Improving Interpretability in Trajectory Generation: A Case Study on Efficient Latent Space Utilization

Mohammed El Dor, Marta Ribeiro, Junzi Sun, Jacco Hoekstra
Operations & Environment, Aerospace Faculty
Delft University of Technology
Delft, The Netherlands

Abstract—**The development of realistic synthetic trajectory data is essential for the planning and validation of future Air Traffic Management concepts. Deep generative models have emerged as state-of-the-art methods for synthetic trajectory generation due to their ability to learn complex dynamics. In particular, Variational Autoencoders (VAEs) have gained significant attention because of their stable training. However, for the generated data to be trusted, model interpretability is crucial. A key aspect for interpretability is the analysis of the latent space, where the essential features and patterns of trajectory behavior are encoded. However, in practice, many latent dimensions remain weakly utilized or collapse toward the prior, limiting interpretability and effective use of latent capacity.**

In this work, we study latent dimension activity in temporal VAEs trained on real-world aircraft trajectory data, with a specific focus on how Kullback-Leibler (KL) divergence regularization, a penalty added to the model's loss function, affects the number of active latent dimensions. We adopt a temporal convolutional VAE with fixed latent dimensionality and formulate the KL divergence as an average over latent dimensions, thereby controlling the information content per dimension.

Additionally, we then introduce a modified training objective that explicitly suppresses near-collapsed dimensions through masking and reweighting of the KL term. Our results show that this intervention significantly increases the number of active latent dimensions without degrading reconstruction accuracy. Ultimately, this will improve understanding of the decisions behind trajectory generation, enabling wider adoption.

Keywords—**Variational autoencoders, latent space utilization, posterior collapse, trajectory modeling**

I. Introduction

The transition toward Trajectory-Based Operations (TBO) fundamentally changes the role of aircraft trajectories in air traffic management (ATM). Rather than being implicit outcomes of tactical control, trajectories become explicit, shared, and continuously updated objects used by ground automation and decision-support systems. Future ATM concepts assume that rich trajectory information will replace traditional flight plans and serve as a common reference for planning, synchronization, and complexity management across stakeholders [1]. This vision implicitly requires reliable models that can compactly represent, reproduce, and generate realistic trajectories from data, an aspect that remains largely underexplored in practical settings, especially when generative use (sampling) is required [2].

Deep generative models provide a natural framework for learning compact latent representations of complex trajectory distributions. Recent work has demonstrated that VAEs in particular can learn realistic aircraft trajectory distributions in terminal areas and can serve downstream tasks such as uncertainty quantification and risk estimation [5]–[8]. However, despite their promise, VAEs often exhibit a recurring practical issue: only a subset of latent dimensions meaningfully contributes to the representation, while many dimensions become effectively inactive. The result is that the model is not fully utilizing its capacity, and less meaningful information is used for creation of the trajectories. Additionally, models trained for high-fidelity reconstruction, such as VAEs, can produce latent distributions that deviate from the assumed prior. Potentially, different dimensions of the latent space may be used. These new dimensions may lead to the production of new, unrealistic trajectories. In trajectory generation contexts relevant to TBO, both phenomena are problematic: unused latent capacity reduces representational efficiency, and poor prior sampling undermines the ability to generate plausible "what-if" trajectories on demand.

This work revisits the frequently used VAE approach for trajectory modelling with a specific focus on the behavior and utilization of latent dimensions under KL divergence regularization. In other words, we aim to increase the number of active latent dimensions to produce more meaningful and interpretable trajectories, while reducing empty dimensions that could lead to unrealistic or inconsistent trajectories when sampled. We adopt a temporal convolutional VAE architecture inspired by prior work on terminal-area trajectory modeling [5], but re-implement it in a streamlined setting aligned with our data representation and analysis objectives. The contributions of this paper are as follows:

- A controlled empirical analysis of latent dimension activity in temporal VAEs trained on real-world trajectory data, using a KL divergence formulated as an average over latent dimensions.
- A principled procedure for selecting the KL weight based on an explicit reconstruction reference and a target KL value per latent dimension, enabling interpretable and reproducible regularization scaling.
- An experimental evaluation of masking and weighting

strategies as anti-collapse interventions, demonstrating their effect on the number of active latent dimensions under fixed model capacity.

The remainder of the paper is organized as follows. Section II reviews related work on data-driven trajectory modeling and generative approaches. Section III describes the data representation, model architecture, and training objective. Section IV details the experimental setup. Section V presents reconstruction, KL regularization, and latent activity results, including comparisons between non-masked and masked objectives, and provides qualitative generation examples. Finally, Section VI concludes the paper and discusses implications for trajectory modeling in TBO contexts.

II. Literature Review

Data-driven approaches rely on statistical and machine learning methods that aim to predict or generate trajectories directly from past observations. When it comes to modelling flight trajectories, stochastic processes appear to be the most appropriate tool: within a given flow, every flight is a realization of the same random function [10]. Unlike model-driven methods, no simplifications of the physical models are required, and uncertainty can be incorporated more naturally [11]. By learning directly from observed trajectories, these methods approximate the underlying data distribution and generate new samples that reflect real-world variability. Additionally, in model-driven methods, only the consequences of uncertainty (e.g., deviations in trajectories) can be reproduced, while the sources of uncertainty such as ATC maneuvers, human factors, or unexpected weather remain unmodeled [7], [10]. Data-driven methods overcome this limitation by directly estimating the distribution of observed trajectories and sampling from it, which makes synthetic trajectory generation more representative of actual operations.

Several trajectory generation approaches are based on Markov chain formulations. For example, [12] apply a Markov chain model to observed en-route trajectories, where stochasticity captures deviations caused by wind, navigation, or ATC interventions. In their work, randomness is introduced into a 2-D straight flight path with constant speed, and importance splitting is then applied to estimate conflict probabilities. Although this demonstrates the potential of Markov-based methods for incorporating randomness, the approach remains limited to simple, straight trajectories with constant altitude and speed.

Clustering methods have also been applied. Eckstein et al [13] use K-means clustering to define groups of 1-D climb profiles, with each centroid representing the nominal trajectory of its cluster. Murça et al, for example, employ clustering for arrival trajectories at São Paulo Airport, modeling each cluster with a multivariate Gaussian distribution [2]. Synthetic trajectories are then generated by sampling from the corresponding distribution.

Dimensionality reduction is another technique for trajectory generation. Eckstein et al [14] apply Principal Component Analysis (PCA) to project trajectories onto lower-dimensional spaces and generate new trajectories within this latent representation. First, this is extended by projecting altitude and speed profiles using PCA, fitting a distribution in the latent space, and generating new profiles via inverse transformation [15]. Second, functional PCA is further developed as trajectories are now treated as continuous functions rather than discrete points [16]. However, PCA-based methods do not always ensure that generated trajectories fully respect the statistical properties of the original data.

More advanced machine learning methods have also been investigated. Generative Adversarial Networks (GANs) [17] represent a common non-linear alternative for trajectory generation, though they are notoriously difficult to train due to the Min-Max optimization problem. Moreover, GANs tend to reproduce common patterns effectively but may fail to generate rare or outlier trajectories, such as emergency maneuvers. Other statistical approaches include the use of statistical copulas, as explored by Timothe et al [10] for modeling the distribution of go-around trajectories at Zurich Airport. This method successfully reproduced lateral trajectory shapes but could not extend to vertical profiles.

To overcome the limitations of earlier approaches, recent research has turned to deep generative models. In particular, trajectory generation methods based on VAEs [7] [5] [6] have demonstrated strong potential. By learning a latent representation of complex four-dimensional trajectories, VAE-based models capture both common and rare behaviors while naturally incorporating uncertainty. However, most existing works primarily evaluate these models through reconstruction accuracy or downstream performance, while the structure, utilization, and effective dimensionality of the learned latent space remain largely unexplored. Extensions such as VQ-VAE and Deep VAE frameworks enable the generation of realistic, high-dimensional synthetic trajectories that more closely reflect operational variability [6].

Building on this line of research, Cho et al [18] introduced the use of flow matching techniques for generating trajectories. This method leverages probabilistic modeling to assess route feasibility in terminal airspace and provides a structured approach for capturing the inherent uncertainty of urban air mobility operations. Flow matching offers a promising alternative to classical generative methods by aligning generated trajectories with observed distributions in a more controlled and probabilistic manner. In addition, Sun et al [19] proposed a transformer-based trajectory prediction model conditioned on Eurocontrol B2B flight messages. This approach utilizes the attention mechanism to model long-range dependencies in flight trajectories and integrates operational information from B2B datasets. The method demonstrates strong potential for supporting air traffic demand forecasting by generating realistic trajectory patterns that adapt to both traffic flow and operational constraints.

It is evident that VAEs are the state-of-the-art in the explored architectures. However, to the best of the author's knowledge, no work has yet covered the aspect of explainability of the generation procedure. This paper aims to cover this gap, targeting the latent space to increase explainability. By making the latent dimensions more uniformly utilized, we aim to achieve disentanglement between the dimensions, thereby enabling a clearer understanding of how each latent variable contributes to the final result. We apply to this work to the use case of aircraft trajectories generation.

III. Methodology - Variational Autoencoder Architecture

We adopt a VAE architecture for modeling aircraft trajectories as fixed-length multivariate time series. The overall design is inspired by prior work on temporal convolutional VAEs for trajectory generation in terminal airspace, notably the Temporal Convolutional Networks (TCN)–VAE framework introduced in [5], which demonstrated that temporal convolutional encoders and decoders are well suited to capture the strong temporal correlations present in aircraft trajectories.

In contrast to classical fully connected VAEs, the encoder and decoder are implemented using TCNs, enabling efficient modeling of long-range temporal dependencies while preserving causality. The encoder maps each input trajectory, $q_\phi(z \mid x)$, to the parameters of a diagonal Gaussian posterior distribution:

$$q_\phi(z \mid x) = \mathcal{N}(\mu(x), \mathrm{diag}(\sigma^2(x))),$$

where $\mu \in \mathbb{R}^D$ and $\sigma^2 \in \mathbb{R}^D$ denote the latent mean and variance, respectively, and D is the latent dimensionality. Latent samples are obtained using the reparameterization trick [3].

Prior work [5] [7] employs a VampPrior to increase the flexibility of the latent prior distribution and improve coverage of the latent space for trajectory generation. This introduces complexity in the latent distribution, making the interpretation and analysis of latent dimensions more difficult. In this study, we replace the VampPrior with a standard isotropic Gaussian prior $p(z) = \mathcal{N}(0, I)$. This choice is motivated by the objective of the paper: rather than maximizing sample diversity alone, we aim to study how the balance between reconstruction fidelity and latent regularization affects latent dimension utilization, posterior collapse, and downstream sampling behavior. A simple prior allows direct inspection of per-dimension KL divergence and facilitates controlled experiments on the effect of the β coefficient in the VAE objective, which will be further explained in Subsection III-A.

The decoder mirrors the encoder architecture and reconstructs trajectories from latent samples by mapping z back to the original time-series space. Trajectory generation is then performed by sampling from structured regions of the latent space obtained through clustering of encoded latent means. This approach allows us to separate the effects of latent regularization from those of sampling strategy, which is central to the experimental analysis presented in this paper.

A. Training Objective and Loss Formulation

The model is trained by minimizing total loss $\mathcal{L}_{\mathcal{TOTAL}}$:

$$\mathcal{L}_{\text{TOTAL}} = \mathcal{L}_{\text{recon}} + \beta \cdot w_{\text{KL}}(t) \cdot \mathcal{L}_{\text{KL}}, \tag{1}$$

where:

- The reconstruction term $\mathcal{L}_{\text{recon}}$ is defined as the mean squared error (MSE) between the reconstructed and original trajectories, where B denotes the batch size, F the number of features, and T the sequence length. Let $x_b \in \mathbb{R}^{F\times T}$ denote an input trajectory and $\hat{x}_b$ its reconstruction. The reconstruction loss is given by:

$$\mathcal{L}_{\text{recon}} = \frac{1}{B \cdot F \cdot T} \sum_{b=1}^{B} \sum_{f=1}^{F} \sum_{t=1}^{T} \left(\hat{x}_{b,f,t} - x_{b,f,t}\right)^2 \tag{2}$$

- β controls the relative importance of latent regularization.
- $w_{\text{KL}}(t)$ is a time-dependent warm-up factor that increases according to a polynomial warm-up schedule and remains constant thereafter, gradually increasing the influence of the KL term during training.
- The latent loss term $\mathcal{L}_{\text{KL}}$ is defined as:

$$\mathcal{L}_{\text{KL}} = \frac{1}{B \cdot D} \sum_{b=1}^{B} \sum_{d=1}^{D} \mathrm{KL}_{b,d} \tag{3}$$

where D denotes the latent dimensionality. This formulation explicitly controls the information content per latent dimension while allowing the total KL divergence to scale with the latent capacity. The encoder outputs the parameters of a diagonal Gaussian posterior $q_\phi(z \mid x_b) = \mathcal{N}(\mu_b, \mathrm{diag}(\sigma_b^2))$, while the prior is chosen as an isotropic standard normal distribution $p(z) = \mathcal{N}(0, I)$. For each latent dimension d, the dimension-wise KL divergence for sample b is given by:

$$\mathrm{KL}_{b,d} = \frac{1}{2}\left(\mu_{b,d}^2 + \sigma_{b,d}^2 - 1 - \log \sigma_{b,d}^2\right) \tag{4}$$

In contrast to the classical β-VAE formulation, where the KL divergence is summed over latent dimensions, we compute the KL term as an average over latent dimensions and over the batch.

In a second stage of experimentation, we introduce a modified masked and weighted KL regularization $\mathcal{L}_{\text{KL}}^{\text{masked}}$. Instead of penalizing all latent dimensions equally, the KL term is computed only over latent dimensions whose average KL divergence exceeds a predefined threshold, indicating non-negligible information content. The remaining active dimensions are reweighted according to their relative KL magnitude, encouraging a more balanced allocation of information across the latent space. Let KL_d denote the KL divergence of latent dimension d, averaged over the batch. The masked and weighted KL objective is defined as:

$$\mathcal{L}_{\text{KL}}^{\text{masked}} = \frac{\sum_{d\in\mathcal{A}} w_d \, \mathrm{KL}_d}{\sum_{d\in\mathcal{A}} w_d}, \tag{5}$$

where $\mathcal{A}$ denotes the set of active latent dimensions and w_d are dimension-specific weights. Masking prevents near-collapsed dimensions from contributing to the KL objective, ensuring that the regularization signal is concentrated on informative dimensions. Inactive dimensions may re-enter the active set only if reconstruction gradients increase their KL above the threshold. The new total training objective is thus given by:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \beta \cdot w_{\text{KL}}(t) \cdot \mathcal{L}_{\text{KL}}^{\text{masked}} \tag{6}$$

We should note that this masked KL objective is a practical anti-collapse heuristic and does not preserve a strict ELBO interpretation as demonstrated in [3].

IV. Experimental design

This section explains the employed methods in Section III. In this section we present the data processing choices and assumptions (see Subsection IV-A) before delving into the reasoning behind our choices of β values (Subsections IV-B and IV-C). We then explain how the masked loss was deployed and the masking threshold (Subsection IV-D). Finally, we discuss how latent activity and latent collapse are detected and parametrized (Subsection IV-E).

A. Problem Setup and Data Representation

We model aircraft trajectories as multivariate time series of fixed length. Each trajectory is represented as a matrix in $\mathbb{R}^{F\times T}$, where F denotes the number of features and T the number of time steps. In this work, $T = 200$ and $F = 4$, corresponding to planar position (x, y), altitude, and time increment Δt between successive observations.

The trajectory dataset used in this study is derived from the arrival trajectories to runway 14, trimmed to a circle of radius 40 nm, at Zurich Airport introduced in [5]. In the original work, as well as in this work, trajectories were extracted from ADS-B data provided by the OpenSky Network [20], trimmed within the Arrival Sequencing and Metering Area (ASMA), temporally resampled to a fixed length of 200 points using linear interpolation, and aligned to a common terminal reference point along the final approach. This preprocessing yields smooth and temporally consistent trajectories that are representative of complex terminal-area operations. In contrast to the original paper, instead of using track angle and ground speed, we present trajectories directly in Cartesian coordinates (x, y) together with altitude and time increments. This representation simplifies the interpretation of reconstruction errors in the physical plane and avoids learning implicit geometric correlations between angular and velocity-based variables.

Prior to training, each feature is normalized to the range $[0, 1]$ using min–max scaling computed over the training set. The dataset is split into training and validation subsets using a fixed random seed to ensure reproducibility. All experiments reported in this work rely on the same data split and preprocessing pipeline.

B. KL Weight, β, and Latent Regularization

The objective of this work is to analyze how the choice of the KL weight β influences the reconstruction loss and the activity and utilization of latent dimensions. We do not tune β to optimize a single performance metric, instead we use β as a scale parameter to control the relative strength of latent regularization with respect to a known reconstruction error scale according to the optimization objective that minimizes the total losses presented in equations 1 and 6. Specifically, we consider a low β value where reconstruction dominates, and then a higher β, where increased KL regularization pressure may lead to partial or complete posterior collapse of latent dimensions. This controlled variation allows us to characterize how latent dimensions become inactive as KL pressure increases, and to assess the effectiveness of anti-collapse interventions under a fixed and interpretable regularization scale.

C. Loss Scaling and β Selection

The reconstruction loss $\mathcal{L}_{\text{recon}}$ defined in Equation (2) is computed in normalized space. The planar coordinates (x, y) originally span over a range of 80 nm. Each feature is independently scaled to $[0, 1]$ using min–max normalization fitted on the training as stated earlier. Based on that we build our reasoning later on upon 2 assumptions:

- We fix a minimum achievable reconstruction of value 5×10^{-7} and a minimum KL divergence of approximately $0.5\ nats/dim$. The chosen reconstruction loss was derived from the results of setting $\beta = 0$ in V-A. As for the KL divergence of 0.5, it was not a minimum value that we could not surpass, instead it was a minimum that we wish to not get below.
- The reconstruction loss is uniformly distributed among all 4 features. With that the reconstruction loss in equation 2 will be equal among all 4 features and hold the same value.

With both assumptions it can be approximated that the horizontal loss across the horizontal plane is approximately $105\ m$ along each axis x and y per point on the trajectory. This conversion provides a physical interpretation of the target reconstruction loss used in the experiments.

To define meaningful values of β, we specify target operating ranges for both loss terms as stated in the first assumption. A balanced reference value of β is therefore defined by equating the order of magnitude of the reconstruction and KL terms late in training:

$$\beta_{\text{balanced}} = \frac{\mathcal{L}_{\text{recon}}^{\text{target}}}{\mathcal{L}_{\text{KL}}^{\text{target}}} \approx 1 \times 10^{-6}.$$

Other β values are obtained by scaling this reference value by one order of magnitude in both directions, resulting reconstruction-dominated and regularization-dominated operating regimes.

D. Masked loss Threshold Selection

In Subsection III-A we discussed two KL divergence losses, and this section specifies the threshold chosen in the masking KL divergence objective defining set $\mathcal{A}$ in equation 5. The average KL divergence in non masked settings with highest regularization $\beta = 10^{-5}$ was 0.64 (check Table II), and since we want to allow a noticeable range between this value and the threshold, it was chosen that the threshold will be 1 for considering a dimension approaching collapse and excluded from the set $\mathcal{A}$. In the previous section (Subsection IV-C) we stated that the minimum was 0.5, as this values is what we desire to not exceed, but we can put a higher constraint as we are not optimizing, but rather studying the effect of β regularization.

E. Latent Activity and Collapse Metrics

After training with different β values and applying both losses presented in Subsection III-A, we aim to study the latent space utilization. Latent space utilization is assessed using a set of dimension-wise metrics, computed independently for each latent dimension on the validation set. These metrics are designed to capture both the presence of information in each dimension and the manner in which this information is encoded. For each latent dimension d, we compute:

- The KL divergence between the approximate posterior and the prior, averaged over the dataset KL_d; Equation 4 with the validation set as the batch,
- The expected posterior variance, $\mathbb{E}[\sigma_d^2]$.
- The standard deviation of the posterior mean $Std(\mu_d)$

Together, these quantities provide a structured description of latent behavior at the per-dimension level.

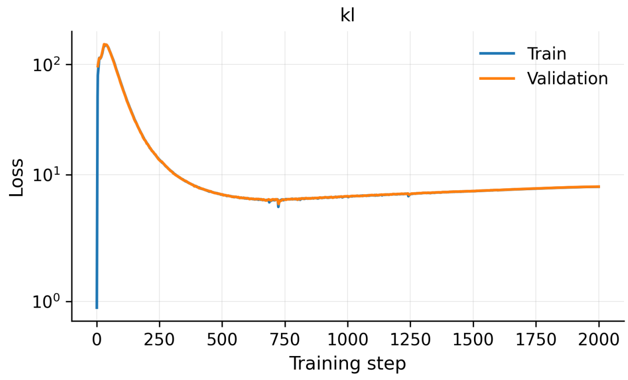
The KL divergence serves as the primary indicator of latent activity. A latent dimension is considered *active* if its average KL contribution exceeds a fixed threshold τ, indicating a non-negligible deviation from the prior and the presence of information about the input data. Dimensions whose KL contribution falls below this threshold are classified as inactive and interpreted as having collapsed toward the prior.

The remaining metrics are used to further qualify latent behavior. The standard deviation of μ_d captures the degree to which a latent dimension varies across the dataset and reflects data-dependent structure in the latent representation, while the expected posterior variance indicates the level of uncertainty associated with that dimension. Latent dimensions exhibiting negligible KL values, near-zero variation in μ_d, and posterior variance close to that of the prior are indicative of full posterior collapse, whereas active dimensions are characterized by non-trivial KL values, structured variation in μ_d, and posterior variances that deviate from the unit prior.

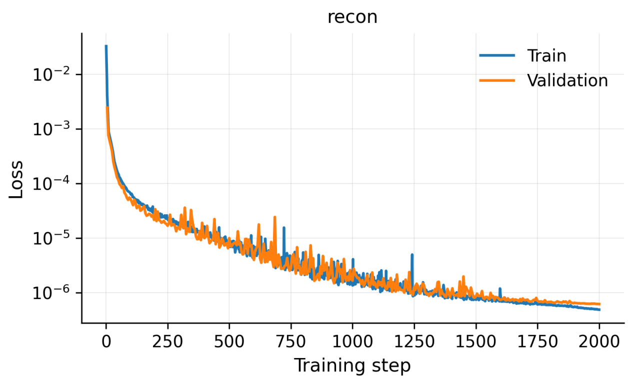
The mentioned criteria are assessed independently per dimension to ensure a balanced representation. A few dominant dimensions can lead to the rest of the dimensions to be relatively inactive. Thus, the balance and uniformity of these values across the latent dimensions is considered as an indication of latent utility.

V. Results

This section presents the empirical findings corresponding to the experimental design described in Section IV. We begin with the reconstruction-only baseline ($\beta = 0$) to establish the reference loss scale and characterize latent behavior without regularization pressure (Subsection V-A). We then analyze the effect of varying the KL weight β under both masked and non-masked objectives (Subsection V-B), including reconstruction–regularization trade-offs and steady-state KL behavior. Subsequently, we examine latent activity and posterior collapse across operating regimes, comparing masked and non-masked training (Subsection V-B, Latent activity analysis). Finally, we provide qualitative trajectory generation results at the balanced operating point $\beta = 10^{-6}$ (Subsection V-C).



(a) KL divergence (not optimized)



(b) Reconstruction loss

Figure 1. Training and validation loss curves for the reconstruction-only baseline ($\beta = 0$). The KL divergence is shown for diagnostic purposes but does not contribute to the optimization objective.

A. Reconstruction Baseline and Loss Scale ($\beta = 0$)

We first establish a reconstruction-only baseline by training the temporal VAE with $\beta = 0$, effectively removing the KL divergence from the optimization objective. This experiment provides a reference reconstruction scale and allows us to examine latent behavior in the absence of explicit regularization pressure. Figure 1 reports the evolution of the KL divergence (a) and the reconstruction loss (b). The KL divergence was reported even though no optimization was applied on it for both training and validation sets. Since $\beta = 0$, the total loss is equal to the reconstruction loss.

The reconstruction loss decreases rapidly during the early stages of training and continues to improve steadily over time, eventually converges to a final value of 5.35×10^{-7} and 6.17×10^{-7} in the training validation set, respectively. The close agreement between training and validation curves indicates stable optimization and no observable overfitting.

Although the KL divergence is not optimized in this regime, it is reported to characterize the latent behavior induced purely by reconstruction-driven training. After an initial transient phase, the KL divergence stabilizes at approximately 7.79 nats on the training set and 7.81 nats on the validation set. The decrease in the KL divergence throughout the training

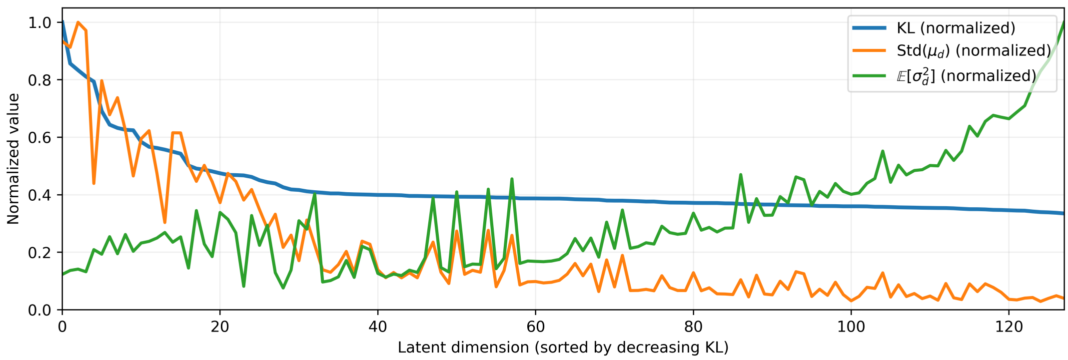


Figure 2. Dimension-wise latent statistics at convergence for the reconstruction-only baseline ($\beta = 0$), computed on the validation set. Latent dimensions are sorted by decreasing KL_d.

shows that better reconstruction requires the latent position of trajectories to approach the origin. It is theorized that this behavior is due to the variational sampling from the latent space that forces continuity and thus pushes trajectories closer to each other making them approach the origin.

Table I summarizes the final training and validation losses and defines the reconstruction error scale used to select and interpret β values in the following experiments.

As for the latent dimension's activity, the criteria discussed in Subsection IV-E were applied. Figure 2 visualizes these statistics across the $D = 128$ latent dimensions, sorted by KL_d. To enable comparison on a common axis, each metric is normalized to $[0, 1]$ independently across dimensions; the plot therefore emphasizes *relative* patterns rather than absolute magnitudes. Two qualitative behaviors are visible. First, the highest-KL_d dimensions also exhibit the largest variability in μ_d, indicating that these coordinates carry substantial data-dependent signal and contribute strongly to reconstruction. Second, many remaining dimensions show lower $\mathrm{std}(\mu_d)$ and larger posterior variances, consistent with weaker data dependence and greater uncertainty, even though KL regularization is absent. This non-uniform allocation of information suggests that latent capacity is not used evenly across dimensions, motivating the later analysis of latent activity and collapse under $\beta > 0$ and under explicit masking/weighting interventions.

B. *Effect of β on Reconstruction and KL Regularization*

After establishing the baseline in the previous section we start the experiments of varying β as explained in IV-C, tested values are $\beta \in \{10^{-7}, 10^{-6}, 10^{-5}\}$. These values were tested in combination with both objectives, masked and non-masked, presented in Subsection III-A.

In Figure 3 we see the corresponding validation reconstruction losses. Higher values of β lead to systematically worse reconstruction accuracy, reflecting the trade-off imposed by latent regularization. While $\beta = 10^{-7}$ achieves reconstruction errors close to the $\beta = 0$ baseline, the reconstruction loss increases by approximately one order of magnitude for $\beta = 10^{-5}$. Masking the loss function came with a slightly increased reconstruction loss. Importantly, all runs converge smoothly and exhibit stable validation behavior, indicating that the observed differences are due to regularization effects rather than optimization instability. Table II summarizes the final loss values for different beta values for both masked and non-masked losses.

In Figure 4, we see the evolution of the *validation* KL divergence during training for the different β values with both masking and non-masking objective functions. All models exhibit a similar initial KL transient during early training, followed by convergence to distinct steady-state KL levels. As expected, increasing β results in progressively stronger

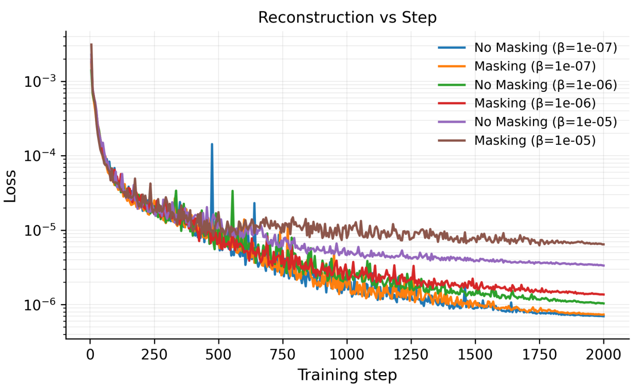


Figure 3. Validation reconstruction loss during training for different β values (log scale) for both masking and no-masking loss functions.

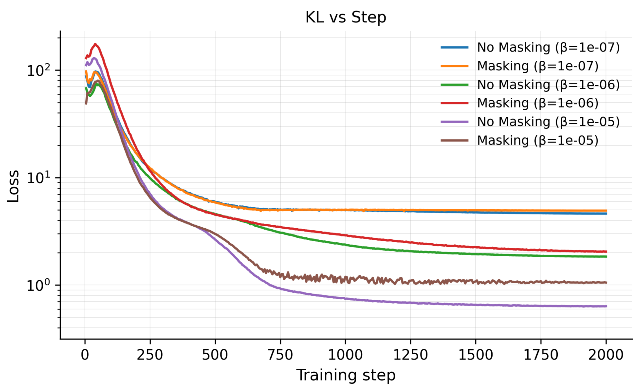
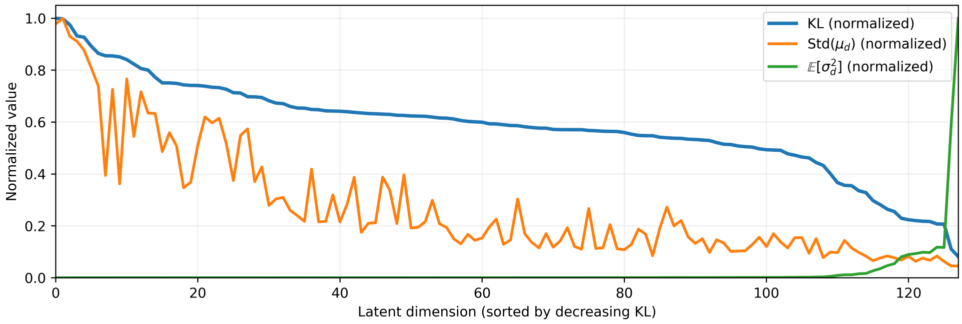
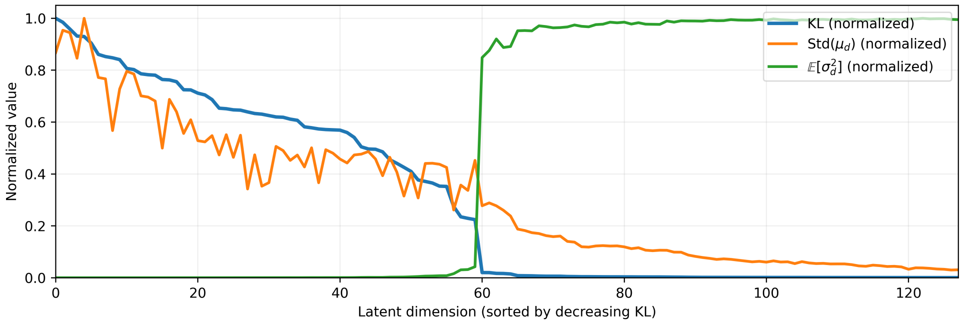


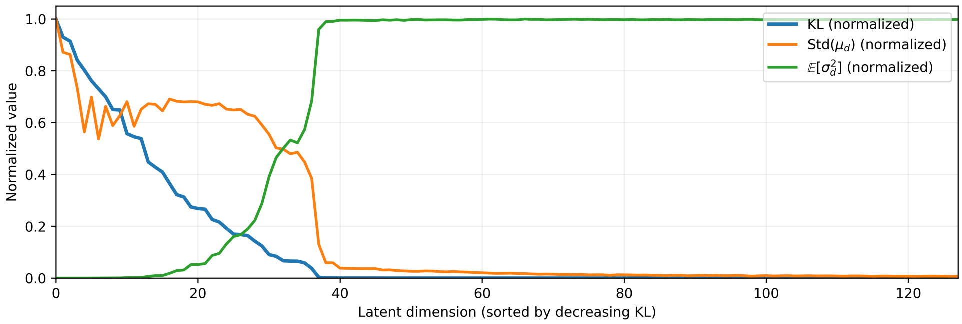
Figure 4. Validation KL divergence during training for different β values (log scale) for both masking and no-masking loss functions.



(a) Non-masked, $\beta = 10^{-7}$



(b) Non-masked, $\beta = 10^{-6}$



(c) Non-masked, $\beta = 10^{-5}$

Figure 5. Dimension-wise latent statistics at convergence for non-masked training across different β values.

pressure toward the prior, yielding lower final KL values. For $\beta = 10^{-7}$, the model retains a relatively high KL divergence, indicating weak regularization, whereas $\beta = 10^{-5}$ enforces a much tighter posterior alignment with the prior. Again the final values of the losses can be seen in Table II. To compare the different objectives with and without masking, we focus exclusively on the validation losses, since the training losses are weighted and dependent on the active dimensions and their relative strength (see Equation 5). To enable comparison, the validation loss for masked settings was recomputed based on the non masked criteria from Equation 3. The effect of masking on the validation KL divergence is more evident

TABLE I. Final training and validation losses for the reconstruction-only baseline ($\beta = 0$) at the end of training.

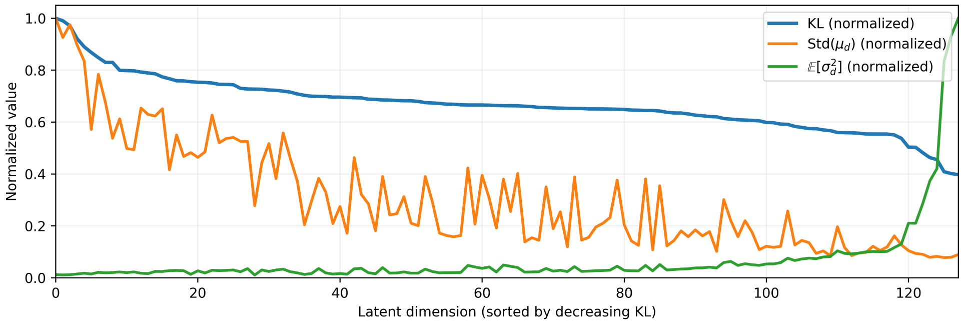
Metric	**Train**	**Validation**
Reconstruction loss	5.35×10^{-7}	6.17×10^{-7}
KL divergence	7.79	7.81
Total loss	5.35×10^{-7}	6.17×10^{-7}

TABLE II. Final losses at step 2000 for different β values. Reconstruction values are reported in units of 10^{-6}.

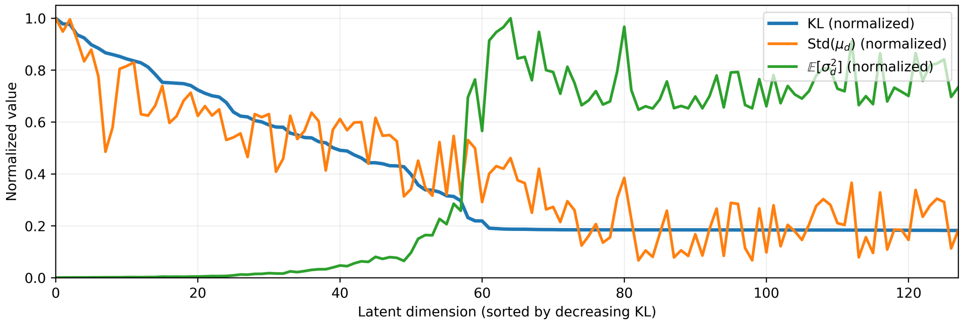
β	Mask	KL_{train}	KL_{val}	Rec_{train}	Rec_{val}
10^{-7}	No	4.62	4.63	0.607	0.697
	Yes	5.04	4.93	0.620	0.736
10^{-6}	No	1.84	1.84	0.928	1.040
	Yes	3.48	2.05	1.220	1.366
10^{-5}	No	0.63	0.64	2.890	3.350
	Yes	1.84	1.06	6.450	6.450

when β is higher. This confirms the expected behavior that, when β increases, more dimensions are prone to collapse. The masked KL divergence loss will thus have greater effect in stopping the collapse and increasing the average KL divergence. These results establish a clear operating spectrum for β and motivate the selection of $\beta \approx 10^{-6}$ as a balanced regime in which non-trivial KL regularization is achieved without a severe degradation of reconstruction fidelity.

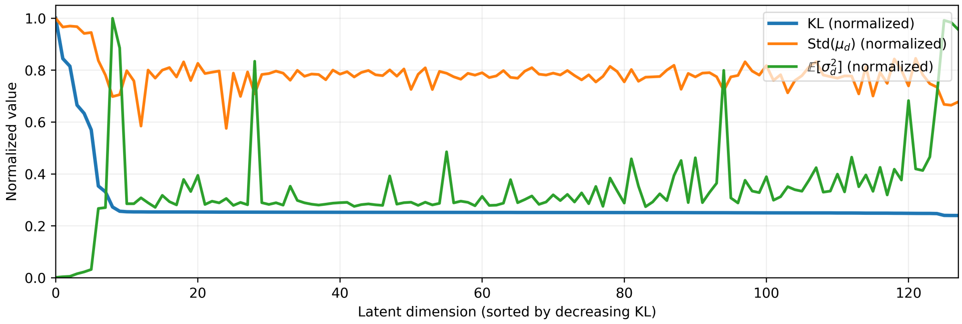
In the following sections, we analyze how these different



(a) Masked, $\beta = 10^{-7}$



(b) Masked, $\beta = 10^{-6}$



(c) Masked, $\beta = 10^{-5}$

Figure 6. Dimension-wise latent statistics at convergence for masked training across different β values.

operating points affect latent dimension activity and posterior collapse behavior.

1) Latent activity analysis: We analyze how latent information is distributed across dimensions under different KL regularization regimes and how masking affects this allocation. The analysis is based on the per-dimension statistics shown in Figs. 5 and 6, where latent dimensions are sorted by decreasing KL_d.

a) Non-masked training.: For weak regularization ($\beta = 10^{-7}$), the KL divergence is distributed across a relatively large subset of latent dimensions. Although a decreasing profile is visible, a substantial fraction of dimensions exhibits non-negligible KL contributions. The corresponding $\mathrm{std}(\mu_d)$ values follow a similar trend, indicating that these dimensions encode meaningful data-dependent variation. Posterior variances $\mathbb{E}[\sigma_d^2]$ remain below the prior level for many dimensions, confirming active latent usage.

At the balanced operating point ($\beta = 10^{-6}$), a separation emerges between active and inactive dimensions. A moderate subset of latent coordinates retains significant KL contributions, while the remaining dimensions exhibit KL values close to zero. These low-KL dimensions are characterized by near-zero variation in μ_d and posterior variances approaching that of the prior, consistent with partial posterior collapse. Importantly, KL mass becomes increasingly concentrated in the leading dimensions, indicating uneven allocation of information.

Under stronger regularization ($\beta = 10^{-5}$), collapse becomes pronounced. Only a small number of leading dimensions retain substantial KL contributions, while the majority collapses toward the prior. The corresponding $\mathrm{std}(\mu_d)$ values decrease sharply beyond the leading subset, and posterior variances approach unity for most dimensions. This regime is characterized by strong information concentration and effective reduction of usable latent capacity.

b) Masked training.: Introducing the masking strategy alters the distribution of latent activity. For $\beta = 10^{-7}$, masking has limited effect, as most dimensions are already active under weak regularization. However, the KL distribution becomes slightly more uniform.

At $\beta = 10^{-6}$, the impact of masking is more pronounced. Compared to the non-masked baseline, KL contributions are distributed across a larger number of dimensions. Instead of a sharp drop after a small subset of dominant coordinates, the KL profile exhibits a broader plateau. This indicates that masking prevents near-collapsed dimensions from diluting the regularization signal and encourages more balanced utilization of latent capacity. The associated $\mathrm{std}(\mu_d)$ values confirm that additional dimensions encode structured variation rather than noise.

For $\beta = 10^{-5}$, masking mitigates the severe collapse observed in the non-masked model. Although regularization pressure remains strong, a larger subset of latent dimensions maintains non-trivial KL values. While some concentration of information persists, the overall distribution is more gradual and less dominated by a few coordinates. This demonstrates that the masking mechanism counteracts collapse even in regularization-dominated regimes.

c) Information concentration and effective capacity.: Across all β regimes, non-masked training tends to concentrate KL divergence in a small leading subset of dimensions as regularization increases, effectively reducing the usable latent dimensionality. In contrast, masked training redistributes information more evenly and preserves a larger number of active dimensions under identical architectural capacity.

These results confirm that posterior collapse in temporal VAEs manifests not only as inactive dimensions but also as strong imbalance in information allocation. The masking intervention increases latent space utilization without altering model architecture, thereby improving effective capacity under fixed latent dimensionality.

C. Trajectory Generation

To illustrate the generative behavior of the model under the balanced regularization regime, we present a qualitative sample of generated trajectories for $\beta = 10^{-6}$ in Figure 7. This value corresponds to the operating point identified earlier, where reconstruction fidelity and KL regularization are of comparable magnitude.

Figure 8 shows trajectories generated by the non-masked VAE at $\beta = 10^{-6}$. Latent vectors are sampled from clustered regions of the aggregated posterior in latent space and decoded into full trajectories using the trained decoder.

The generated samples exhibit smooth lateral motion and structured terminal-area behavior consistent with the training distribution, demonstrating that meaningful trajectory generation remains feasible at the balanced regularization level.

The quality of the generations are not assessed in this study for now and will be left for future work. Olive et al [11] presented evaluation metrics for generated trajectories that will be assessed in future work.

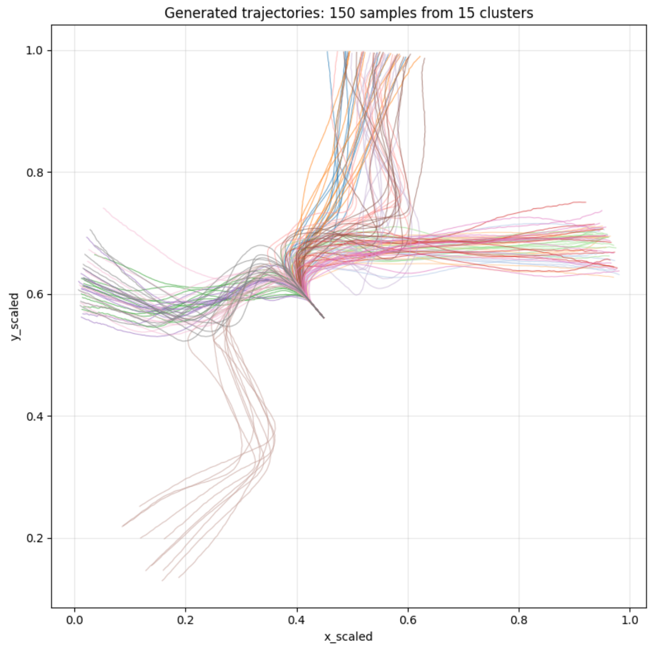


Figure 7. Sample of generated trajectories for $\beta = 10^{-6}$. Latent samples are drawn from clustered regions of the aggregated posterior and decoded into full trajectories.

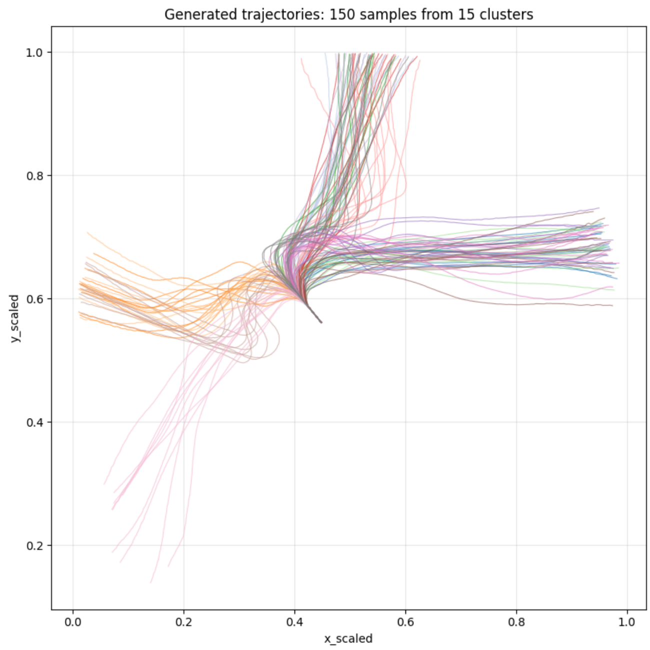


Figure 8. Qualitative generation results for the non-masked VAE with $\beta = 10$ -6. Latent samples are drawn from clustered regions of the aggregated posterior and decoded into full trajectories.

VI. Conclusion & Future Work

This work investigated how to improve interpretability on trajectory generation by improving latent dimension utilization in temporal Variational Autoencoders, with particular emphasis on the role of KL regularization strength and per-dimension information allocation. The study focused on understanding how reconstruction scaling, KL weighting, and loss formulation influence posterior collapse and effective latent capacity in practical trajectory modeling settings.

By formulating the KL divergence as an average over latent dimensions and selecting weighting coefficient based on an explicit reconstruction reference and target per-dimension KL level, we established an interpretable regularization scale. The resulting experiments revealed a consistent pattern: as the weighting coefficient increases, KL divergence becomes increasingly concentrated in a small subset of latent coordinates, leading to partial or severe posterior collapse. To address this issue, we introduced a masked KL objective that suppresses near-collapsed dimensions and concentrates regularization on informative coordinates.

We validated our method on real-world aircraft trajectory data. Empirical results demonstrated that our intervention substantially increases the number of active latent dimensions and yields a more uniform allocation of information, without modifying the network architecture. Importantly, this improved latent utilization is achieved under fixed latent dimensionality and comparable reconstruction scales. Qualitative trajectory generation results further confirm that meaningful and smooth trajectories can be synthesized under controlled regularization, supporting the practical viability of the proposed operating regime.

Overall, the findings highlight that posterior collapse in trajectory VAEs is not merely a binary phenomenon but a gradual redistribution of information across latent dimensions. Explicit monitoring and control of per-dimension KL behavior are therefore essential for achieving interpretable and efficient latent representations in trajectory-based generative models.

We also add that the proposed framework is not inherently tied to the specified airport and runway. Retraining the model on data from different airports, runway layouts, or meteorological conditions is straightforward within the proposed framework, as the regularization design is independent of the specific operational domain. The insights derived here regarding the relationship between KL weighting, posterior collapse, and latent utilization are expected to transfer across settings, providing a principled starting point for adapting the model to new environments. This makes the approach a viable foundation for broader deployment across diverse airports and traffic conditions.

Future work will extend this analysis to adaptive weighting coefficient schedules, alternative prior structures, and larger-scale operational datasets, as well as investigate the implications of latent utilization for downstream decision-support tasks in Trajectory-Based Operations environments.

ACKNOWLEDGEMENT

The authors are grateful to the European Commission for supporting the present work, performed within the VITOLMINS project, funded by the European Union's Horizon Europe research and innovation program under grant agreement no. 101180121 (VITOLMINS). This publication solely reflects the authors' view, and neither the European Union nor the funding agency can be held responsible for the information it contains.

REFERENCES

[1] ICAO Air Traffic Management Requirements and Performance Panel (ATMRPP), "TBO Concept, Draft V. 10.0," International Civil Aviation Organization (ICAO), Working Draft, 2017.

[2] M. C. Rocha Murça and M. de Oliveira, "A Data-Driven Probabilistic Trajectory Model for Predicting and Simulating Terminal Airspace Operations," 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC), San Antonio, TX, USA, 2020, pp. 1-7, doi: 10.1109/DASC50938.2020.9256644.

[3] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in Proc. Int. Conf. Learning Representations (ICLR), 2014.

[4] Diederik PK, Max W (2019), "An Introduction to Variational Autoencoders". Foundations and Trends in Machine Learning, Vol. 12 No. 4 pp. 307–392, doi: https://doi.org/10.1561/2200000056

[5] Timothé Krauth, Adrien Lafage, Jérôme Morio, Xavier Olive, Manuel Waltert, Deep generative modelling of aircraft trajectories in terminal maneuvering areas, Machine Learning with Applications, Volume 11, 2023, 100446, ISSN 2666-8270, https://doi.org/10.1016/j.mlwa.2022.100446.

[6] A. Murad and M. Ruocco, "Synthetic Aircraft Trajectory Generation Using Time-Based VQ-VAE," 2025 Integrated Communications, Navigation and Surveillance Conference (ICNS), Brussels, Belgium, 2025, pp. 1-10, doi: 10.1109/ICNS65417.2025.10976929.

[7] Krauth, Timothé & Figuet, Benoit & Olive, Xavier & Morio, Jérôme. (2023). Collision Risk Assessment in Terminal Manoeuvring Areas based on Trajectory Generation Methods.

[8] Timothé Krauth, Jérôme Morio, Xavier Olive, Benoit Figuet, Advanced collision risk estimation in terminal manoeuvring areas using a disentangled variational autoencoder for uncertainty quantification, Engineering Applications of Artificial Intelligence, Volume 133, Part B, 2024, 108137, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2024.108137.

[9] Tomczak, J. & Welling, M.. (2018). VAE with a VampPrior. Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, in Proceedings of Machine Learning Research 84:1214-1223 Available from https://proceedings.mlr.press/v84/tomczak18a.html.

[10] Krauth, T.; Morio, J.; Olive, X.; Figuet, B.; Monstein, R. Synthetic Aircraft Trajectories Generated with Multivariate Density Models. Eng. Proc. 2021, 13, 7. https://doi.org/10.3390/engproc2021013007

[11] Olive, X., Sun, J., Murça, M. C. R., & Krauth, T. (2021). A Framework to Evaluate Aircraft Trajectory Generation Methods. Paper presented at 14th USA/Europe Air Traffic Management Seminar.

[12] Damien Jacquemart, Jérôme Morio, Conflict probability estimation between aircraft with dynamic importance splitting, Safety Science, Volume 51, Issue 1, 2013, Pages 94-100, ISSN 0925-7535, https://doi.org/10.1016/j.ssci.2012.05.010.

[13] M. Poppe and J. Buxbaum, "Clustering climb profiles for vertical trajectory analysis," in Proceedings of the 10th SESAR Innovation Days, 2020

[14] A. Eckstein, "Data driven modeling for the simulation of converging runway operations," Proceedings of the 4th International Conference on Research in Air Transportation, 2010.

[15] M. Henry, S. Schmitz, N. Revenko, and K. Kelbaugh, "A Monte Carlo simulation for evaluating airborne collision risk in intersecting runways," in Proceedings of the AIAA Modeling and Simulation Technologies Conference, 2013, p. 4598.

[16] G. Jarry, H. Almoctar, D. Delahaye, and C. Hurter, "Interactive Trajectory Modification and Generation with FPCA," in ICRAT 2020, 9th International Conference for Research in Air Transportation, 2020.

[17] G. Jarry, N. Couellan, and D. Delahaye, "On the use of generative adversarial networks for aircraft trajectory generation and atypical approach detection," in Proceedings of the 6th ENRI International Workshop on ATM/CNS, Tokyo, Japan, Oct 2019. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-02267170

[18] Jungwoo Cho and Seongjin Choi. Toward safe integration of uam in terminal airspace: Uam route feasibility assessment using probabilistic aircraft trajectory prediction, 2025. URL https://arxiv.org/abs/2501.16599.

[19] Vos, R. W., Sun, J., & Hoekstra, J. M. (2024). A Transformer-based Trajectory Prediction Model to Support Air Traffic Demand Forecasting. In E. Neiderman, M. Bourgois, D. Lovell, & H. Fricke (Eds.), Proceedings International Conference on Research in Air Transportation Article ICRAT 2024-87

[20] Matthias Schäfer, Martin Strohmeier, Vincent Lenders, Ivan Martinovic and Matthias Wilhelm. "Bringing Up OpenSky: A Large-scale ADS-B Sensor Network for Research". In Proceedings of the 13th IEEE/ACM International Symposium on Information Processing in Sensor Networks (IPSN), pages 83-94, April 2014.