residual error over time with respect to the maximum standard deviation in model prediction over time, for each case in the test dataset. The correlation coefficient of 0.85 between these two metrics indicates a strong linear relationship, reinforcing the idea that the uncertainty in predictions within the ensemble can be used to detect out-of-distribution cases likely to lead to erroneous predictions of CO₂ outputs. Thus, we can use the predicted uncertainty to filter out or flag potential outliers at inference time. In Fig. 10c, we can see that filtering the test cases to only keep those for which the predicted uncertainty remains lower than 0.4 efficiently removes outliers with high residual errors, and successfully identifies cases for which the surrogate model generalizes well.

Thus, using an ensemble of models enables self-aware surrogate models with a simple way to avoid extrapolation at

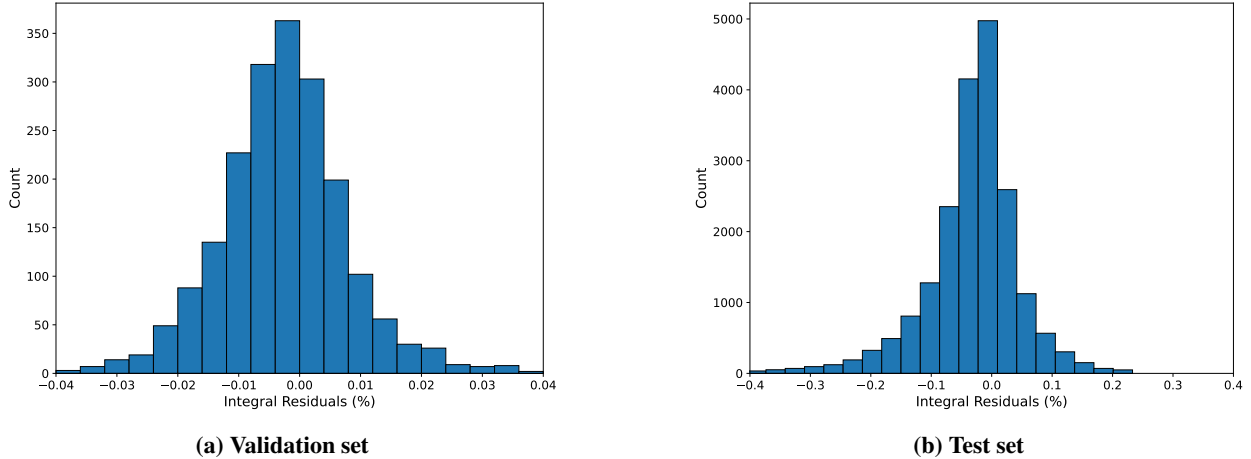


Fig. 9 Distributions of per-case relative residuals on the predicted cumulative CO₂ emissions for the ensemble

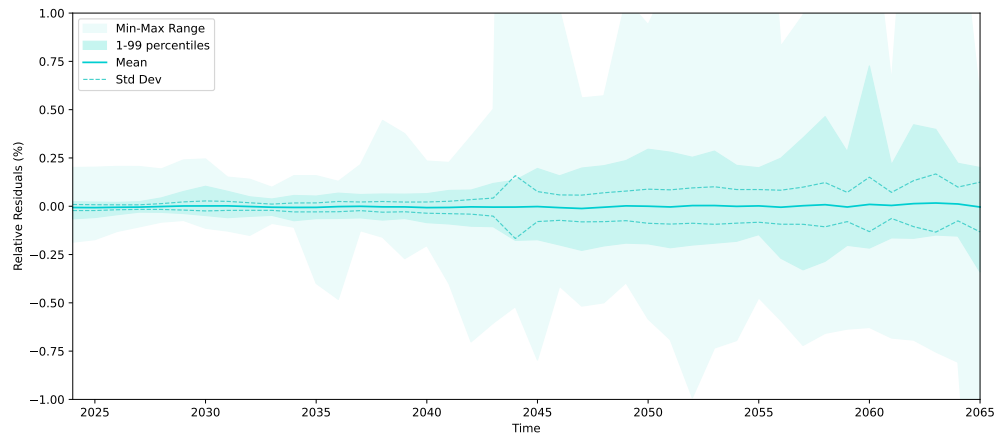
inference time. This is especially important for time-series inputs, as unlike scalar variables for which the training input space is well-defined, there is no simple way of checking for OOD cases with time series.

VI. Sensitivity Analysis

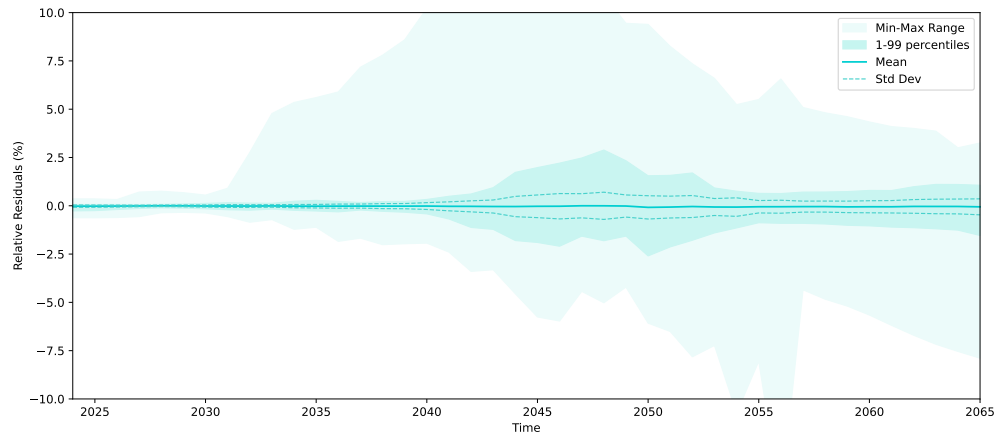
In the methodology introduced by [4] and [5], transition scenarios for the aviation industry are composed of various stakeholder decisions, made at different points in time. In order to evaluate the CO₂ emissions resulting from specific chains of decisions, each scenario is translated into time-series variables representing the industry impacts of these decisions. However, due to uncertainty in the magnitude of those impacts, some randomness was included into the translation from decisions to impact time series, as depicted in Fig. 1b. Thus, the ensemble surrogate model constructed in the previous sections enables the analysis of the distribution of CO₂ emissions caused by this uncertainty in decision impacts. Specifically, we decided to analyze the distribution of potential CO₂ emissions resulting from Scenario 1 and Scenario 4 as described in section B of the Appendix of [5]. To do so, we ran a Monte Carlo simulation where for each case, the scenario decisions remain the same but their respective impacts vary randomly within the ranges defined by the Impact Matrix provided in [4]. The surrogate ensemble was used to predict the CO₂ emission trajectory corresponding to each case of the Monte Carlo simulation, thus providing a distribution of potential CO₂ emissions resulting from each scenario. These results are shown in Fig. 12. In the first years until 2040, we observe that the CO₂ emissions have low uncertainty as decisions have not had enough time to start significantly impacting the industry. Then, we notice that the uncertainty in CO₂ emissions resulting from both transition scenarios increases during the transition phase from 2040 to 2060. This transition phase corresponds to the time when most critical decisions have been made and they are gradually starting to create changes in the industry, as improved aircraft and more efficient fuels are getting adopted. Due to uncertainty in the time needed for each decision to take effect, as well as uncertainty in the magnitude of these effects, this transition period results in higher uncertainty in CO₂ emissions. Finally, after 2060, the emissions stabilize and we observe a drop in uncertainty. It is particularly interesting to see that despite high uncertainty in the decisions' impacts, the amount of reduction in emissions achieved by 2065 remains rather robust in both cases, with an uncertainty under $\pm 3\%$ of 2024 CO₂ emissions in the final year of simulation. These results highlight that while a lot of uncertainty remains on the potential improvements brought about by new technologies and operational paradigms, it is still possible to identify transition pathways leading to high reductions in emissions and robust to this uncertainty.

VII. Conclusions

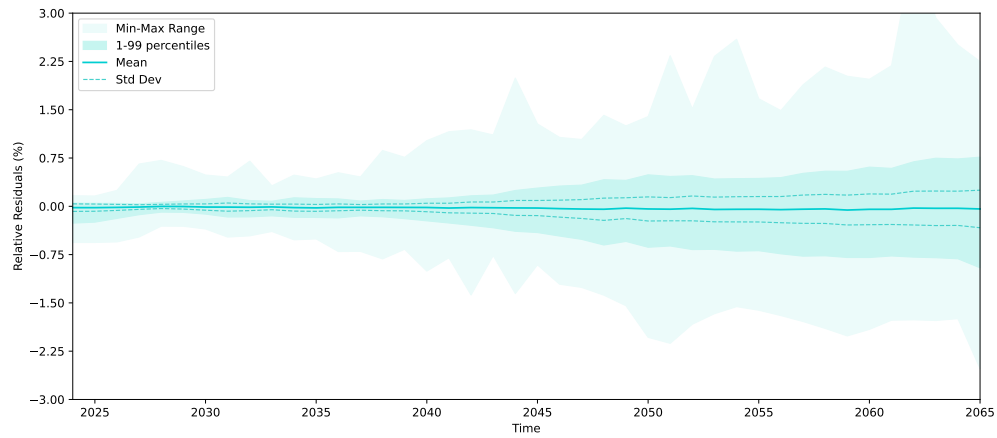
This study set out to construct a surrogate model of the scenario analysis introduced by [4] and [5], to improve the scalability of the approach by accelerating the scenario evaluation step and provide the ability to analyze the robustness of each scenario to variations in its decisions' impacts. For that purpose, a novel approach to generating designs of experiments for problem with time-series input variables has been introduced. It has been shown that this methodology has the potential to outperform existing scalar DoE methods on time-series problems. Moreover, by extending the



(a) Relative residuals for the validation dataset (%)



(b) Relative residuals for the test dataset (%)



(c) Relative residuals for the test dataset after filtering (%)

Fig. 10 Distribution of relative residuals on CO₂ emissions for the ensemble model

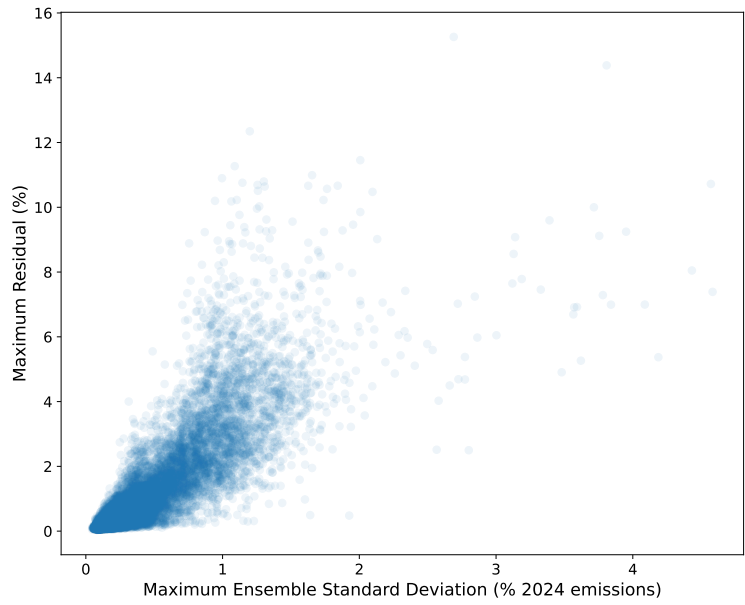
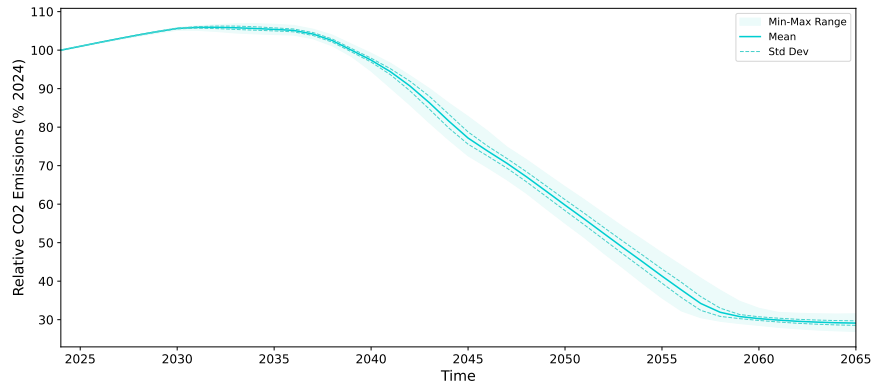
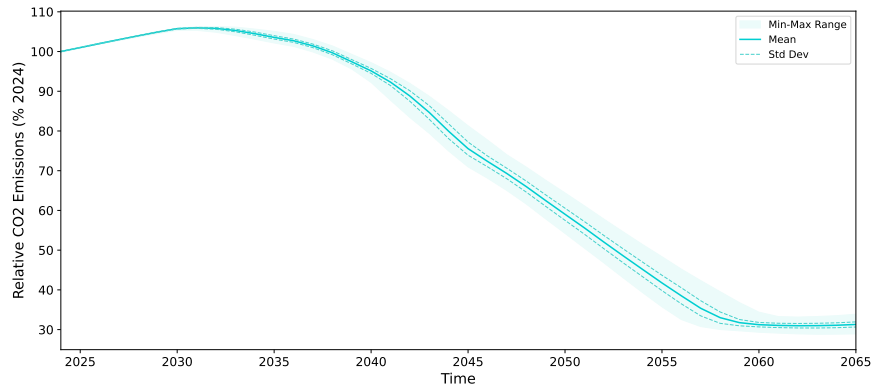


Fig. 11 Correlation between the inner uncertainty of the ensemble and the maximum prediction error



(a) Distribution of relative residuals each year for the validation dataset.



(b) Distribution of relative residuals each year for the test dataset

Fig. 12 Relative residuals for the ensemble model.

definition of usual DoE evaluation metrics to the temporal case, this paper has provided a baseline to which future temporal does can be compared. Moreover, this work provided a review on deep learning approaches to time-series forecasting, confirming that the TCN approach developed in [20] has the potential to outperform RNNs on time-series prediction tasks. Moreover, we provided a simple ensemble-based approach to improve the performance of time-series predictions and enable self-aware surrogate models able to identify extrapolation at inference time. Finally, the trained ensemble was used to analyze the robustness of the emission reductions achieved through the best-performing transition scenarios identified in [5]. We showed that despite high uncertainty in each decision’s impact, the overall reduction in CO₂ emissions can be quite robust. We believe that such insights into the robustness of transition scenarios are instrumental in supporting the decision-making process of the aviation industry stakeholders.

Appendix

A. Information Matrix derivation

We consider

$$\tilde{Y} = \tilde{X}\beta + \tilde{\varepsilon}.$$

which is the tensor notation of the N linear models

$$\tilde{Y}_n = \tilde{X}_n\beta + \tilde{\varepsilon}_n.$$

Assuming that the temporal components of the noise are not correlated, i.e.,

$$\forall n, \tilde{\varepsilon}_n \sim \mathcal{N}(0, \mathbf{I}_T),$$

and assuming that the noise between different trials is also not correlated, we can write the likelihood function

$$\mathcal{L}(\beta) = \prod_{n=1}^N p(\tilde{Y}_n | \tilde{X}_n, \beta) = \frac{1}{(2\pi)^{\frac{NT}{2}}} \exp\left(-\frac{1}{2} \sum_n \|\tilde{Y}_n - \tilde{X}_n\beta\|^2\right).$$

Thus, the log-likelihood is

$$\ell(\beta) = -\frac{NT}{2} \log(2\pi) - \frac{1}{2} (\tilde{Y}_n - \tilde{X}_n\beta)^\top (\tilde{Y}_n - \tilde{X}_n\beta)$$

After careful derivations, we find that

$$\forall i, j, \frac{\partial^2 \ell}{\partial \beta_i \partial \beta_j} = -\mathcal{I}(\beta)_{ij} = -\sum_{n=1}^N \left(\tilde{X}_n^{(j)\top} \tilde{X}_n^{(i)} \right)$$

and we conclude that the information matrix is

$$\mathcal{I}(\beta) = \sum_{n=1}^N \tilde{X}^{(n)\top} \tilde{X}^{(n)}.$$

Acknowledgments

The authors would like to express their gratitude to the project sponsor, NASA Glenn Research Center, for their support. The authors would particularly like to acknowledge Dr. Eric Hendricks and Mr. Ty Marien for their guidance and feedback on this work. The team also wishes to acknowledge the valuable guidance provided by Chrysoula Lydia Pastra, which was instrumental in the realization of this project.

References

- [1] International Civil Aviation Organization, “Long-Term Aspirational Goals (LTAG),” Tech. rep., ICAO, 2022. URL <https://www.icao.int/environmental-protection/LTAG/Pages/LTAGreport.aspx>.

- [2] International Air Transport Association, “Air Passenger Market Analysis,” Tech. rep., IATA, 2023. URL <https://www.iata.org/en/iata-repository/publications/economic-reports/air-passenger-market-analysis-july-2023/>.
- [3] Air Transport Action Group (ATAG), “Waypoint 2050,” , 2021. URL <https://www.icao.int/environmental-protection/LTAG/Pages/LTAGreport.aspx>.
- [4] Almarzooqi, R. M., Chen, H., Leicht, B. W., Bagdatli, B., Pfaender, H., and Mavris, D., “Scenario Generation for Strategic Decision-Making Towards Net-Zero Aviation,” *AIAA SCITECH 2025 Forum*, 2025. <https://doi.org/10.2514/6.2025-2838>.
- [5] Baali, I., Vinod, Y., Almarzooqi, R., Bagdatli, B., Pfaender, H., and Mavris, D., “Scenario Modeling and Analysis for Strategic Decision-Making Towards Net-Zero Aviation,” *AIAA SCITECH 2025 Forum*, 2025. <https://doi.org/10.2514/6.2025-2839>.
- [6] Hassan, M., Pfaender, H., and Mavris, D., “Probabilistic assessment of aviation CO2 emission targets,” *Transportation Research Part D: Transport and Environment*, Vol. 63, 2018, pp. 362–376. <https://doi.org/https://doi.org/10.1016/j.trd.2018.06.006>.
- [7] Federal Aviation Administration, “FAA Aerospace Forecast,” Report, 2023. URL <https://www.faa.gov/dataresearch/aviation/aerospaceforecasts/faa-aerospace-forecast-fy-2024-2044>.
- [8] Kirby, M. R., “A methodology for technology identification, evaluation, and selection in conceptual and preliminary aircraft design,” Ph.D. thesis, Georgia Institute of Technology, 2001. URL <https://www.proquest.com/dissertations-theses/methodology-technology-identification-evaluation/docview/251778503/se-2>.
- [9] Huan, X., Jagalur, J., and Marzouk, Y., “Optimal experimental design: Formulations and computations,” , 2024. URL <https://arxiv.org/abs/2407.16212>.
- [10] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., “Generative Adversarial Nets,” *Advances in Neural Information Processing Systems*, Vol. 27, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf.
- [11] Kingma, D. P., and Welling, M., “Auto-Encoding Variational Bayes,” , 2022. URL <https://arxiv.org/abs/1312.6114>.
- [12] Ho, J., Jain, A., and Abbeel, P., “Denosing Diffusion Probabilistic Models,” *Advances in Neural Information Processing Systems*, Vol. 33, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Curran Associates, Inc., 2020, pp. 6840–6851. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.
- [13] Desai, A., Freeman, C., Wang, Z., and Beaver, I., “TimeVAE: A Variational Auto-Encoder for Multivariate Time Series Generation,” , 2021. URL <https://arxiv.org/abs/2111.08095>.
- [14] Eldar, Y., Lindenbaum, M., Porat, M., and Zeevi, Y., “The farthest point strategy for progressive image sampling,” *IEEE Transactions on Image Processing*, Vol. 6, No. 9, 1997, pp. 1305–1315. <https://doi.org/10.1109/83.623193>.
- [15] Li, J., Zhou, J., Xiong, Y., Chen, X., and Chakrabarti, C., “An Adjustable Farthest Point Sampling Method for Approximately-sorted Point Cloud Data,” , 2022. URL <https://arxiv.org/abs/2208.08795>.
- [16] Müller, M., “Dynamic time warping,” *Information Retrieval for Music and Motion*, Vol. 2, 2007, pp. 69–84. https://doi.org/10.1007/978-3-540-74048-3_4.
- [17] Nketiah, E. A., Chenlong, L., Yingchuan, J., and Aram, S. A., “Recurrent neural network modeling of multivariate time series and its application in temperature forecasting,” *PLOS ONE*, Vol. 18, No. 5, 2023, pp. 1–18. <https://doi.org/10.1371/journal.pone.0285713>.
- [18] Gers, F., Schmidhuber, J., and Cummins, F., “Learning to forget: continual prediction with LSTM,” *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, Vol. 2, 1999, pp. 850–855 vol.2. <https://doi.org/10.1049/cp:19991218>.
- [19] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y., “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” , 2014. URL <https://arxiv.org/abs/1412.3555>.
- [20] Bai, S., Kolter, J. Z., and Koltun, V., “An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling,” , 2018. URL <https://arxiv.org/abs/1803.01271>.
- [21] Breiman, L., “Bagging predictors,” *Machine Learning*, Vol. 24, No. 2, 1996, pp. 123–140.