

ReVAR: a data-driven algorithm for generating aero-optic phase screens

JEFFREY W. UTLEY,^{1,*}  GREGERY T. BUZZARD,¹ CHARLES A. BOUMAN,²
AND MATTHEW R. KEMNETZ³ 

¹Department of Mathematics, Purdue University, West Lafayette, Indiana 47907, USA

²Departments of Electrical and Computer Engineering and Biomedical Engineering, Purdue University, West Lafayette, Indiana 47907, USA

³Department of Engineering Physics, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio 45433, USA

*utleyj@purdue.edu

Received 1 April 2026; accepted 5 May 2026; posted 11 May 2026; published 10 June 2026

The propagation of light through a turbulent flow field around an aircraft results in optical distortions commonly known as aero-optic effects. The development of methods to mitigate these effects requires large amounts of realistic aero-optic data. However, methods for obtaining these data, including experiment, computational fluid dynamics, and simple phase screen generation algorithms (e.g., boiling flow), each have significant drawbacks such as high cost, high computation, limited quantity, and/or inaccurate statistics. More recently, data-driven algorithms have been proposed that are computationally efficient and can synthesize aero-optic data to match the statistics of measured data, but these approaches still have drawbacks, including limited quality, inaccurate statistics, and the use of complicated algorithms. In this paper, we introduce ReVAR (Re-whitened Vector AutoRegression), a data-driven algorithm for generating synthetic aero-optic data that match the statistics of measured data. A key contribution in this algorithm is long-range autoregression, a linear predictive model that combines a standard autoregression with a set of low-pass filters of the data to fit both short-range and long-range temporal statistics. ReVAR uses long-range AR together with a spatial re-whitening step to convert measured aero-optic data to temporally and spatially uncorrelated white noise. ReVAR can then generate synthetic aero-optic data by reversing this process using white noise input. Using two measured turbulent boundary layer data sets, we demonstrate that ReVAR better matches the temporal power spectrum and other key metrics of the measured data than do two conventional phase screen generation methods and an existing single time-lag autoregressive model. © 2026 Optica Publishing Group under the terms of the

Optica Open Access Publishing Agreement

<https://doi.org/10.1364/JOSAA.600450>

1. INTRODUCTION

Aerodynamic turbulence around an aircraft can interfere with the accurate transmission or reception of light for purposes of range finding or imaging. In particular, atmospheric density variations near the aircraft lead to refractive index variations, causing optical distortions [1–3] called aero-optic effects [4–6]. These effects are quantified as phase aberrations [1,7,8] and can be measured by optical sensors [9–11] in both wind tunnel experiments [12–14] and flight tests [15,16]. The resulting data, called aero-optic phase screens, are used to develop predictive control methods that augment adaptive-optic (AO) systems to mitigate aero-optic effects [17–19]. Specifically, predictive control methods use aero-optic phase screen data to train algorithms such as dynamic mode decomposition [17,18,20,21], neural networks [22–24], and autoregressive (AR) models [19]. Thus, large data sets of aero-optic phase screens are essential to provide adequate training data for these algorithms.

However, obtaining large data sets of aero-optic phase screens through physical experiments is expensive and time-intensive. First, supersonic wind tunnels can measure only very short time series of aero-optic data [25–27]. In addition, wind tunnels can introduce nonphysical artifacts in the measured data [28], are costly to run, and can simulate only limited aircraft environments [15]. Likewise, generating in-flight data is also expensive, time-intensive, and requires extensive planning and coordination [15].

As an alternative to measured data, a variety of simulation methods have been investigated to generate synthetic aero-optic phase screens. Computational fluid dynamics (CFD) algorithms can simulate aero-optic phase aberrations directly [29–31], but these algorithms face a strict trade-off between physical relevance and computational efficiency [32]. Specifically, high-fidelity CFD is often computationally expensive [1,33,34], while computationally cheaper low-fidelity CFD produces less realistic data [1,33]. In addition, because CFD algorithms require physical parameters to be entered by a user,

it is difficult to match CFD simulations to measured aero-optic phase screens.

Other phase screen generation methods such as frozen flow [35] and boiling flow [32,36,37] can also generate phase screen data using the Kolmogorov theory of turbulence [38] and the Taylor frozen-flow hypothesis [39]. These algorithms are computationally inexpensive and can generate data using few parameters. Further, Utley *et al.* [32] introduced an algorithm to estimate these parameters from measured aero-optic phase screens. However, because aero-optic effects do not follow the Kolmogorov theory of turbulence [40,41], the phase screens generated by these methods do not match the spatial statistics of measured aero-optic phase screens [32,37]. Although Utley *et al.* [32] introduce an anisotropic boiling flow algorithm that generates phase screens that more closely match the spatial correlations of aero-optic data, this model fails to match the temporal power spectrum (TPS) of the measured data.

More recently, Vogel *et al.* [40] and Faghihi *et al.* [42] proposed data-driven algorithms for generating aero-optic phase screens that do not require the Kolmogorov theory of turbulence and can directly estimate parameters from data. Vogel *et al.* [40] propose a single time-lag vectorized AR model that directly fits the spatial statistics of measured aero-optic phase screen data using principal component analysis (PCA). However, this approach is unable to model long-range temporal correlations. Faghihi *et al.* [42] applied a linear state-space model to the top PCA coefficients of measured aero-optic phase screens. This algorithm also does not fully capture long-range temporal correlations, and it relies on a nonlinear estimation algorithm from [43]. Further, it restricts the synthetic phase screens to a lower-dimensional subset of fewer than 100 principal components, which is not consistent with the behavior of physical data.

In this paper, we introduce ReVAR (Re-whitened Vector AutoRegression), a data-driven algorithm for generating synthetic data that matches the statistics of measured aero-optic phase screens. ReVAR is computationally efficient for fitting parameters to data and for generating synthetic data. A key component of ReVAR is a linear predictive model that we call long-range autoregression, which includes the following novel contributions:

- A multiple time-lag AR model to generate synthetic aero-optic phase screen data that matches high-frequency temporal correlations of both the measured data and the streamwise slopes of the measured data.
- Multiple low-pass filters used to augment this AR model in order to match the long-range temporal correlations of the measured data.

Our experiments show that ReVAR has improved error metrics and better matches the TPS of two measured turbulent boundary layer (TBL) data sets [9] relative to isotropic and anisotropic boiling flow [32,36] and to the single time-lag AR model proposed by Vogel *et al.* [40]. Further, ReVAR better matches the structure function of the measured data than boiling flow and is comparable to the single time-lag AR model.

This paper builds on previous work from [32,44]. An open-source Python package implementing the ReVAR algorithm is publicly available [45].

2. ReVAR: ALGORITHM DESIGN

ReVAR takes a time series of measured aero-optic phase screens as input and uses two steps, parameter estimation and data synthesis, to generate synthetic phase screen data. The first step estimates parameters $\hat{\theta}$ of a discrete-time multivariate Gaussian random process whose statistics are time-shift invariant. The second step of ReVAR, data synthesis, uses the parameters $\hat{\theta}$ to generate a sample of this random process. The resulting sample has the same spatial and temporal statistics as the measured aero-optic phase screens. We use the term synthetic aero-optic phase screen data (or synthetic data for short) to describe this generated data.

A. Parameter Estimation from Measured Data

Figure 1 shows the three steps of parameter estimation from measured aero-optic phase screens: pre-processing, long-range autoregression, and re-whitening. Here, we refer to the measured aero-optic phase screens as measured aero-optic data or measured data for short.

- *Pre-processing*: Normalize the measured data by removing its sample mean and standard deviation and then compute a spatial principal component analysis (PCA) of the normalized data. This step outputs the principal coefficients of the measured data.
- *Long-range autoregression*: Fit the temporal statistics of the measured data by using the principal coefficients representation to train a linear predictive model, which we call long-range autoregression. This step outputs the time series of residuals of the long-range AR model, which are temporally uncorrelated but spatially correlated.
- *Re-whitening*: Fit the spatial statistics of the measured data by computing a second spatial PCA from the residuals of the long-range AR model. The output of this step is spatially and temporally uncorrelated white noise.

These steps estimate parameters

$$\hat{\theta} = \left(\underbrace{(\hat{\mu}_X, \hat{\Sigma}_X), (\hat{E}, \hat{\Lambda}, N_c)}_{\text{Pre-Processing}}, \underbrace{(\hat{\alpha}, \hat{A})}_{\text{Long-Range AR}}, \underbrace{(\hat{\mu}_\xi, \hat{\Sigma}_\xi, \hat{U})}_{\text{Re-whitening}} \right) \quad (1)$$

that determine the spatial and temporal statistics of a Gaussian random process.

By taking white noise input and reversing these steps, we can then generate synthetic data with the same statistics as the measured data.

1. Pre-processing

Before fitting the statistics of the measured data, we normalize the spatial distribution of the data and reduce the spatial dimensionality of the data. These steps (i) ensure that the parameters of ReVAR do not depend on any units present in the measured data and (ii) reduce computational expense in the linear predictive model.

We first reshape the measured data images into vectors. We denote the n th frame of the measured data by $X_n^{\text{meas}} \in \mathbb{R}^{N_p}$,

ReVAR Parameter Estimation

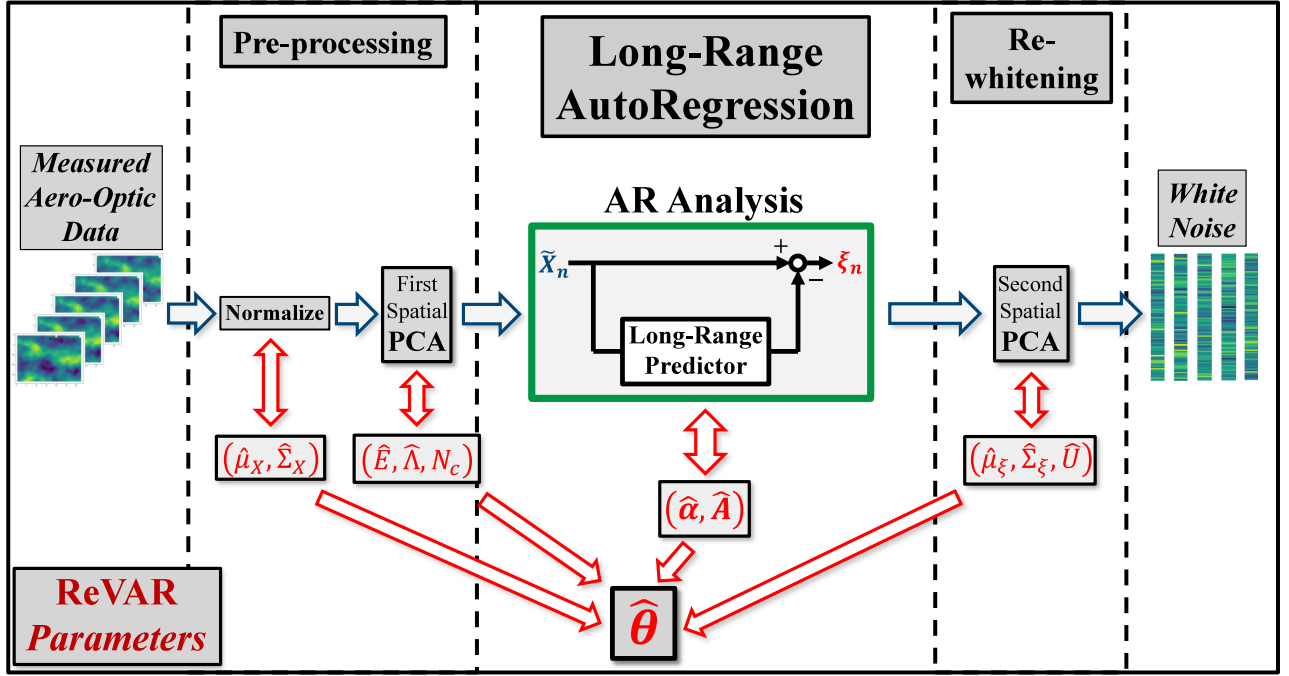


Fig. 1. Parameter estimation from measured aero-optic data. This process estimates the parameters θ of a multivariate Gaussian random process. After normalizing by the time-averaged statistics through Eq. (2) and computing a spatial principal component analysis (PCA) through Eqs. (3)–(5), ReVAR trains a linear predictive model defined through Eq. (9), which we call long-range autoregression (AR); long-range AR uses the long-range predictor of Eq. (7) (illustrated in Fig. 2). Finally, ReVAR takes the residuals of the long-range AR model and computes a second spatial PCA using Eq. (11). Importantly, this converts the residuals to white noise through Eq. (12), so we can reverse this process with white noise input to generate synthetic aero-optic data.

where N_p is the number of pixels in each image. Here, the time-index n takes values from 0 to $N_T - 1$, where N_T is the number of time steps of measured data used for parameter estimation.

Data normalization: To remove units from the measured data and normalize the data's spatial distribution, we normalize by the pixel-wise sample mean and standard deviation of the data, $\hat{\mu}_X, \hat{\sigma}_X \in \mathbb{R}^{N_p}$. After computing these statistics, we take the diagonal matrix $\hat{\Sigma}_X = \text{diag}(\hat{\sigma}_X)$ and compute the normalized n th frame as

$$X_n = \hat{\Sigma}_X^{-1} (X_n^{\text{meas}} - \hat{\mu}_X). \quad (\text{Normalized Data}) \quad (2)$$

Spatial dimensionality reduction: To reduce computational expense in the linear predictive model, we first perform a spatial PCA and restrict to a specified subset of components. To do this, we use the normalized data to compute the spatial covariance matrix $\hat{R}_X$, where for pixels i and j , the (i, j) th entry of $\hat{R}_X$ is $(1/N_T) \sum_n X_{n,i} X_{n,j}$. We then take the singular value decomposition (SVD) of $\hat{R}_X$:

$$\hat{R}_X = \hat{E} \hat{\Lambda} \hat{E}^T, \quad (\text{First Spatial PCA}) \quad (3)$$

where $\hat{E}, \hat{\Lambda} \in \mathbb{R}^{N_p \times N_p}$ and the columns of $\hat{E}$ are the orthonormal *principal components*, with variance given by the corresponding diagonal entry $\hat{\lambda}_i$ of $\hat{\Lambda}$. We then represent the normalized data values using this basis of principal components. This yields the *principal coefficients* [46,47] for the n th frame:

$$\tilde{X}_n = \hat{E}^T X_n. \quad (\text{Principal Coefficients}) \quad (4)$$

We define the prediction subspace as the span of the top N_c principal components that contain 99% of the normalized data's spatial variance and then project the principal coefficients onto the prediction subspace:

$$\tilde{X}_n^{(P)} = P_{N_c} \tilde{X}_n \in \mathbb{R}^{N_c}. \quad (\text{Top } N_c \text{ Principal Coefficients}) \quad (5)$$

Here, $P_{N_c} \in \mathbb{R}^{N_c \times N_p}$ extracts the first N_c components of the vector $\tilde{X}_n$.

This choice of prediction subspace allows the linear predictive model to preserve the spatial statistics of the normalized data while reducing computation time in synthetic data generation.

2. Long-Range Autoregression

In this section, we introduce long-range autoregression, which uses two low-pass filters to capture long-range temporal effects that are not captured by a standard AR model with fixed time lags [48].

Appendix A describes the theoretical foundation of the long-range AR model and compares long-range AR to the standard vectorized AR model and to the autoregressive moving average (ARMA) model. Importantly, long-range AR differs significantly from ARMA in that ARMA applies a moving average to the residual vectors ξ_n , whereas long-range AR instead applies

linear prediction to low-pass filters of the data itself and assumes that the residuals of the model are temporally uncorrelated (for further details, see Appendix A).

Long-range temporal effects: To capture long-range (or, equivalently, low-frequency) temporal effects, we compute two causal low-pass filters of the top N_c principal coefficients $\tilde{X}_n^{(P)}$:

$$Y_{i,n} = (1 - \alpha_i) Y_{i,n-1} + \alpha_i \tilde{X}_n^{(P)}, \quad (6)$$

where $i \in \{1, 2\}$ and $Y_{i,n} = 0$ for $n < 0$. Each filter is a first-order low-pass IIR filter (exponential moving average) with transfer function $H(z) = \alpha_i / (1 - (1 - \alpha_i)z^{-1})$, unit DC gain, and a single pole at $z = 1 - \alpha_i$, yielding an effective cutoff frequency proportional to α_i . We send these low-pass filters as input to a linear predictor, as illustrated in Fig. 2. The values $\alpha = (\alpha_1, \alpha_2)$ determine the cutoff frequencies, and therefore bandwidths, of these low-pass filters. We estimate α by specifying two cutoff frequencies and deriving the corresponding values (α_1, α_2) , as discussed in greater detail in Appendix B.

We use multiple low-pass filters with different cutoff frequencies to increase the low-frequency resolution of long-range AR. Since we represent low-frequency temporal corrections using a basis of low-pass filters of the data, increasing the number of basis elements allows us to increase the frequency resolution of this representation. We found empirically that using two low-pass filters was sufficient for capturing low-frequency temporal effects for the data sets used in this paper, but the method could be applied to any number of filters.

Next-state prediction: We combine the low-pass filters from Eq. (6) with a standard vectorized autoregression to produce a linear prediction of the next state from prior states. The use of low-pass filters introduces long-range temporal correlations in data generated by this linear next-state prediction, so we refer to this predictor as a *long-range predictor*.

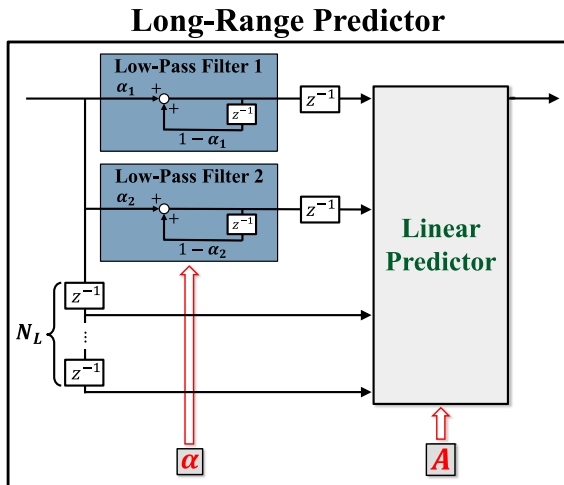


Fig. 2. Long-range predictor corresponding to Eq. (7). This predictor takes in a time history of data and extracts two components: (i) the previous N_L time steps and (ii) two low-pass filters of the entire time series, computed through Eq. (6). The predictor then takes a linear combination of these two components with weights A to estimate the next time step. We use the low-pass filters to introduce long-range (equivalently, low-frequency) temporal correlations in data generated by this predictor without increasing the number of time lags N_L to arbitrarily large values. We use two distinct low-pass filters to increase the low-frequency resolution of this predictor.

Figure 2 illustrates the long-range predictor. Given a value $N_L \geq 1$, called the *number of time lags*, this predictor is given by

$$\hat{X}_n(\mathbf{A}) = \underbrace{\sum_{\ell=1}^{N_L} A_{X,\ell} \tilde{X}_{n-\ell}^{(P)}}_{\text{AR Component}} + \underbrace{A_{Y,1} Y_{1,n-1} + A_{Y,2} Y_{2,n-1}}_{\text{Low-Pass Filter Component}}. \quad (\text{Long-Range Predictor}) \quad (7)$$

Here, $\mathbf{A}$ refers to the set of matrices $A_{X,\ell}$ and $A_{Y,1}, A_{Y,2}$, called the *prediction weights*. The AR component of Eq. (7) captures and generates short-range temporal correlations, with the specific range restricted by the value of N_L . Incorporating a low-pass filter component in this predictor allows us to capture and generate long-range temporal correlations without increasing N_L to arbitrarily large values. The form of Eq. (6) and the one-step time delay of the low-pass filters in Eq. (7) ensure that this long-range predictor is *strictly* causal, so that we can use Eq. (7) to generate synthetic data.

To generate synthetic data using Eq. (7), we first need to estimate the matrices $\mathbf{A}$ from the measured data. For this, we use a least-squares fit from N_L to the end of the time series (N_T):

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\operatorname{argmin}} \left\{ \sum_{n=N_L}^{N_T-1} \left\| \tilde{X}_n^{(P)} - \hat{X}_n(\mathbf{A}) \right\|^2 \right\}. \quad (8)$$

Here, $\hat{X}_n$ is the long-range predictor from Eq. (7) and $\tilde{X}_n^{(P)}$ are the top N_c principal coefficients, which were computed through Eq. (5). Details on our algorithm for solving Eq. (8), including its computational expense, are discussed in Appendix C.

We note that computation of Eq. (7) is of complexity $O(N_c^2(N_L + 2)) = O(N_c^2 N_L)$ when assuming a fixed number of low-pass filters. Hence, by applying linear prediction to the top N_c principal coefficients $\tilde{X}_n^{(P)} \in \mathbb{R}^{N_c}$ instead of the full principal coefficient vectors $\tilde{X}_n \in \mathbb{R}^{N_p}$, we reduce the computational expense of generating data using Eq. (7) by a factor of N_p^2 / N_c^2 . In the data presented below, this is more than a factor of four.

Long-range AR model: We use the long-range predictor from Eq. (7) together with additive noise to generate synthetic data. Since the long-range predictor is restricted to the top N_c components, we first pad with zeros and then add noise. Hence, we model the principal coefficients $\tilde{X}_n$ from Eq. (4) as a *long-range AR process*:

$$\tilde{X}_n = P_{N_c}^T \hat{X}_n(\mathbf{A}) + \xi_n, \quad (\text{Long-Range AR}) \quad (9)$$

where $\hat{X}_n$ is the long-range predictor of Eq. (7) and $P_{N_c}^T \in \mathbb{R}^{N_p \times N_c}$ pads with zeros. In the generation process described by Eq. (9), the vectors ξ_n are sampled from a distribution that must be estimated from the measured data.

During parameter estimation, the ξ_n appear as *residuals* of the long-range AR model. We use the prediction weights $\hat{\mathbf{A}}$, as estimated through Eq. (8), to compute these residuals as

$$\xi_n(\hat{\mathbf{A}}) = \tilde{X}_n - P_{N_c}^T \hat{X}_n(\hat{\mathbf{A}}). \quad (10)$$

Here, we use the notation $\xi_n(\hat{\mathbf{A}})$ to emphasize the dependence of the residuals on the prediction weights $\hat{\mathbf{A}}$.

Given the assumptions of the long-range AR model as in Appendix A and the computation of $\hat{\mathbf{A}}$ using a least-squares fit, the residuals $\xi_n(\hat{\mathbf{A}})$ are temporally uncorrelated [48,49]. However, they may be spatially correlated, which we address in the next section.

3. Re-Whitening

Since the residuals $\xi_n(\hat{\mathbf{A}})$ computed through Eq. (10) may be spatially correlated, we use a second spatial PCA to fully decorrelate these residuals. That is, we take the residuals and compute the sample mean vector $\hat{\mu}_\xi$, spatial covariance matrix $\hat{R}_\xi$, and SVD of $\hat{R}_\xi$:

$$\hat{R}_\xi = \hat{U} \hat{\Sigma}_\xi \hat{U}^T. \quad (\text{Second Spatial PCA}) \quad (11)$$

Importantly, the time series of vectors

$$W_n(\hat{\mathbf{A}}) = \hat{\Sigma}_\xi^{-1/2} \hat{U}^T (\xi_n(\hat{\mathbf{A}}) - \hat{\mu}_\xi) \quad (12)$$

is spatially and temporally uncorrelated *white noise* [47]. For this reason, we call the final step of parameter estimation *re-whitening*.

4. Parameter Estimation Algorithm

The process we have outlined in Section 2.A converts the measured data X_n^{meas} to white noise through Eqs. (2), (4), (7), (10), and (12). In more detail, ReVAR applies the following steps to estimate the parameters θ from Eq. (1):

1. Determine the sample mean and standard deviation vectors (μ_X, σ_X) from data and normalize the data through Eq. (2). This step estimates the normalization parameters $(\hat{\mu}_X, \hat{\Sigma}_X)$ and outputs the normalized data X_n .
2. Take the normalized data X_n , compute a spatial PCA as in Eq. (3), and project the data onto the basis of principal components through Eqs. (4) and (5). This step estimates the first spatial PCA parameters $(\hat{E}, \hat{\Lambda}, N_c)$ and outputs the principal coefficients $\tilde{X}_n$.
3. Take the principal coefficients $\tilde{X}_n$ and estimate both the low-pass filter parameters α using Appendix B and the prediction weights $\mathbf{A}$ through Eq. (8). Then, compute the residuals of the long-range AR model, $\xi_n(\hat{\mathbf{A}})$, through Eq. (10). This step estimates the long-range AR parameters $(\hat{\alpha}, \hat{\mathbf{A}})$ and outputs the residuals $\xi_n(\hat{\mathbf{A}})$.
4. Take the residuals $\xi_n(\hat{\mathbf{A}})$ and compute the second spatial PCA as in Eq. (11). This step estimates the second spatial PCA parameters $(\hat{\mu}_\xi, \hat{\Sigma}_\xi, \hat{U})$ and can be used to convert the residuals to white noise W_n through Eq. (12).

Since this process converts the measured data to white noise, we can then take white noise input and invert Eqs. (2), (4), (7), (10), and (12) using the estimated parameters $\hat{\theta}$ to generate synthetic data.

B. Data Synthesis

Figure 3 illustrates the steps of ReVAR synthesis. This process samples from a Gaussian random process with parameters $\hat{\theta}$ from Eq. (1) by taking white noise input and applying the inverse of the transformation illustrated in Fig. 1. The inverse transformation applies the following steps:

- *Spatial-correlating*: Inverse of re-whitening. Convert the input white noise to spatially correlated noise using the second spatial PCA. The output of this step is a time series of vectors that has the same spatial statistics as the residuals of the linear predictive model applied to the measured data from Eq. (10).
- *Long-range AR synthesis*: Inverse of AR analysis. Convert the spatially correlated noise to temporally correlated data by applying the linear predictive model. The output of this step has the same statistics as the principal coefficients of the measured data from Eq. (4). This step is the inverse of AR analysis from Fig. 1.
- *Post-processing*: Inverse of pre-processing. Convert the temporally correlated data to synthetic data images by inverting the first spatial PCA and denormalizing the data. The output of this process is a synthetic data set with the same spatial and temporal statistics as the measured data.

In more detail, we can generate a time series of synthetic data images with any length N_s by inverting Eqs. (2), (4), (7), (10), and (12). This data generation algorithm applies the following steps:

1. Generate $N_s + N_L$ independent white noise vectors $W_n \in \mathbb{R}^{N_p}$ for $-N_L \leq n < N_s$.
2. For $1 \leq m \leq N_L$, convert the white noise vectors W_{-m} into “initial” principal coefficient vectors

$$\tilde{X}_{-m}^{(P)} = P_{N_c} \hat{\Lambda}^{1/2} W_{-m} \quad (13)$$

that have the same spatial statistics as the top N_c principal coefficients computed from measured data through Eq. (5).

3. Invert the second spatial PCA to convert the white noise W_n to spatially correlated noise ξ_n for $0 \leq n < N_s$ by modifying Eq. (12) into

$$\xi_n = \hat{U} \hat{\Sigma}_\xi^{1/2} W_n + \hat{\mu}_\xi. \quad (14)$$

4. Recursively apply the long-range predictor to compute $\hat{X}_n(\hat{\mathbf{A}})$ for $0 \leq n < N_s$ using Eqs. (6) and (7).
5. Apply long-range AR synthesis to compute *synthetic principal coefficients* $\tilde{X}_n$ for $0 \leq n < N_s$ through Eq. (9).
6. Invert the first spatial PCA and data normalization to convert $\tilde{X}_n$ to synthetic data vectors X_n^{syn} for $0 \leq n < N_s$ by combining Eqs. (4) and (2):

$$X_n^{\text{syn}} = \hat{\Sigma}_X \hat{E} \tilde{X}_n + \hat{\mu}_X. \quad (\text{Synthetic Data}) \quad (15)$$

Reshape the synthetic data vectors X_n^{syn} into single-channel images.

The output of this process is a time series of N_s single-channel images, each with the same size as the measured data images. Note that since N_s is not specified, the synthetic time series can have arbitrary duration.

ReVAR Data Synthesis

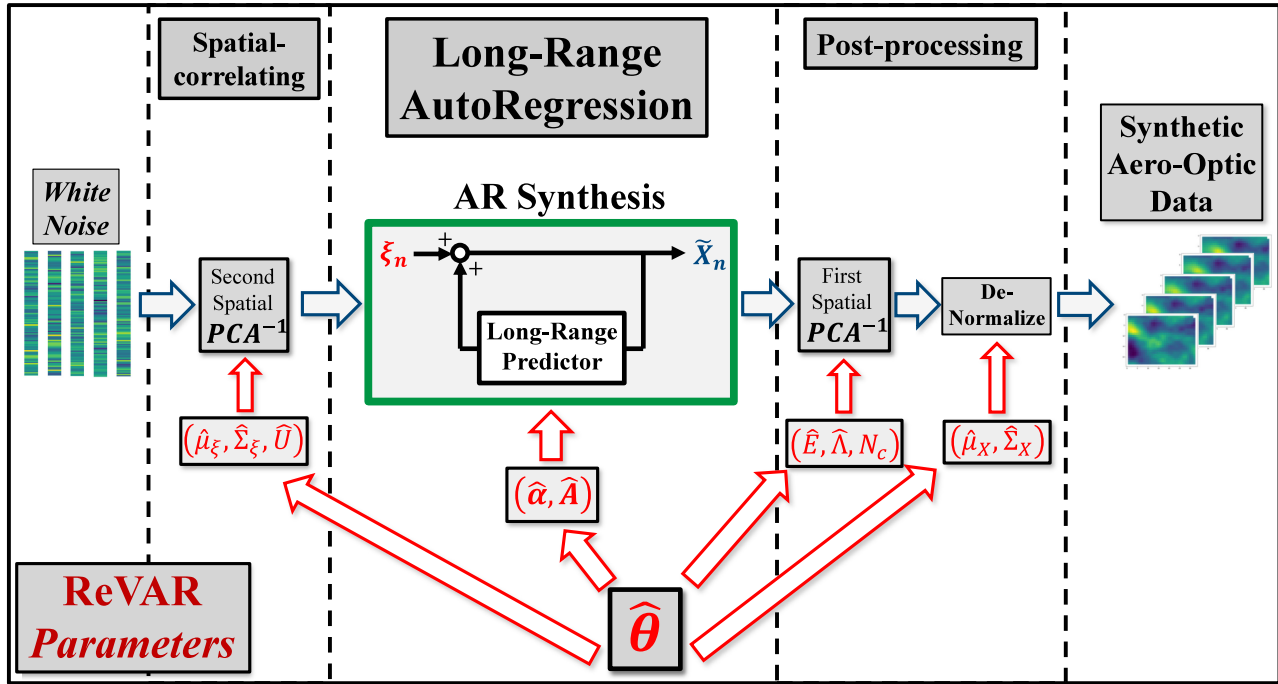


Fig. 3. Diagram of ReVAR synthesis. This procedure samples from a multivariate Gaussian random process with the parameters $\hat{\theta}$ from Eq. (1) by taking white noise input and applying the inverse of the transformation illustrated in Fig. 1 through Eqs. (14), (6), (7), (9), and (15). The synthetic aero-optic phase screen data match the statistics of the measured data.

3. DATA AND METRICS

In this section, we describe the measured data along with the metrics used to evaluate our method.

A. Measured Data

We evaluate our method using empirical optical path difference (OPD) data [50], measured through a wind tunnel experiment investigating a TBL. OPD is related to phase aberrations ϕ via $\text{OPD} = (\lambda/2\pi)\phi$, where λ is the wavelength of the detected light waves [40]. Details on the experimental setup, data measurement method using a Shack–Hartmann wavefront sensor (SHWFS), and post-processing can be found in [9,51] and Section 5.2 of [32].

Table 1 lists details of the OPD images resulting from this experiment. Here, the OPD data have units of microns, and N_c is determined by retaining 99% of the variance from the first spatial PCA.

Table 1. Measured Data Sets F06 and F12

Property	Data Set	
	F06	F12
Number of pixels, N_p	735	380
Number of components, N_c	264	182
Number of time steps, N_T	150,600	251,100
Sampling frequency, f_s (kHz)	100	130

B. Quality Metrics

We evaluate each method using the temporal power spectrum (TPS) applied to both the OPD and the streamwise slopes of the OPD and using metrics applied to the root-mean-square (RMS) amplitude and 2D structure function of the OPD.

The TPS of aero-optic data reveals the temporal frequencies at which the aberrations are most concentrated. This information is essential for training predictive control algorithms, which aid AO correction in specific frequency ranges [52–54]. In fact, some researchers have used the TPS of aero-optic phase aberrations to evaluate the accuracy of predictive control algorithms [17,20,55]. Thus, we compute the TPS of the OPD data (which we call the OPD TPS for short) and denote this quantity by S_{OPD} . In addition to the OPD, the streamwise slopes (i.e., the partial derivative of the OPD images in the x direction) are related to the deflection angle θ_x , which is the quantity most directly measured by the SHWFS [9,56,57]. For this reason, we also compute the TPS of these slopes (which we call the slopes TPS) and denote this quantity by S_{θ_x} . We compute these streamwise slopes using a discrete spatial gradient of the OPD images along the x axis and estimate each TPS using the approach described in Section 5.3 of [32].

In addition to the TPS, the RMS of aero-optic OPD data, often called the OPD_{rms} , describes the overall level of aero-optic aberration [1,6,7], and the structure function D_{OPD} evaluates the spatial statistics of the aero-optic data. We compute OPD_{rms} using an average over both space and time and estimate D_{OPD} using the method described in Section 4 of [32].

We use the normalized root-mean-squared error (NRMSE) to compare the TPS and structure function of the measured data with those of the synthetic data. We use the notation $\text{NRMSE}(\hat{y}, y^{\text{data}})$ for the NRMSE between a quantity $\hat{y}$ obtained from synthetic data and a quantity y^{data} obtained from measured data. Details on our normalization method and a formula for $\text{NRMSE}(\hat{y}, y^{\text{data}})$ can be found in Section 5.3 of [32].

We report the following errors, where we use the same naming conventions as with $(\hat{y}, y^{\text{data}})$:

- *Slopes TPS error*: $\text{NRMSE}(\hat{S}_{\theta_x}, S_{\theta_x}^{\text{data}})$.
- *OPD TPS error*: $\text{NRMSE}(\hat{S}_{\text{OPD}}, S_{\text{OPD}}^{\text{data}})$.
- *OPD_{rms} error*:
The relative error $|\widehat{\text{OPD}}_{\text{rms}} - \text{OPD}_{\text{rms}}^{\text{data}}| / \text{OPD}_{\text{rms}}^{\text{data}}$.
- *Structure function error*: $\text{NRMSE}\left(\sqrt{\hat{D}_{\text{OPD}}}, \sqrt{D_{\text{OPD}}^{\text{data}}}\right)$.

Following the approach of [32], we take the square root of D_{OPD} . We use this scaling so that both the small and the large structure function values influence the error metric proportionally.

4. RESULTS

In this section, we show results from ReVAR applied to measured aero-optic phase screen data. In Section 4.B, we compare ReVAR with other data-driven approaches. In Section 4.C, we evaluate ReVAR with various values of N_L . We evaluate each method using the error metrics of the previous section. In Section 4.D, we describe the computational advantages of using a prediction subspace.

A. Parameter Estimation

We estimate the parameters of ReVAR, isotropic and anisotropic boiling flow [32,36], and the single time-lag AR model from Vogel et al. [40] from each data set F06 and F12 from Table 1. For both measured data sets (described in Section 3.A), we used the first 80% of the time series for estimation and used the remaining 20% for error evaluation. We used the methods from [32] to estimate the parameters of isotropic and anisotropic boiling flow, the method in [40] to estimate the parameters of the single time-lag AR model, and the method described in Section 2.A with $N_L = 4$ to estimate the parameters of ReVAR.

B. Data Generation Results

After estimating the parameters for each data set and each method in Section 4.A, we generated corresponding synthetic data. For both data sets, we used each method to generate 20 data sets, each with 20 times the samples of the 20% error evaluation time series. We computed the errors from Section 3.B using these generated data sets versus the 20% error evaluation data and then averaged over the 20 data sets.

Figure 4 and Table 2 show the TPS and structure function results and list the corresponding error metrics for data set F06. Here, we plot the slopes TPS as a function of frequency f (kHz); plots of the pre-multiplied slopes TPS as a function of Strouhal number can be found in Appendix D. ReVAR closely matches the measured slopes and OPD TPS within

4% NRMSE, while the remaining methods only match the TPS within 7%–26% NRMSE. Most notably, ReVAR closely matches the measured TPS at low frequencies, whereas the remaining methods do not match the D.C. component of the measured data. Correspondingly, ReVAR matches the OPD_{rms} with 0.13% relative error, while the remaining methods have OPD_{rms} relative errors between 1% and 13%. Lastly, Vogel's method and ReVAR both closely match the measured structure function within 3.2% NRMSE. In contrast, both boiling flow methods have structure function NRMSE exceeding 25%. Isotropic boiling flow instead gives an isotropic fit to the measured structure function, while anisotropic boiling flow roughly fits its contours.

Figure 5 and Table 3 show analogous results for data set F12. As with Fig. 4, we plot the slopes TPS as a function of frequency f (kHz); the pre-multiplied slopes TPS as a function of Strouhal number is plotted in Appendix D. Similar to the results for F06, ReVAR fits both TPS more closely than the remaining methods, especially at low frequencies, and fits the structure function comparably to Vogel's method. Here, ReVAR has TPS NRMSE below 2% and OPD_{rms} relative error below 0.1%, while the remaining methods have TPS NRMSE between 5% and 12% and OPD_{rms} relative errors between 0.8% and 4%. Further, while ReVAR and Vogel's method have structure function NRMSE below 3%, both isotropic and anisotropic boiling flow's structure function NRMSE exceed 18%. However, we note that Vogel's method has a slightly lower structure function NRMSE than ReVAR, even after averaging over 20 synthetic data sets. This suggests that Vogel's vectorized AR model implementation, which predicts every vector component [40], may model spatial statistics marginally better than ReVAR's implementation using a prediction subspace.

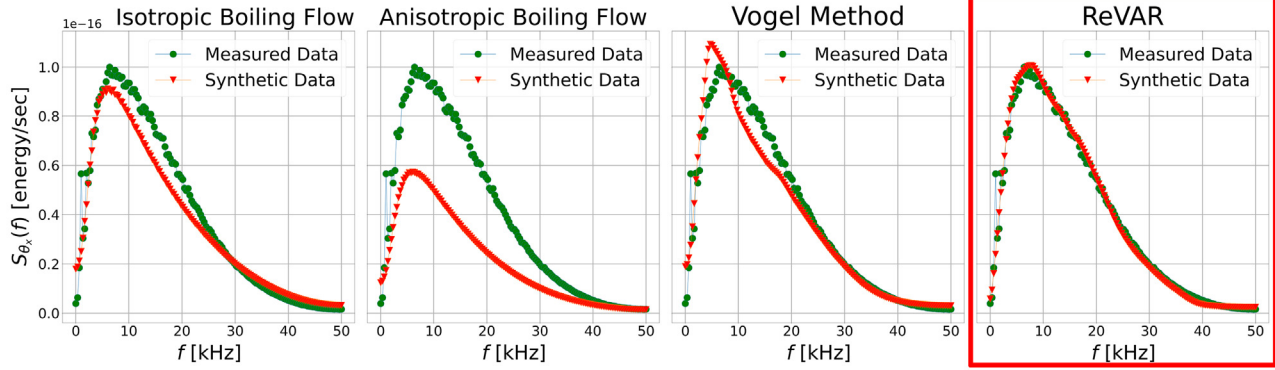
C. ReVAR Results as a Function of Number of Lags, N_L

In this section, we compare TPS NRMSE results from ReVAR with different values of N_L , where N_L is the number of time lags used by the AR component of the long-range predictor in Eq. (7). These results motivate our use of $N_L = 4$ in Section 4.B.

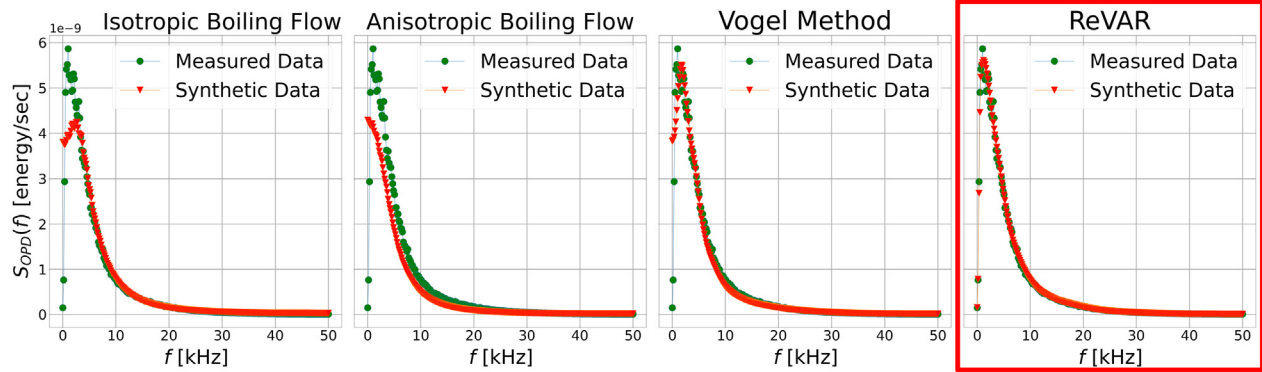
Specifically, we evaluate ReVAR with various N_L values from 1 to 6 on both data sets F06 and F12 from Table 1 using the training and data generation methods described in Sections 4.A and 4.B. Here, we use only the slopes and OPD TPS NRMSE described in Section 3.B.

Figure 6 shows plots of the slopes and OPD TPS NRMSE across different values of N_L . These results show that both TPS NRMSE sharply decrease as N_L increases from 1 to 4, indicating a significant improvement in the TPS fit. However, there is only a marginal improvement in the TPS fit as N_L increases from 4 to 6. For data set F06, the OPD TPS NRMSE increases slightly and the slopes TPS NRMSE remains relatively constant as N_L increases from 4 to 6. Further, for data set F12, the OPD and slopes TPS NRMSE decrease by less than 0.4% and 1% NRMSE, respectively. Given the $O(N_L^3)$ computational expense of training a long-range AR model from Eq. (C10) and the marginal increase in accuracy when increasing N_L beyond 4, we use $N_L = 4$ for both measured data sets.

F06: Slopes Temporal Power Spectrum



F06: OPD Temporal Power Spectrum



F06: Structure Function

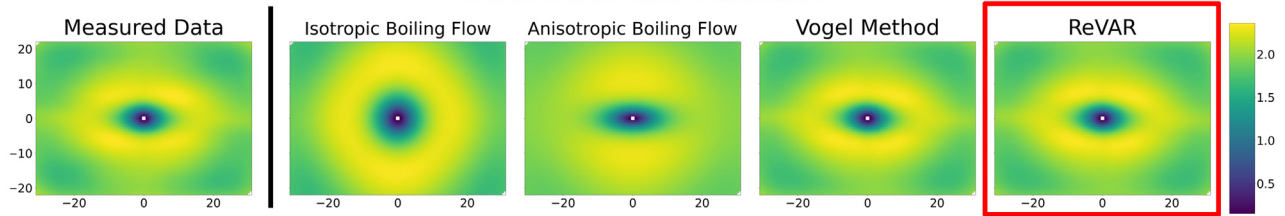


Fig. 4. Results from each method for data set F06. *First and second rows:* Comparisons of the slopes and OPD TPS (respectively) obtained from the measured data (green line with circular markers) and synthetic data (orange line with triangular markers), where OPD has units of microns. Boiling flow and Vogel's method both show a mismatch from the measured slopes and OPD TPS at frequencies below 30 and 5 kHz (respectively), while ReVAR (red box) closely matches both measured TPS at all frequencies. *Third row:* Comparison of the 2D structure functions obtained from measured data (leftmost column) and synthetic data (second-fifth columns). Isotropic boiling flow does not match the anisotropic statistics of the measured structure function, and anisotropic boiling flow only roughly matches its contours, while Vogel's method and ReVAR (red box) both closely match the measured structure function.

Table 2. Error Metrics—Data Set F06

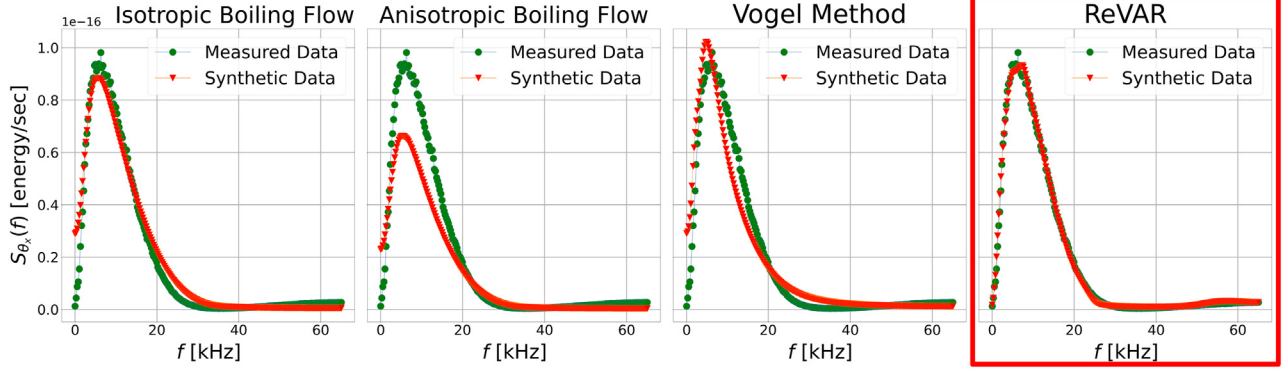
Error	Method			
	Isotropic Boiling Flow	Anisotropic Boiling Flow	Vogel's Method	ReVAR
Slopes TPS NRMSE (%)	8.35	26.86	8.27	3.75
OPD TPS NRMSE (%)	8.79	11.62	7.44	1.77
OPD _{rms} relative error (%)	1.28	13.17	1.36	0.13
Structure function NRMSE (%)	32.60	25.25	3.18	3.18

D. ReVAR Generation Computational Expense Results

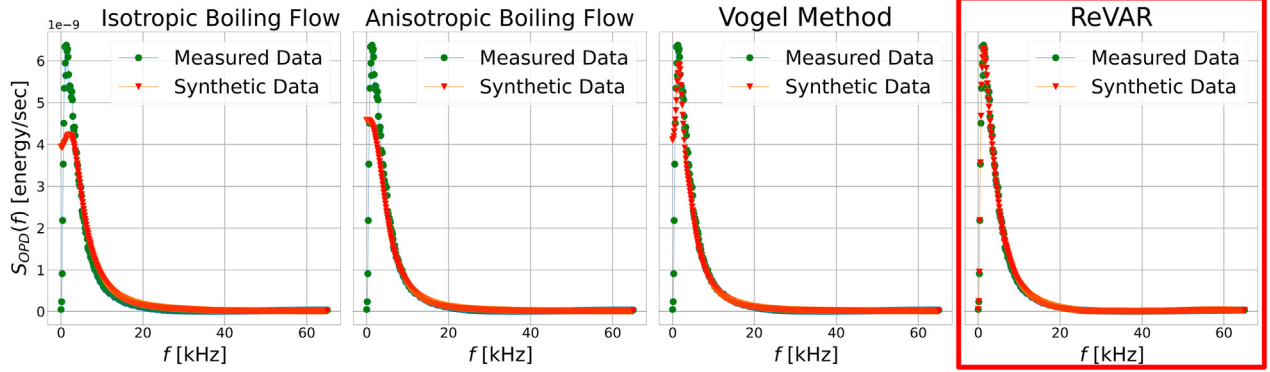
In this section, we discuss the computational advantage of using a reduced subspace for data generation.

Recall that we define N_c as the number of principal components required to capture 99% of the normalized data's spatial variance. For data set F06, the estimated subspace dimension was $N_c = 264$, compared to a full pixel count of $N_p = 735$. For data set F12, $N_c = 182$, compared to $N_p = 380$. Thus, the top N_c principal coefficient vectors $\tilde{X}_n^{(P)}$ (5) have less than half the

F12: Slopes Temporal Power Spectrum



F12: OPD Temporal Power Spectrum



F12: Structure Function

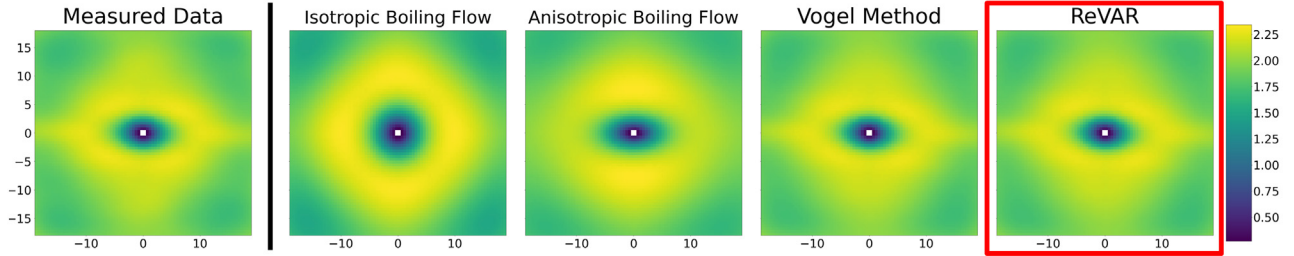


Fig. 5. Results analogous to Fig. 4 but for data set F12.

number of components of the original measured data vectors X_n^{meas} .

Table 3. Error Metrics—Data Set F12

Error	Method			
	Isotropic Boiling Flow	Anisotropic Boiling Flow	Vogel's Method	ReVAR
Slopes TPS NRMSE (%)	5.39	11.88	8.74	1.60
OPD TPS NRMSE (%)	11.64	12.56	9.50	1.70
OPD _{rms} relative error (%)	0.85	4.05	1.00	0.05
Structure function NRMSE (%)	28.65	18.17	2.73	2.90

Projecting the measured data onto this prediction subspace significantly reduces the computational expense of data generation using a long-range AR model. Specifically, the long-range predictor of Eq. (7) operates entirely within the prediction subspace, applying linear prediction to only the top N_c principal coefficients $\tilde{X}_n^{(P)}$. During data generation, evaluating Eq. (7) requires several matrix–vector products that scale quadratically with the vector dimensionality. Applying this predictor to vectors of length N_c rather than N_p reduces the computational expense of this synthesis step by a factor of N_p^2/N_c^2 , which is more than four for both data sets.

This significant reduction in the computational expense of long-range AR synthesis heavily outweighs the additional calculations necessary for using the prediction subspace. Namely, our ability to use this prediction subspace relies on the first spatial PCA (3), which contributes an additional matrix–vector

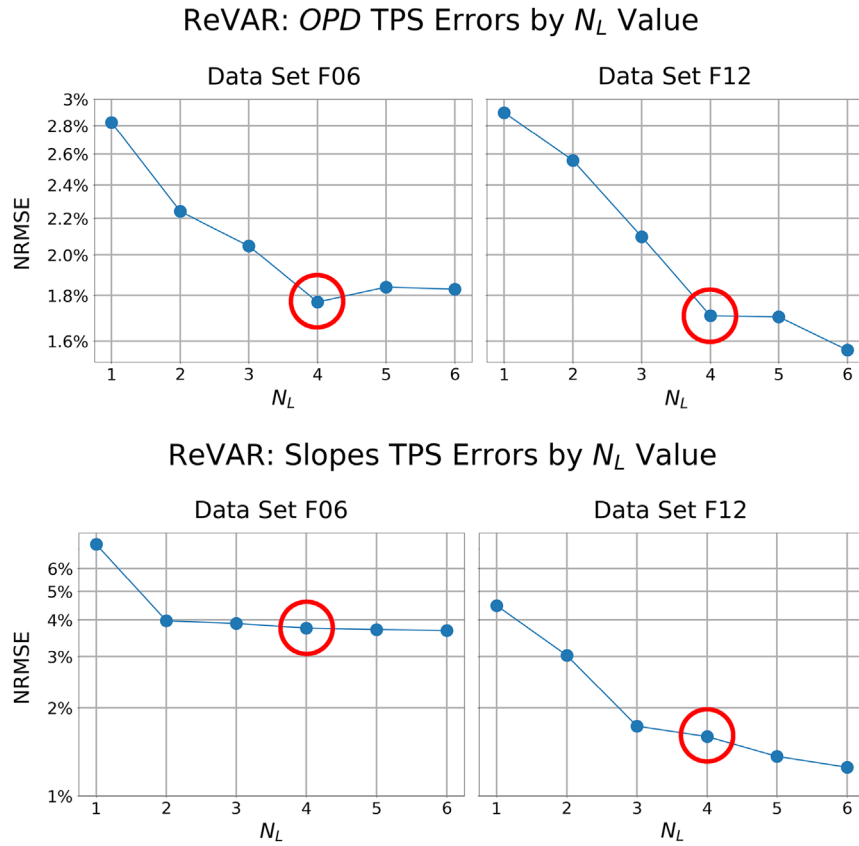


Fig. 6. Accuracy of ReVAR as a function of the number of time lags, N_L . Each plot shows the TPS NRMSE for various numbers of time lags N_L for data set F06 (left) and data set F12 (right). *Top*: OPD TPS NRMSE. *Bottom*: Slopes TPS NRMSE. For both data sets, the NRMSE values decrease sharply from $N_L = 1$ to $N_L = 4$ and then either decrease marginally or increase as N_L increases from 4 to 6. To balance this accuracy against the $O(N_L^3)$ computational expense from (C10), we use $N_L = 4$ for both data sets.

product $\hat{E} \tilde{X}_n$ in Eq. (15). However, since we use $N_L = 4$ time lags and two low-pass filters in Eq. (7), the long-range predictor dominates the computational expense of data generation through several matrix–vector products. Thus, reducing the computational expense of Eq. (7) by more than a factor of four absorbs the extra matrix–vector product in Eq. (15). Accounting for this extra factor, using this prediction subspace reduces the total computational expense of synthetic data generation by more than a factor of two.

5. CONCLUSION

In this paper, we introduced a data-driven algorithm, ReVAR (Re-whitened Vector AutoRegression), for generating synthetic aero-optic phase screen data with statistics that closely match those of measured aero-optic phase screens. ReVAR generates this data computationally and does not encounter the prohibitive monetary or computational expense associated with wind tunnel experiments, flight tests, or computational fluid dynamics simulations. Further, it results in more accurate temporal statistics than existing data-driven approaches by leveraging a long-range autoregressive (AR) model that fits both short-range and long-range temporal correlations within measured data.

We tested ReVAR on two measured turbulent boundary layer data sets F06 and F12 and compared the results with a single time-lag AR model suggested by Vogel et al. [40] and two versions of a conventional phase screen generation method, boiling flow [32,36]. Our experiments demonstrate the following through Tables 2 and 3 and Figs. 4 and 5:

- *Fit of temporal correlations*: ReVAR closely matches the temporal power spectra (TPS) of both the measured data and its streamwise slopes. Across both data sets, ReVAR matched both TPS within 4% NRMSE, while the TPS NRMSE of existing methods ranged from roughly 5% to 27%. Further, ReVAR captures the TPS at low frequencies much more closely than these existing methods (see Figs. 4 and 5).

- *Accuracy of overall variance*: ReVAR matches the true OPD_{rms} of both measured data sets within 0.2% relative error. In contrast, the OPD_{rms} relative errors for existing methods ranged from 0.8% to over 13.2%.

- *Fit of spatial correlations*: For both data sets, ReVAR matches the structure function within roughly 3% NRMSE. These results are comparable to Vogel’s method and superior to both boiling flow models (which yielded NRMSE between 18% and 33%).

Further, the results in Section 4.D indicate that ReVAR’s use of a spatial prediction subspace reduces the computational

expense of synthetic data generation by a factor of two for both measured data sets F06 and F12. Thus, ReVAR closely matches the statistics of the measured data sets considered in this paper while maintaining strict computational efficiency.

APPENDIX A: LONG-RANGE AR: THEORETICAL FOUNDATION

In this section, we outline and distinguish standard AR models, ARMA models, and the long-range AR model, particularly in their ability to capture long-range temporal correlations. For this, we consider a generic discrete-time multivariate Gaussian random process X_n in a general, purely autoregressive form:

$$X_n = \hat{X}_n + \xi_n. \quad (\text{General AR Formulation}) \quad (\text{A1})$$

Here, $\hat{X}_n$ is the linear prediction of X_n based on previous data values and ξ_n is temporally uncorrelated noise, which we think of as the residuals of the prediction $\hat{X}_n$.

Standard AR: A standard AR model takes Eq. (A1) and assumes that $\hat{X}_n$ depends linearly on N_L previous data values; the fixed value of N_L is called the number of time lags [48,49,58]. This results in the standard formulation for the linear prediction in an AR process:

$$\hat{X}_n = \sum_{\ell=1}^{N_L} A_\ell X_{n-\ell}, \quad (\text{Standard AR Predictor}) \quad (\text{A2})$$

where A_ℓ are the matrices containing the weights of this linear prediction. This predictor uses a finite number of prediction weight matrices $A_1, \dots, A_{N_L}$, all independent of the time-index n ; these can be estimated efficiently using a least-squares fit to data.

However, the use of a fixed N_L restricts the temporal correlation range captured by these models. One could increase N_L to capture a specified temporal range, but this increases both the computational expense (see Appendix C) and number of prediction weights, which then increases the variance of the estimators of both the prediction weights A_ℓ and the linear prediction of X_n [48].

ARMA: An ARMA model uses the standard AR predictor of Eq. (A2) for $\hat{X}_n$. However, it relaxes the assumption that the residuals ξ_n of this prediction are temporally uncorrelated and instead introduces a moving average of the past residual vectors ξ_n [48,58]. This results in the formulation

$$X_n = \underbrace{\sum_{\ell=1}^{N_L} A_\ell X_{n-\ell}}_{\text{AR Component}} + \underbrace{\sum_{i=1}^{N_j} B_i \xi_{n-i}}_{\text{Residual MA}} + \xi_n, \quad (\text{ARMA Formulation}) \quad (\text{A3})$$

where N_L is the number of time lags as in the standard AR model of Eq. (A2), and N_j is a finite number of moving averages.

The moving average component introduces a temporal long-range dependence into the generation of X_n but does not directly relate the long-term temporal statistics of the generated data to the next state prediction.

Long-range AR: To couple the distant past to the next state prediction more directly, long-range AR incorporates a moving average on the states rather than on the residuals. Each average serves as a low-pass temporal filter that can be tuned to approximate a specified temporal frequency cutoff. In practice, we use two filters but formulate more generally using N_m , where each has the form

$$Y_{i,n-1} = (1 - \alpha_i) Y_{i,n-2} + \alpha_i X_{n-1}. \quad (\text{Low-Pass Temporal Filter}) \quad (\text{A4})$$

The parameter α_i determines the temporal frequency response of each filter and is estimated as in Appendix B. We include these averages with the previous N_L time lags to obtain the next state prediction:

$$X_n = \underbrace{\sum_{\ell=1}^{N_L} A_\ell X_{n-\ell}}_{\text{AR Component}} + \underbrace{\sum_{i=1}^{N_m} B_i Y_{i,n-1}}_{\text{Low-Pass Filter}} + \xi_n. \quad (\text{Long-Range AR Formulation}) \quad (\text{A5})$$

Expanding $Y_{i,n-1}$ in Eq. (A5) shows that X_n depends on the full range $X_0, \dots, X_{n-1}$ rather than on only the previous N_L states as in standard AR and ARMA. This structure allows long-range AR to model long-range temporal correlations without significantly increasing the computation time of training calculations. With $N_m = 2$, only two additional prediction weight matrices must be added to the least-squares fit. This provides a time and memory efficient use of the entire time history to predict the next state of the process.

APPENDIX B: ESTIMATING THE LOW-PASS FILTER PARAMETERS α

To estimate α in Eq. (6), we specify the cutoff frequencies of the associated low-pass filters $Y_{i,n}$. We set the cutoff frequencies to one and two orders of magnitude below the peak of the TPS of the top N_c principal coefficients $\tilde{X}_n^{(P)}$ and derive the weights $\hat{\alpha}$ that yield these cutoff frequencies.

More precisely, we first find the peak frequency $\hat{f}_c$ of the TPS of the top N_c principal coefficients $\tilde{X}_n^{(P)}$. For this we use the analytical structure of the amplitude spectrum for a TBL, which increases from a value of zero at D.C. to a unique global maximum [59]. We estimate the TPS of each vector component (in units of energy per time step) using the approach of [35,60] by applying a Hamming window on multiple subsets of the time samples and then averaging the resulting TPS estimates over all components and subsets. Importantly, this procedure does

not include the (temporal) sampling frequency of the measured data, so the resulting peak location $\hat{f}_c$ has units of cycles per time step.

Using techniques similar to [61], we find that by taking $\hat{\alpha}_i = 1 - \exp(-2\pi f_i)$, the amplitude of the transfer function for Y_i falls to $1/\sqrt{2}$ at the frequency $\omega = 2\pi f_i$ (from a value of 1 at $\omega = 0$), thus satisfying the commonly used definition of a cutoff frequency for a low-pass filter.

Hence, we make a linear approximation to the exponential to choose the values of α that result in approximate cutoff frequencies $\hat{f}_c/10$ and $\hat{f}_c/100$:

$$\hat{\alpha}_1 = \frac{2\pi \hat{f}_c}{10} \approx 1 - \exp\left(-\frac{2\pi \hat{f}_c}{10}\right), \quad (\text{B1})$$

$$\hat{\alpha}_2 = \frac{\hat{\alpha}_1}{10}. \quad (\text{B2})$$

APPENDIX C: LONG-RANGE PREDICTOR: ESTIMATION OF PREDICTION WEIGHTS

Recall that the long-range predictor of Eq. (7) has the form

$$\hat{X}_n(\mathbf{A}) = \underbrace{\sum_{\ell=1}^{N_L} A_{X,\ell} \tilde{X}_{n-\ell}^{(P)}}_{\text{AR Component}} + \underbrace{A_{Y,1} Y_{1,n-1} + A_{Y,2} Y_{2,n-1}}_{\text{Low-Pass Filter Component}}. \quad (\text{C1})$$

In this section, we describe the least-squares calculations used to estimate the prediction weights $\mathbf{A}$ and report on the computational expense of these calculations.

Least-squares calculations: To compute $\hat{\mathbf{A}}$ through Eq. (8), we estimate the rows of the matrices $A_{X,\ell}$ and $A_{Y,i}$ independently of one another, where $A_{X,\ell}$ and $A_{Y,i}$ are the prediction weight matrices associated with the ℓ th lag and i th low-pass filter, respectively. To do this, we consider a set of N_c least-squares problems, each of size $N_T - N_L$ by $N_c \times (N_L + 2)$.

Here, we iterate over the top N_c principal coefficients, indexed by $0 \leq j < N_c$. We denote the j th rows of the matrices $A_{X,\ell}$ and $A_{Y,i}$ by $[A_{X,\ell}]_j \in \mathbb{R}^{N_c}$ and $[A_{Y,i}]_j \in \mathbb{R}^{N_c}$, respectively. We can estimate the rows of these matrices independently of one another since the principal coefficient vectors $\tilde{X}_n$ are spatially uncorrelated at each time step [46,47]. This estimation method is derived from the closed-form solution of $\hat{\mathbf{A}}$ found in [48].

For each j , we define the following objects:

- First, we define a vector $[\tilde{X}]_j \in \mathbb{R}^{N_T - N_L}$ containing the time series of the j th principal coefficient, as computed from training data:

$$[\tilde{X}]_j = [\tilde{X}_{N_L}]_j [\tilde{X}_{N_L+1}]_j \cdots [\tilde{X}_{N_T-1}]_j^T. \quad (\text{C2})$$

Recall that N_L is the number of time lags and N_T is the number of time steps of training data. By iterating over the range $N_L \leq n < N_T$, we are taking the full data range (excluding the first N_L time steps).

- Next, define a matrix $Z \in \mathbb{R}^{(N_T - N_L) \times (N_c(N_L + 2))}$ containing time series of the top principal coefficients $\tilde{X}_n^{(P)}$ and low-pass filters $Y_{i,n}$:

$$Z = \begin{bmatrix} [\tilde{X}_{N_L-1}^{(P)}]^T \cdots [\tilde{X}_0^{(P)}]^T & [Y_{1,N_L-1}]^T & [Y_{2,N_L-1}]^T \\ [\tilde{X}_{N_L}^{(P)}]^T \cdots [\tilde{X}_1^{(P)}]^T & [Y_{1,N_L}]^T & [Y_{2,N_L}]^T \\ \vdots & \ddots & \vdots \\ [\tilde{X}_{N_T-2}^{(P)}]^T \cdots [\tilde{X}_{N_T-N_L-1}^{(P)}]^T & [Y_{1,N_T-2}]^T & [Y_{2,N_T-2}]^T \end{bmatrix}. \quad (\text{C3})$$

Each row of Z contains previous time steps of data that are multiplied by the prediction weights to estimate the corresponding value in the vector $[\tilde{X}]_j$. Here, we include ranges of $\tilde{X}_n^{(P)}$ and $Y_{i,n}$ containing $N_T - N_L$ time steps. Each column corresponding to $\tilde{X}_n^{(P)}$ contains a distinct, shifted temporal window of the data.

- Last, define a vector $\beta_j \in \mathbb{R}^{N_c(N_L + 2)}$ containing the j th rows of the (unknown) prediction weight matrices $A_{Y,i}$ and $A_{X,\ell}$:

$$\beta_j = [A_{X,1}]_j \cdots [A_{X,N_L}]_j [A_{Y,1}]_j [A_{Y,2}]_j^T. \quad (\text{C4})$$

We solve for these prediction weights β_j .

Since there are many more data points than time lags, we obtain the tall, thin least-squares problem

$$\hat{\beta}_j = \underset{\beta_j}{\operatorname{argmin}} \left\{ \left\| [\tilde{X}]_j - Z\beta_j \right\|^2 \right\} \quad (\text{C5})$$

and solve this using a conventional least-squares algorithm [62,63].

Recall that we associated each vector β_j to the j th rows of the matrices $\hat{A}_{X,\ell}$ and $\hat{A}_{Y,i}$. Thus, after computing the estimates $\hat{\beta}_j$ through Eq. (C5), we use the ordering of the rows $[A_{X,\ell}]_j$ and $[A_{Y,i}]_j$ in Eq. (C4) to place these estimated values into the matrices $\hat{A}_{X,\ell}$ and $\hat{A}_{Y,i}$. This is explicitly given by

$$[\hat{A}_{X,\ell}]_{j,k} = [\hat{\beta}_j]_{(\ell-1) \times N_c + k}, \quad (\text{C6})$$

$$[\hat{A}_{Y,i}]_{j,k} = [\hat{\beta}_j]_{(i-1+N_L) \times N_c + k}. \quad (\text{C7})$$

If we enforce $A_{Y,1} = A_{Y,2} = \mathbf{0}$ (i.e., we use a standard vectorized AR model with N_L time lags), this is equivalent to the method of directly computing $\hat{\mathbf{A}}$ from [48].

Computational expense of training: Assuming that the number of low-pass filters used by the long-range predictor is fixed at two, the values (N_c, N_L) determine the complexity of this computation. Given the complexity of finding each solution $\hat{\beta}_j$ in Eq. (C5), which can be derived from [62,63], the complexity of the full least-squares calculations (for all top principal coefficient indices $0 \leq j < N_c$) is then

$$O(N_T N_c^3 (N_L + 2)^2 + N_c^4 (N_L + 2)^3) \quad (\text{C8})$$

$$= O(N_c^3 N_L^2 (N_T + N_c N_L)). \quad (\text{C9})$$

If the number of training time steps N_T remains fixed, then the complexity of computing $\hat{\mathbf{A}}$ is

$$O(N_c^4 N_L^3). \quad (\text{C10})$$

Applying the long-range predictor of Eq. (7) to only the top N_c principal coefficients instead of all N_p principal coefficients significantly reduces both the computational expense of the least-squares calculations and the number of parameters of this linear predictive model.

APPENDIX D: PRE-MULTIPLIED SLOPES TPS PLOTS AS A FUNCTION OF STROUHAL NUMBER

In this section, we show the slopes TPS results for both measured data sets F06 and F12 in a format more commonly used in the aero-optics field.

Specifically, we plot the pre-multiplied slopes TPS, evaluated as a function of the unit-less Strouhal number St_δ . We use the boundary layer thickness $\delta^* = 15.6$ mm from [9,51] and estimate the convective velocities to be $U_c = 172.37$ m/s for F06 and $U_c = 160.26$ m/s for F12. Further details on the formulation of St_δ and our methodology for computing U_c can be found in Appendix C of [32].

Figure 7 shows the resulting power spectra from both data sets and each method. The horizontal axes display the Strouhal

number St_δ on a logarithmic scale, and the vertical axes display the pre-multiplied TPS values $St_\delta \times S_{\theta_x}(St_\delta)$ on a linear scale. To prevent aliasing artifacts in these plots, we restrict the Strouhal number values in these plots to a maximum of $St_\delta = 3$ and $St_\delta = 2$ for F06 and F12, respectively. These pre-multiplied TPS plots show a similar fit to the measured slopes TPS as the unscaled results in Figs. 4 and 5.

Funding. United States Air Force Research Laboratory Directed Energy Directorate (AFRL/RDKL) (FA9451-20-2-0008); Ralph W. and Grace M. Showalter Research Trust.

Acknowledgment. The authors thank the Showalter Trust and the United States Air Force for supporting this research. The measured data sets used in this article were taken at the University of Notre Dame in the trisonic wind tunnel facility within the Hessert Laboratory for Aerospace Research; for more details concerning the experiment, please refer to Ref. [9].

Disclosures. The views expressed are those of the authors and do not necessarily reflect the official policy or position of the Department of the Air Force, the Department of Defense, or the U.S. Government. Approved for public release; distribution is unlimited. Public Affairs release approval # AFRL-2026-0858. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are publicly available in Ref. [50]. The Python package used to generate the synthetic aero-optic phase screens is publicly available in Ref. [45].

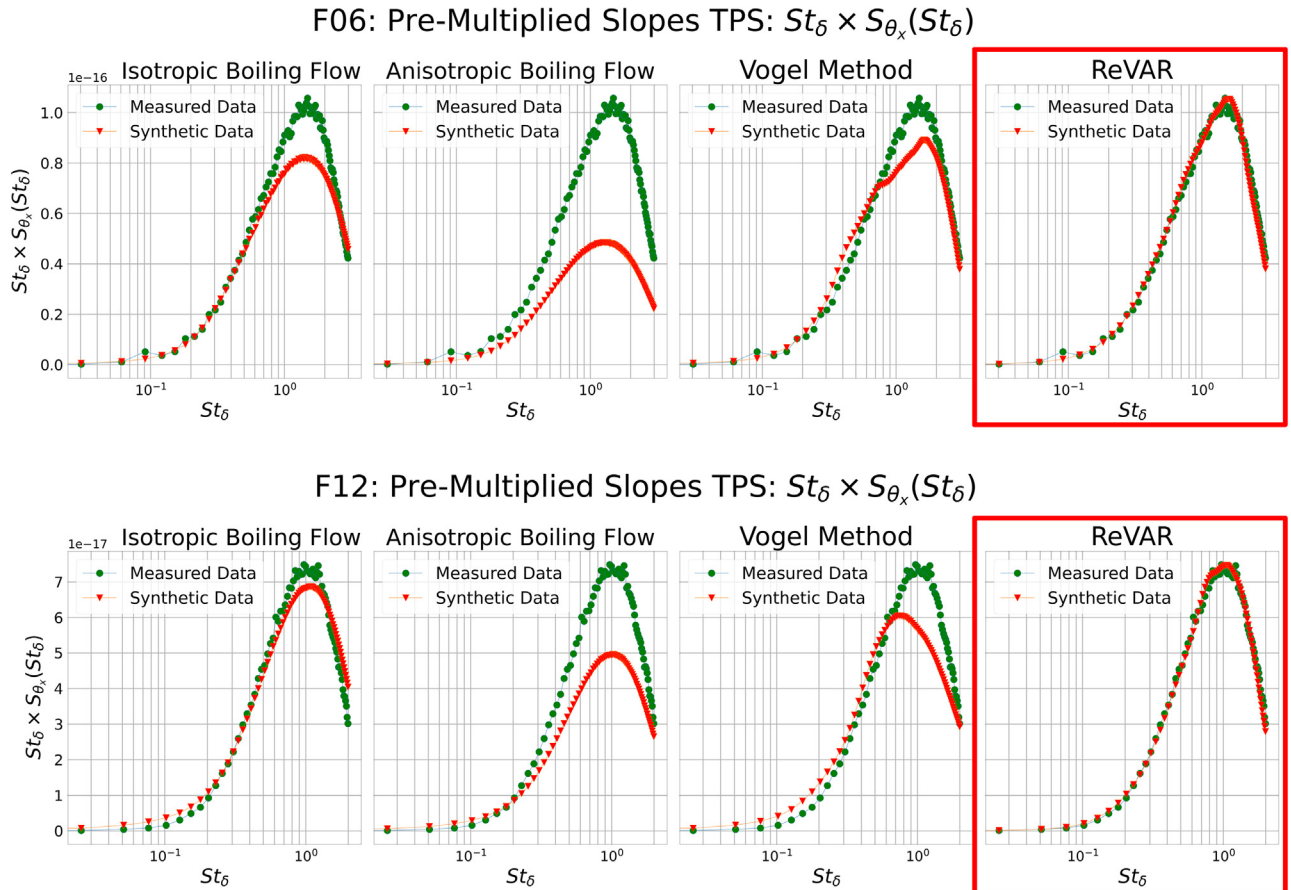


Fig. 7. Comparisons of the pre-multiplied slopes temporal power spectrum (TPS) as a function of Strouhal number St_δ obtained from measured data (green line with circular markers) and synthetic data (orange line with triangular markers) for each method. *Top:* data set F06. *Bottom:* data set F12. These results show consistent accuracy with the standard slopes TPS in Figs. 4 and 5.

REFERENCES

1. M. Wang, A. Mani, and S. Gordeyev, "Physics and computation of aero-optics," *Annu. Rev. Fluid Mech.* **44**, 299–321 (2012).
2. E. J. Jumper and S. Gordeyev, "Physics and measurement of aero-optical effects: past and present," *Annu. Rev. Fluid Mech.* **49**, 419–441 (2017).
3. E. J. Fitzgerald and E. J. Jumper, "The optical distortion mechanism in a nearly incompressible free shear layer," *J. Fluid Mech.* **512**, 153–189 (2004).
4. E. J. Jumper, S. Gordeyev, and M. R. Whiteley, *Aero-Optical Effects* (Wiley, 2023).
5. G. W. Sutton, "Aero-optical foundations and applications," *AIAA J.* **23**, 1525–1537 (1985).
6. E. J. Jumper and E. J. Fitzgerald, "Recent advances in aero-optics," *Prog. Aerosp. Sci.* **37**, 299–339 (2001).
7. M. R. Visbal and D. P. Rizzetta, "Effect of flow excitation on aero-optical aberration," in *46th AIAA Aerospace Sciences Meeting and Exhibit* (American Institute of Aeronautics and Astronautics, 2008), p. 1074.
8. M. Kalensky, S. Gordeyev, M. R. Kemnetz, *et al.*, "Aero-optical effects, part I. System-level considerations: tutorial," *J. Opt. Soc. Am. A* **41**, 2163–2174 (2024).
9. M. R. Kemnetz and S. Gordeyev, "Optical investigation of large-scale boundary-layer structures," in *54th AIAA Aerospace Sciences Meeting* (2016), pp. 1460.
10. J. M. Geary, *Introduction to Wavefront Sensors*, Vol. **TT18** of Tutorial Texts in Optical Engineering (SPIE Optical Engineering, 1995).
11. R. B. Holmes, "Adaptive optics for directed energy: fundamentals and methodology," *AIAA J.* **60**, 5633–5644 (2022).
12. S. Gordeyev, E. Jumper, B. Vukasinovic, *et al.*, "Fluidic control of a turret wake, part II: aero-optical effects," in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition* (American Institute of Aeronautics and Astronautics, 2009).
13. B. Vukasinovic, A. Glezer, S. Gordeyev, *et al.*, "Flow control for aero-optics application," *Exps. Fluids* **54**, 1492 (2013).
14. B. Vukasinovic, A. Glezer, S. Gordeyev, *et al.*, "Hybrid control of a turret wake," *AIAA J.* **49**, 1240–1255 (2011).
15. E. J. Jumper, M. A. Zenk, S. V. Gordeyev, *et al.*, "Airborne aero-optics laboratory," *Opt. Eng.* **52**, 071408 (2013).
16. E. J. Jumper, S. Gordeyev, D. Cavaliere, *et al.*, "Airborne aero-optics laboratory - transonic (AAOL-T)," in *53rd AIAA Aerospace Sciences Meeting* (2015).
17. B. D. Shaffer, A. J. McDaniel, and C. C. Wilcox, "Predictive modeling of wavefront error using dynamic mode decomposition," *Proc. SPIE* **11490**, 114900E (2020).
18. S. Sahba, D. Sashidhar, C. C. Wilcox, *et al.*, "Dynamic mode decomposition for aero-optic wavefront characterization," *Opt. Eng.* **61**, 013105 (2022).
19. R. Burns, S. Gordeyev, E. Jumper, *et al.*, "Estimation of aero-optical wavefronts using optical and non-optical measurements," in *52nd AIAA Aerospace Sciences Meeting - AIAA Science and Technology Forum and Exposition (SciTech)* (2014), p. 0319.
20. B. D. Shaffer, A. J. McDaniel, C. C. Wilcox, *et al.*, "Dynamic mode decomposition based predictive model performance on supersonic and transonic aero-optical wavefront measurements," *Appl. Opt.* **60**, G170–G180 (2021).
21. J. N. Kutz, D. Sashidhar, S. Sahba, *et al.*, "Physics-informed machine-learning for modeling aero-optics," *Proc. SPIE* **11817**, 118170E (2021).
22. B. D. Shaffer, J. R. Vorenberg, C. C. Wilcox, *et al.*, "Neural network forecasting of transonic turbulent flow for adaptive optics control," *Proc. SPIE* **12239**, 122390H (2022).
23. R. Burns, E. Jumper, and S. Gordeyev, "A latency-tolerant architecture for airborne adaptive optic systems," in *53rd AIAA Aerospace Sciences Meeting* (2015), p. 0679.
24. R. Burns, E. Jumper, and S. Gordeyev, "A robust modification of a predictive adaptive-optic control method for aero-optics," in *47th AIAA Plasmadynamics and Lasers Conference* (2016), p. 3529.
25. S. Gordeyev and T. J. Juliano, "Optical characterization of nozzle-wall Mach-6 boundary layers," in *54th AIAA Aerospace Sciences Meeting* (2016).
26. S. Gordeyev and T. J. Juliano, "Optical measurements of transitional events in a Mach-6 laminar boundary layer," in *46th AIAA Fluid Dynamics Conference* (2016).
27. K. T. Berger, K. E. Hollingsworth, S. A. Wright, *et al.*, "NASA Langley aerothermodynamics laboratory: hypersonic testing capabilities," in *53rd AIAA Aerospace Sciences Meeting* (2015).
28. S. Gordeyev, E. Jumper, B. Vukasinovic, *et al.*, "Hybrid flow control of a turret wake, part II: aero-optical effects," in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition* (American Institute of Aeronautics and Astronautics, 2010).
29. K. Wang and M. Wang, "Aero-optics of subsonic turbulent boundary layers," *J. Fluid Mech.* **696**, 122–151 (2012).
30. K. Wang, M. Wang, S. Gordeyev, *et al.*, "Computation of aero-optical distortions over a cylindrical turret with passive flow control," in *41st Plasmadynamics and Lasers Conference* (American Institute of Aeronautics and Astronautics, 2010), p. 4498.
31. C. Porter, M. Rennie, and E. Jumper, "Computation of the aero-optical environment of a helicopter using prescribed-wake methods," *AIAA J.* **53**, 532–541 (2015).
32. J. W. Utley, G. T. Buzzard, C. A. Bouman, *et al.*, "Boiling flow estimation for aero-optic phase screen generation," *arXiv* (2026).
33. S. Gordeyev and E. Jumper, "Fluid dynamics and aero-optical environment around turrets," in *40th AIAA Plasmadynamics and Lasers Conference* (American Institute of Aeronautics and Astronautics, 2009), p. 4224.
34. S. Gordeyev and E. Jumper, "Fluid dynamics and aero-optics of turrets," *Prog. Aerosp. Sci.* **46**, 388–400 (2010).
35. L. Poyneer, M. van Dam, and J.-P. Véran, "Experimental verification of the frozen flow atmospheric turbulence assumption with use of astronomical adaptive optics telemetry," *J. Opt. Soc. Am. A* **26**, 833–846 (2009).
36. S. Srinath, L. A. Poyneer, A. R. Rudy, *et al.*, "Computationally efficient autoregressive method for generating phase screens with frozen flow and turbulence in optical simulations," *Opt. Express* **23**, 33335–33349 (2015).
37. J. W. Utley, G. T. Buzzard, C. A. Bouman, *et al.*, "Boiling flow parameter estimation from boundary layer data," *Proc. SPIE* **13619**, 136190L (2025).
38. J. D. Schmidt, "Propagation through atmospheric turbulence," in *Numerical Simulation of Optical Wave Propagation with Examples in MATLAB* (SPIE, 2010), Chap. 9, Vol. **PM199**, pp. 149–184.
39. G. I. Taylor, "The spectrum of turbulence," *Proc. R. Soc. London A* **164**, 476–490 (1938).
40. C. R. Vogel, G. A. Tyler, and D. J. Wittich, "Spatial-temporal-covariance-based modeling, analysis, and simulation of aero-optics wavefront aberrations," *J. Opt. Soc. Am. A* **31**, 1666–1679 (2014).
41. J. P. Siegenthaler, E. J. Jumper, and S. Gordeyev, "Atmospheric propagation vs. aero-optics," in *46th AIAA Aerospace Sciences Meeting and Exhibit* (American Institute of Aeronautics and Astronautics, 2008), p. 1076.
42. A. Faghihi, J. Tesch, and S. Gibson, "Identified state-space prediction model for aero-optical wavefronts," *Opt. Eng.* **52**, 071419 (2013).
43. N.-N. Chen and J. Gibson, "Subspace system identification using a multichannel lattice filter," in *American Control Conference Proceedings* (IEEE, 2004), Vol. **1**, pp. 855–860.
44. J. W. Utley, G. T. Buzzard, C. A. Bouman, *et al.*, "Data-driven synthetic wavefront generation for boundary layer data," *Proc. SPIE* **13149**, 131490A (2024).
45. J. W. Utley, G. T. Buzzard, C. A. Bouman, *et al.*, "AOModel," GitHub (2026), https://github.com/jeffreyutley/aomodel_public.
46. A. Chatterjee, "An introduction to the proper orthogonal decomposition," *Curr. Sci.* **78**, 808–817 (2000).
47. G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," *Annu. Rev. Fluid Mech.* **25**, 539–575 (1993).
48. H. Lütkepohl, *New Introduction to Multiple Time Series Analysis* (Springer-Verlag, 2005).

49. C. A. Bouman, *Foundations of Computational Imaging: A Model-Based Approach* (Society for Industrial and Applied Mathematics, 2022).
50. M. R. Kemnetz and S. Gordeyev, "Dataset for: Optical investigation of large-scale boundary-layer structures," Purdue University Research Computing Data Depot (2016), <https://www.datadepot.rcac.purdue.edu/bouman/>.
51. M. Kemnetz, "Analysis of the aero-optical component of the jitter using the stitching method," Ph.D. thesis (University of Notre Dame, 2019).
52. L. Poyneer and J. P. Véran, "Predictive wavefront control for adaptive optics with arbitrary control loop delays," *J. Opt. Soc. Am. A* **25**, 1486–1496 (2008).
53. D. T. Gavel and D. Wiberg, "Toward Strehl-optimizing adaptive optics controllers," *Proc. SPIE* **4839**, 890–901 (2003).
54. R. N. Paschall and D. J. Anderson, "Linear quadratic Gaussian control of a deformable mirror adaptive optics system with time-delayed measurements," *Appl. Opt.* **32**, 6347–6358 (1993).
55. D. J. Goorskey, J. Schmidt, and M. R. Whiteley, "Efficacy of predictive wavefront control for compensating aero-optical aberrations," *Opt. Eng.* **52**, 071418 (2013).
56. J. P. Siegenthaler, S. Gordeyev, and E. Jumper, "Shear layers and aperture effects for aero-optics," in *36th AIAA Plasmadynamics and Lasers Conference* (2005), p. 4772.
57. J. Cress, S. Gordeyev, M. Post, *et al.*, "Aero-optical measurements in a turbulent, subsonic boundary layer at different elevation angles," in *39th Plasmadynamics and Lasers Conference* (American Institute of Aeronautics and Astronautics, 2005).
58. M. Pourahmadi, *Foundations of Time Series Analysis and Prediction Theory*, Wiley Series in Probability and Statistics (Wiley, 2001).
59. S. Gordeyev, A. E. Smith, J. A. Cress, *et al.*, "Experimental studies of aero-optical properties of subsonic turbulent boundary layers," *J. Fluid Mech.* **740**, 214–253 (2014).
60. A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, Prentice Hall Signal Processing Series, 2nd ed. (Prentice Hall, 1999).
61. N. B. Jones and J. D. M. Watson, *Digital Signal Processing: Principles, Devices and Applications*, Vol. **42** of IEE Control Engineering Series (Peregrinus on behalf of the Institution and Electrical Engineers, 1990).
62. E. Anderson, Z. Bai, C. Bischof, *et al.*, *LAPACK Users' Guide*, 3rd ed. (Society for Industrial and Applied Mathematics, 1999).
63. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins Studies in the Mathematical Sciences (Johns Hopkins University, 1996).