Enhancing HAVOK: Truncation Bounds, Data Reduction, and Rare Event Prediction

Vladislav Snytko
Department of Advanced Computing Sciences
Maastricht University
Maastricht, The Netherlands
v.snytko@student.maastrichtuniversity.nl

***Abstract*—Data-driven techniques have become indispensable for analyzing nonlinear and chaotic dynamical systems, where traditional models fall short. In this work, we build upon the HAVOK (Hankel Alternative View of Koopman) framework to improve the analysis and prediction of such systems. We derive an upper bound on the truncation rank in HAVOK analysis by leveraging the exponential decay structure of singular values in chaotic systems, filling a gap in the literature where no robust systematic approach to rank selection previously existed. Additionally, we introduce a method to significantly reduce the amount of measurement data required for accurate HAVOK reconstruction, addressing the practical bottleneck of HAVOK's relatively high computational cost and the absence of existing acceleration techniques. Finally, we test the power of singular vectors to predict rare events in chaotic systems, such as lobe switching in the Lorenz attractor, exploiting increases in activity of SVD coordinates as a sign of an impending event. Together, these contributions pave the way for future promising research on data-driven methods in the study of chaotic dynamics.**

I. Introduction

In an era defined by vast data availability and powerful machine learning tools, data-driven approaches to dynamical systems are becoming indispensable for analyzing, predicting, and controlling complex phenomena, especially when traditional models are either oversimplified or unavailable. These methods are proving invaluable across fields as varied as weather forecasting, ecological modeling, economics, neuroscience, and materials science [1].

A dynamical system is typically described by differential (continuous-time) or difference (discrete-time) equations that govern the evolution of its state over time. Within this broad class, nonlinear systems—where the system's evolution is governed by nonlinear equations—are notoriously challenging [3].

Among the most striking phenomena in nonlinear systems is chaos. The term chaos in the context of dynamical systems does not imply randomness, but rather deterministic unpredictability. In a chaotic system, the governing equations are fully deterministic, yet even infinitesimally small differences in initial conditions can lead to wildly divergent outcomes over time — a property known as sensitive dependence on initial conditions. As Lorenz famously observed in his 1963 study of atmospheric convection, this sensitivity limits long-term predictability and gives rise to what is popularly known as the "butterfly effect" [2] [4].

A more formal definition of chaotic systems often comes from [5]:

- Sensitivity to initial conditions: Small perturbations grow exponentially, quantified by positive Lyapunov exponents.
- Topological mixing: Trajectories spread across the attractor, ensuring that any open set eventually overlaps with any other.
- Dense periodic orbits: Every neighborhood in phase space contains periodic points.

Classic examples of chaotic systems include the Lorenz system [4], the Rössler attractor [6], the double pendulum, and certain maps such as the logistic map [7].

Because nonlinear and chaotic systems resist analytical solutions, researchers have turned to empirical, data-driven methods:

- Delay-coordinate embedding [8] allows reconstruction of phase-space trajectories from scalar time series.
- Nonlinear forecasting techniques [9] differentiate chaotic signals from noise and enable short-term forecasting.
- Sparse regression methods, such as SINDy [10], identify parsimonious governing equations directly from data.
- Machine learning frameworks [11] have enabled learning dynamics and correcting unknown model terms while embedding physical knowledge.

A notable spot in the data-driven study of chaotic dynamics has recently been occupied by a model realization technique called Hankel Alternative View of Koopman (HAVOK) analysis, developed in [12]. HAVOK does Singular Value Decomposition (SVD) of a Hankel matrix of a system's trajectory to identify a new coordinate basis advantageous due to the dynamics being approximately linear in these new *SVD coordinates*.

Building on the work in [12], we present several useful results, specifically:

- Bound on Truncation Rank in HAVOK Analysis: We found how the well-known exponential structure in the distribution of singular values of chaotic dynamical systems can be applied to compute a top bound on truncation rank used in HAVOK analysis. There are no existing methods for choosing the truncation rank, and the one mentioned in the original HAVOK paper, the method from [16], does not yield consistent results, although

sometimes it is capable of identifying a rank that produces an accurate reconstruction. Thus, the heuristic we present here is an important contribution.

- Minimization of Dataset Requirements for HAVOK Reconstruction: We present a method to drastically reduce the amount of data required to perform HAVOK analysis retaining the same truncation rank. Keeping in mind that HAVOK analysis may take a prolonged time to run on a large amount of data, establishing a method to speed up the computations is a desirable result.
- Prediction of Rare Events With SVD: we elaborate on an SVD-based approach to predict rare events in chaotic dynamical systems (such as the lobe switching in Lorenz attractor), first presented in [12]. The method seems promising and able to take its own spot in the corpus of approaches to the prediction of nonlinear dynamics, although it certainly requires further investigation to fully evaluate its capabilities.

II. Background

In this section we recall a number of concepts which will be important further in the paper. Specifically, Singular Value Decomposition is at the core of the whole work, being used in all of the results we present. HAVOK analysis is the technique we improve with our results in sections III and IV. The concepts of the Hankel matrix and the time-delay embedding are important for understanding the HAVOK method, so we also explain them here.

A. Singular Value Decomposition

Assume $\mathbf{M}$ is an $m \times n$ real or complex matrix. Then Singular Value Decomposition is a factorization of $\mathbf{M}$ of the form:

$$\mathbf{M} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \tag{1}$$

where $\mathbf{U}$ and $\mathbf{V}$ are square orthogonal matrices and $\mathbf{S}$ is a rectangular diagonal matrix. The values in $\mathbf{S}$ are sorted in decreasing order. Throughout the paper we will be using compact SVD, a variant of SVD which retains only non-zero columns and rows in $\mathbf{S}$, hence also deletes the orphaned columns and rows of the other two matrices. We also assume throught the paper that $m < n$, and thus $\mathbf{S}$ and $\mathbf{U}$ are everywhere square matrices $m \times m$, and $\mathbf{V}$ is an $n \times m$ matrix. The compact SVD factorization is illustrated in Figure 1. The figure pictures the matrices $\mathbf{U}, \mathbf{S}$, and $\mathbf{V}^T$ from left to right correspondingly, highlighting columns of $\mathbf{U}$ in red, singular values in blue, and rows of $\mathbf{V}^T$ in green. This triple–the red column, the blue value, and the green row–also forms a rank-1 SVD mode. The sum of these rank-1 modes equals the original decomposed matrix.

$$\begin{array}{cccc|cccc|cccc}
u_{11} & u_{12} & \cdots & u_{1r} & \sigma_1 & 0 & \cdots & 0 & v_{11} & v_{12} & \cdots & v_{1n} \\
u_{21} & u_{22} & \cdots & u_{2r} & 0 & \sigma_2 & \cdots & 0 & v_{21} & v_{22} & \cdots & v_{2n} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\
u_{m1} & u_{m2} & \cdots & u_{mr} & 0 & 0 & \cdots & \sigma_r & v_{r1} & v_{r2} & \cdots & v_{rn}
\end{array}$$

Fig. 1: Illustration of the SVD $M = USV^\top$. Columns of U (in red) form the basis, diagonal entries of S (in blue) are the singular values, and rows of $V^\top$ (in green) give the coordinates for $U\,S$.

As Figure 1 suggests, in our context it is better to interpret SVD as a factorization producing an orthogonal basis $\mathbf{US}$ for column vectors in $\mathbf{M}$ (assuming $m \leq n$) with coordinates given by columns of $\mathbf{V}^T$.

B. Hankel Matrices

A Hankel matrix is a rectangular matrix where the elements in all antidiagonals are equal. Formally, $\mathbf{H}_{i,j} = h_{i+j-2}$, $\forall (i,j) \in (1,...,m) \times (1,...,n)$, as is visualized in Figure 2.

$$\mathbf{H} = \begin{bmatrix}
h_0 & h_1 & h_2 & \cdots & h_{n-1} \\
h_1 & h_2 & h_3 & \cdots & h_n \\
h_2 & h_3 & h_4 & \cdots & h_{n+1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
h_{m-1} & h_m & h_{m+1} & \cdots & h_{m+n-2}
\end{bmatrix}$$

Fig. 2: Illustration of the structure of a Hankel matrix. Note the constant antidiagonals and the fact that consecutive columns are time-shifted copies of each other.

C. Time-delay Embedding and SVD Coordinates

Time-delay embedding is a well-known approach to augmenting the state of a system by adding state history, leveraged for state-space reconstruction [17], signal denoising [18], and time-series forecasting [19]. Specifically, if $X(t) \in \mathbb{R}$ is the state of the system at time t, then $\hat{X}(t) = [x(t - \tau(d-1)), x(t - \tau(d-2)), \ldots, x(t)]$ is the embedded state with the embedding dimension d and time-shift τ. An important result concerning this technique is Takens' delay embedding theorem [8], which states that under certain conditions the attractor in the delay embedded coordinates $\hat{X}(t)$ is diffeomorphic to the original attractor in $X(t)$. When discretely sampled, the states $X(t)$ can be augmented to obtain a Hankel matrix. Performing SVD on the Hankel matrix will result in the translation of the state vectors into the new basis with rows of V giving SVD coordinates, the attractor on which is still diffeomorphic to the original one.

D. HAVOK Analysis

Hankel Alternative View of Koopman (HAVOK) analysis is a technique for data-driven realization of a chaotic dynamical system. The main feature of the HAVOK analysis is that the model that one computes with this method decomposes chaotic dynamics into linear and nonlinear parts. It is linked to the Koopman operator, because the linear part of the model approximates the operator in finite-dimensional space [12]. We recall how the HAVOK method proceeds in Algorithm 1.

Algorithm 1 HAVOK Algorithm

Require: A time series $x(t)$, sampling time t_s, embedding dimension d, and truncation rank r

1: Collect measurements $x(t)$ at consecutive time steps with the sampling time t_s
2: Form the Hankel matrix $\mathbf{H}$ using embedding dimension d
3: Perform singular value decomposition: $\mathbf{H} = \mathbf{U}\mathbf{S}\mathbf{V}^T$
4: Truncate $\mathbf{U}, \mathbf{S}, \mathbf{V}$ to rank r: obtain $\mathbf{U}_r, \mathbf{S}_r, \mathbf{V}_r$
5: Let $\mathbf{V} := \mathbf{V}_r$; numerically compute time derivatives of rows $v_1, \ldots, v_{r-1}$ to get matrix $\mathbf{V}'$
6: Compute linear dynamics matrix: $\mathbf{A} = (\mathbf{V}')^T\mathbf{V}^\dagger$ where $\mathbf{V}^\dagger$ is the pseudoinverse of $\mathbf{V}$
7: Extract the r-th column of $\mathbf{A}$ and denote it as B
8: Remove the r-th row of $\mathbf{A}$ to obtain the reduced matrix $\mathbf{A}_{\text{red}}$
9: Define $v(t) = (v_1, v_2, \ldots, v_{r-1})^T$
10: **return** The HAVOK model:

$$\frac{d}{dt}v(t) = \mathbf{A}_{\text{red}}v(t) + Bv_r(t) \tag{2}$$

III. Bound on Truncation Rank in HAVOK Analysis

In HAVOK analysis, we are required to truncate matrices obtained with SVD at a truncation rank r. This step is both necessary and desirable. First, as we will see, due to noise, HAVOK do not reconstruct the original dynamics well at every $r \in \{1, \ldots, d\}$, because some singular vectors are too undescriptive (again, often due to noise) of the nonlinear dynamics of the system.

Second, it is sometimes necessary to reduce the amount of data one processes through HAVOK, and SVD provides an optimal way to do it. According to the Eckart–Young theorem, the rank-r truncation of SVD provides the best low-rank approximation of the original matrix.

In this section we present a way to find an upper bound on r in cases when the measurements are primarily contaminated with finite-variance white noise. We start by presenting the structure in the distribution of singular values in chaotic dynamical systems contaminated with white noise, and we continue by showing how we can use this structure to obtain an upper bound on the truncation rank. We limit our discussion to the Lorenz system, but, as will be shown at the end of the section, we tested the results on multiple chaotic systems. Here is the configuration of the Lorenz system:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(\rho - z) - y \\ \frac{dz}{dt} &= xy - \beta z \end{aligned} \tag{3}$$

The system was integrated from $t_0 = 0$ to $t_1 = 50$ with $\sigma = 10$, $\rho = 28$, $\beta = \frac{8}{3}$, and initial conditions $[x(0), y(0), z(0)] = [1, 1, 1]$. The sampling time is $t_s = 0.001$.

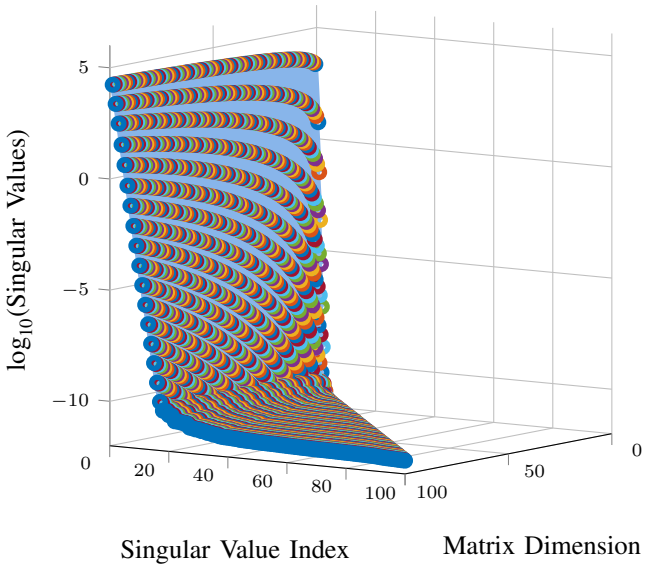


Fig. 3: Log of singular values for different embedding dimensions (number of rows in a Hankel matrix). The distribution starts with exponential decay, which abruptly stops after a certain value on the z-axis.

A. Distribution of Singular Values

The distribution of the singular values of the Lorenz system is shown in Figure 3. The plot demonstrates the logarithm of base 10 of the singular values of Hankel matrices of different dimensions, constructed out of the x coordinate of the Lorenz system integrated with the parameters as stated above. The x-axis on the plot is the embedding dimension, the number of rows of a Hankel matrix, and the y-axis is the index of a singular value in the order they are located in the matrix $\mathbf{S}$. As can be seen, the plot can be divided into two parts: the *exponential* region and the *flat* region. The exponential region corresponds to the singular values being exponentially distributed, a feature that we observed in many chaotic systems. The flat region is the *noise floor* as was mentioned in [13], which was one of the first works studying this structure. Also, following [14], for convenience of the discussion, we call the number of singular values in the linear region as *statistical dimension*. Note that although the statistical dimension increases with the matrix dimension, the threshold that separates the two parts remains almost the same regardless of the matrix dimension ($\sigma_{threshold} \approx 10^{-10.2}$).

The linear region is termed the noise floor because its existence is a direct consequence of the presence of the noise in a signal. It is empirically demonstrated in Figure 4. The plot shows singular value distributions for different standard deviations (σ) of white noise added to the signal (x-coordinate of the Lorenz system) before constructing the Hankel matrix.

The interaction between the level of noise and the statistical dimension can be shown theoretically. First, let's assume that we measure a process $h(t) = y(t) + w(t)$, where $y(t)$ is the true signal we attempt to record and $w(t)$ is the white noise.

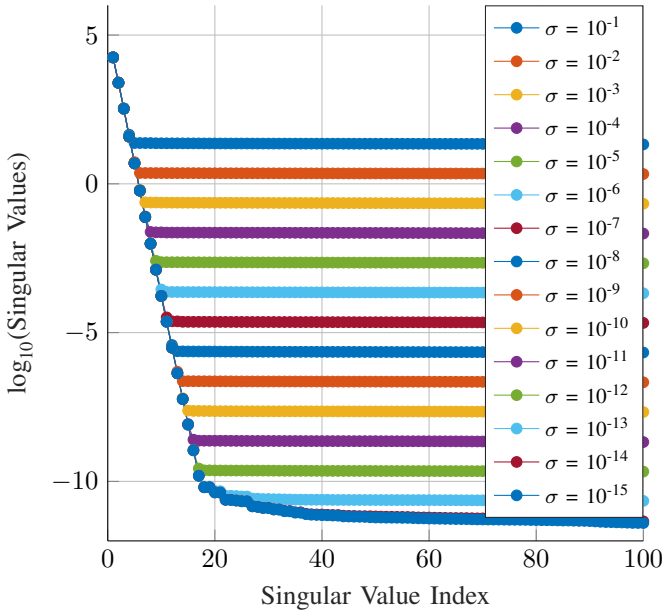


Fig. 4: Singular values for different levels of noise of a Hankel matrix with $d = 100$. The noise moves the linear region up and down, thus justifying the term "noise floor".

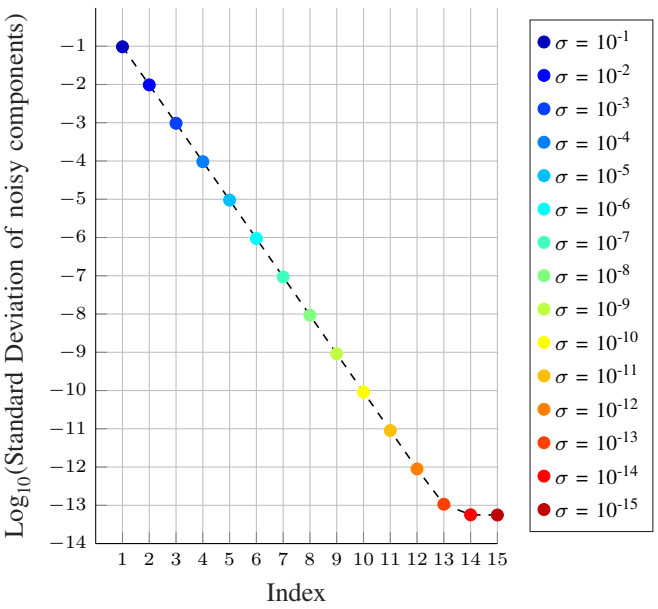


Fig. 5: Logarithm of standard deviations of "noisy" components for different standard deviations of additive Gaussian noise. As can be seen, there is almost a perfect match between these two values from 10^{-1} to 10^{-13}.

After collecting n data points of $h(t)$ we arrange them in an $m \times n - m + 1$ Hankel matrix $\mathbf{H} = \mathbf{Y} + \mathbf{W}$. Recall that the singular values of a matrix $\mathbf{H}$ are the square roots of eigenvalues from the following problem:

$$\mathbf{H}\mathbf{H}^T u_i = \lambda_i u_i \tag{4}$$

u vectors here form the $\mathbf{U}$ matrix in the SVD. Let's expand the matrix $\mathbf{H}$:

$$\begin{aligned} \mathbf{H}\mathbf{H}^T u_i &= \lambda_i u_i \\ (\mathbf{Y}+\mathbf{W})(\mathbf{Y}+\mathbf{W})^T u_i &= \lambda_i u_i \\ (\mathbf{Y}+\mathbf{W})(\mathbf{Y}^T+\mathbf{W}^T) u_i &= \lambda_i u_i \\ (\mathbf{Y}\mathbf{Y}^T + \mathbf{Y}\mathbf{W}^T + \mathbf{W}\mathbf{Y}^T + \mathbf{W}\mathbf{W}^T) u_i &= \lambda_i u_i \end{aligned} \tag{5}$$

Since the white noise is uncorrelated, it follows that $\mathbf{W}\mathbf{W}^T = \mathbf{\Sigma}$, where $\mathbf{\Sigma}_{ij} = \delta_{ij} n\sigma^2$ (δ_{ij} being Kronecker delta here).

$$(\mathbf{Y}\mathbf{Y}^T + \mathbf{Y}\mathbf{W}^T + \mathbf{W}\mathbf{Y}^T + \mathbf{\Sigma}) u_i = \lambda_i u_i \tag{6}$$

Note that $\mathbf{Y}\mathbf{W}^T \approx 0$ and $\mathbf{W}\mathbf{Y}^T \approx 0$, because the expected inner product between the rows x_i and w_i with $i \in \{1, \ldots, m\}$ equals to 0,

$$\begin{aligned} \mathbb{E}(x_i^T w_i) = \mathbb{E}(\sum_{j=1}^{n-m+1} x_{ij} w_{ij}) = \\ \sum_{j=1}^{n-m+1} x_{ij}\mathbb{E}(w_{ij}) = \sum_{j=1}^{n-m+1} x_{ij} 0, \end{aligned} \tag{7}$$

since $w_{ij} \sim \mathcal{N}(0, \sigma^2)$.

Let's return to 6. From 7 it follows that

$$\begin{aligned} (\mathbf{Y}\mathbf{Y}^T + \mathbf{Y}\mathbf{W}^T + \mathbf{W}\mathbf{Y}^T + \mathbf{\Sigma}) u_i &\approx (\mathbf{Y}\mathbf{Y}^T + \mathbf{\Sigma}) u_i \\ (\mathbf{Y}\mathbf{Y}^T + \mathbf{\Sigma}) u_i &\approx \lambda_i u_i \\ \mathbf{Y}\mathbf{Y}^T u_i + n\sigma^2 u_i &\approx \lambda_i u_i \\ \mathbf{Y}\mathbf{Y}^T u_i &\approx (\lambda_i - n\sigma^2) u_i \end{aligned} \tag{8}$$

Thus $S_{ij} \approx \delta_{ij}\sqrt{\tilde{\lambda}_i + n\sigma^2}$, where $\tilde{\lambda}_i = \lambda_i - n\sigma^2$ are the $\sqrt{\tilde{\lambda}_i}$ singular values of the matrix $\mathbf{Y}$ constructed out of the "true" measurements.

From Figure 4 we can induce that in case the noise, either coming from finite-precision arithmetic during integration or from imperfection of measurement tools, is absent from the signal, there is no noise floor. In a continuous setting, this result was proven in [14].

An interesting observation was made that we can infer the standard deviation of the noise from the statistical dimension. Figure 5 shows that the standard deviation of *noisy* SVD components aligns well with the standard deviation of the noise added to the signal. Noisy components are those that correspond to the singular values in the flat region. The discrepancy between the values observed at the bottom of the figure can be explained by the fact that there is an integration noise in the time series, which presumably has a standard deviation around 10^{-14}. The procedure to obtain the standard deviation of the noisy components is given in Algorithm 2.

Algorithm 2 Noise Estimation via SVD and Hankelization

Require: Signal x (time series or structured data)
Ensure: Estimated noise standard deviation σ_{noise}
1: Construct the Hankel matrix $\mathbf{H}$ from the signal x
2: Perform singular value decomposition: $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$
3: Plot $\log_{10}\sigma_i$ versus i and identify the bending point k
4: Reconstruct the noise subspace: $\mathbf{H}_{\text{noise}} = \sum_{i=k+1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$
5: Dehankelize $\mathbf{H}_{\text{noise}}$ to recover noise component $\tilde{x}_{\text{noise}}$
6: Compute the standard deviation: $\sigma_{\text{noise}} = \text{std}(\tilde{x}_{\text{noise}})$
7: **return** Estimated noise level σ_{noise}

The observation in Figure 5 aligns with the theory of Singular Spectrum Analysis (SSA). According to SSA, under certain conditions, when we apply SVD to a Hankel matrix constructed from the sum of two time series, the resulting rank-1 SVD components can be approximately separated into two groups. Each group corresponds to one of the original time series — one representing the original trajectory and the other representing the added time series [15]. The separability, as this property is called in SSA, requires the Hankel matrices of the two time series to span approximately orthogonal column and row spaces, which is essentially the case if the added signal is white noise, as we have shown previously. As presented at the end of the section, the same result holds for several other chaotic dynamical systems.

B. Interrelation between Statistical Dimension and HAVOK Accuracy

In the previous subsection we demonstrated the way to evaluate the influence of noise on the singular values of a chaotic dynamical system. Assuming now we can predict the singular values for different levels of white noise in a signal, and thus predict the statistical dimension, we proceed with showing a simple heuristic with which you can obtain an upper bound on the truncation rank in HAVOK analysis. This is straightforward—the upper bound is the statistical dimension of a system.

Figure 6 pictures errors of attractor reconstruction through HAVOK for different Hankel matrix dimensions d and truncation ranks r. This is the view from above on the Figure 3, but with the points colored according to the error of the reconstruction. The error is calculated as follows:

$$E = \frac{\sum_{i=1}^{n}(x_i - y_i)^2}{\sum_{i=1}^{n} x_i^2}, \tag{9}$$

where x is the original time series and y is the reconstructed one.

The HAVOK reconstruction is computed with the algorithm defined in Algorithm 3.

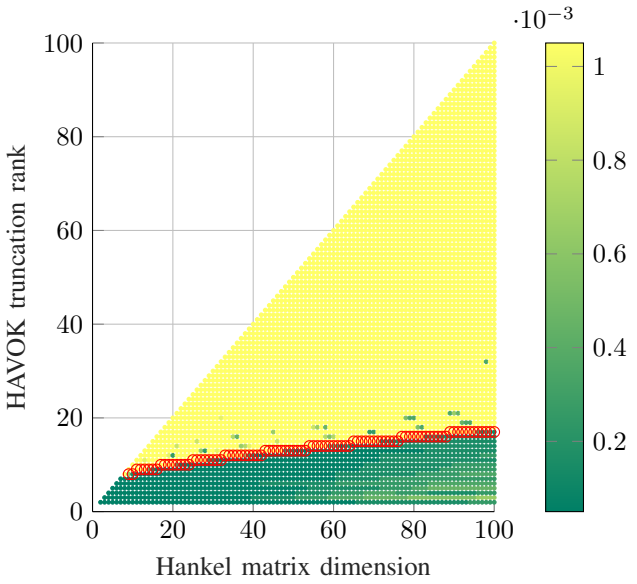


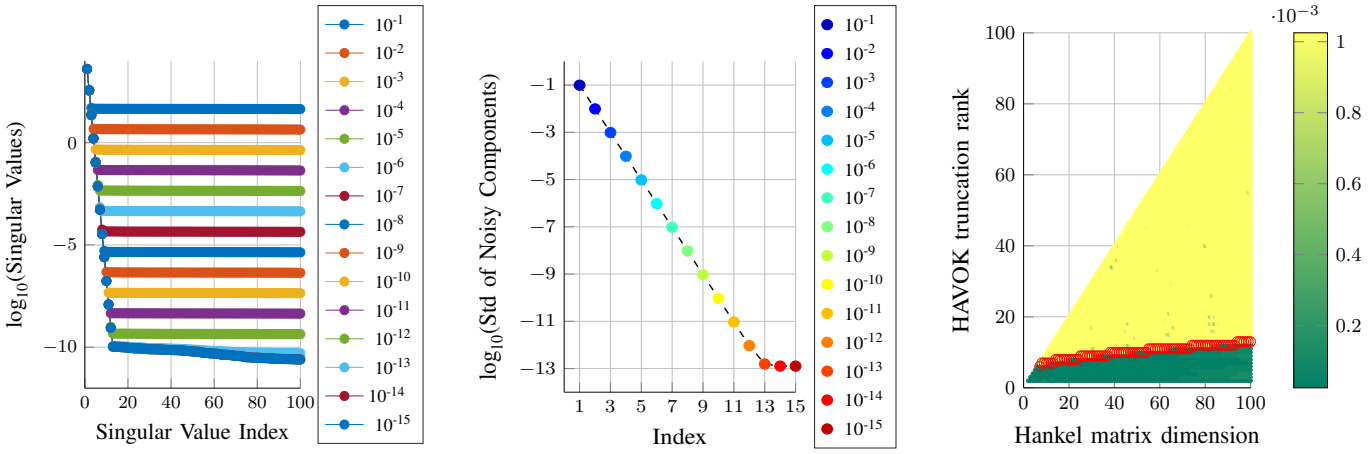
Fig. 6: HAVOK reconstruction errors for different embedding dimensions for the Lorenz system. The red curve highlights singular values after which the noise floor starts. It is clear from the figure that the indices of these singular values are also the largest ones that can be used as truncation ranks.

Algorithm 3 HAVOK Reconstruction

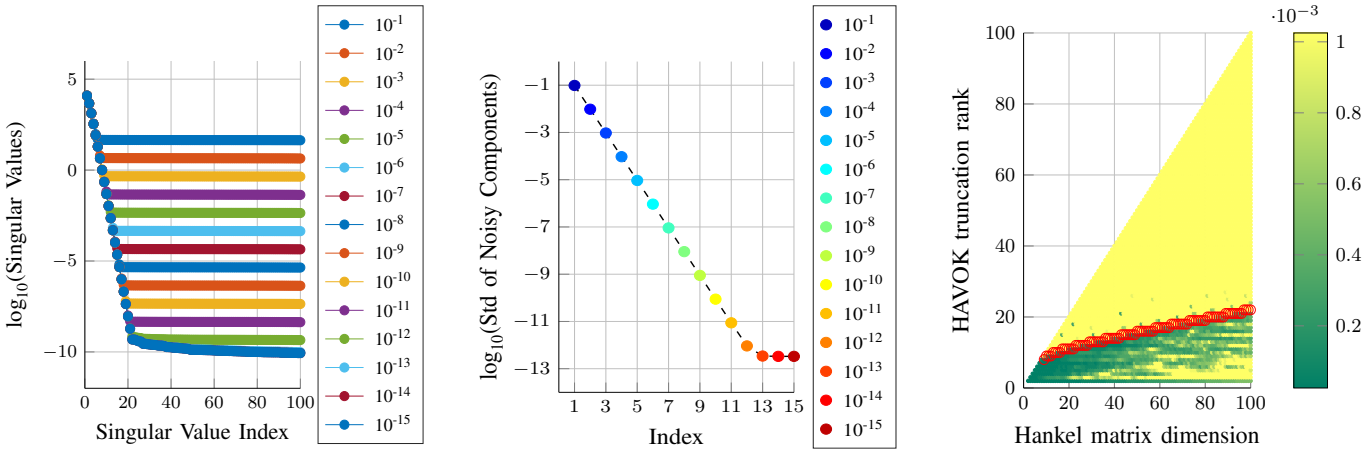
Require: Signal x (time series)
1: Fit HAVOK model using Algorithm 1
2: Integrate the equation 2 you get: obtain states $(\bar{v}_1[t], \bar{v}_2[t], \dots \bar{v}_{r-1}[t])$ for t values from the time span of x
3: Construct a Hankel matrix $\bar{\mathbf{V}}^T$ out of the states you obtained in the previous step
4: Transform $\bar{\mathbf{V}}^T$ back to the original coordinates: $\bar{\mathbf{H}} = \mathbf{U}\mathbf{S}\bar{\mathbf{V}}^T$, where $\mathbf{U}$ and $\mathbf{S}$ come from the HAVOK model
5: Dehankelize $\bar{\mathbf{H}}$ to obtain the reconstructed time series $\bar{x}$
6: **return** The reconstructed time series $\bar{x}$

The red line in 6 marks the statistical dimension of the matrices. The figure clearly shows correspondence between the statistical dimension and the error of HAVOK reconstruction, which can be explained by the fact that, as was mentioned previously, SVD is capable of, at least approximately, separating a contaminated signal into the ”true” signal and the noise. Keeping in mind our previous results, this is likely what happens. The SVD modes corresponding to the singular values larger than the statistical dimension are dominated by noise, and thus are not able to serve as the forcing for the HAVOK model. This gives us a clear heuristic for choosing the truncation rank: the range of acceptable (in terms of the HAVOK reconstruction) truncation ranks is bounded from the above by the statistical dimension of the Hankel matrix.

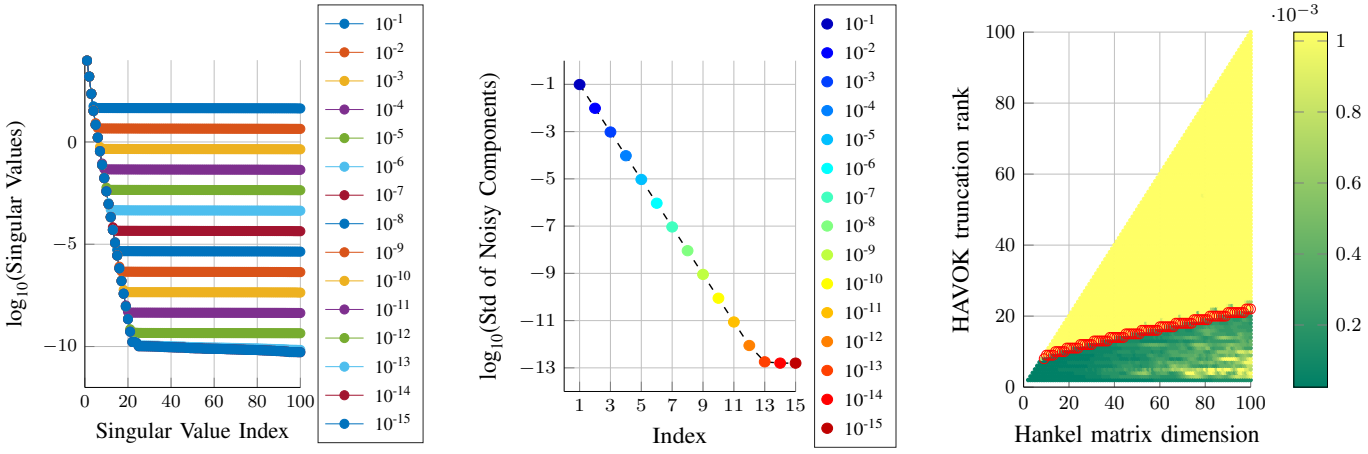
The Figure 7 presents the results from the section for different dynamical systems: Duffing oscillator, Rossler system, and Rikitake system.



(a) Singular values for Duffing oscillator (b) Noise estimation for Duffing oscillator (c) Error pattern for Duffing oscillator



(d) Singular values for Rossler system (e) Noise estimation for Rossler system (f) Error pattern for Rossler system



(g) Singular values for Rikitake system (h) Noise estimation for Rikitake system (i) Error pattern for Rikitake system

Fig. 7: Summary of results for Duffing, Rossler, and Rikitake systems.

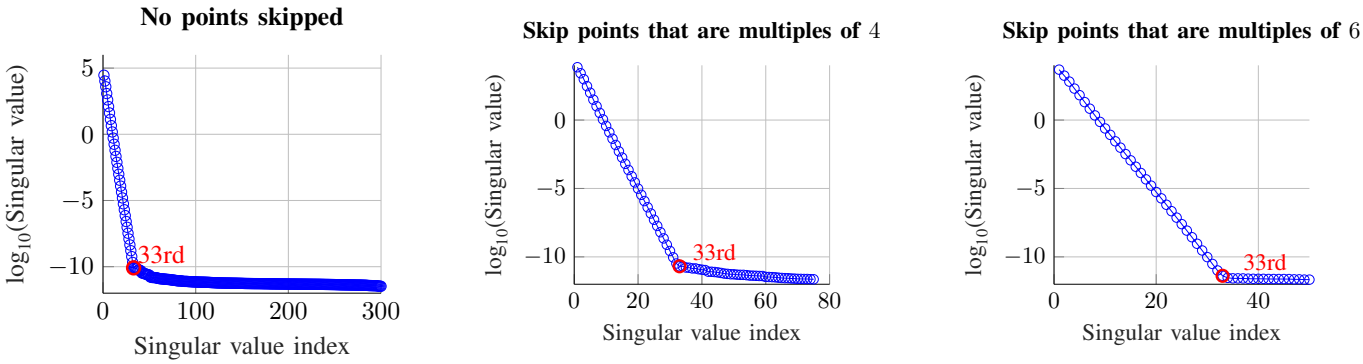


Fig. 8: Singular value distributions for time series with points that are multiples of $1, 4, 6$ skipped. Although we leave out points from the time series, the statistical dimension remains the same, 33.

IV. Reduction of Dataset Requirements for HAVOK Reconstruction

In this section we show a method to reduce the size of the time series for the HAVOK analysis retaining the statistical dimension of the Hankel matrix. The method is simple and borrows the idea from [13].

The approach we present only requires changes to an original time series and thus a Hankel matrix you construct from it. Specifically, before proceeding with the SVD in HAVOK, you first run Algorithm 4, which constructs an "undersampled" Hankel matrix from your original time series in a specific way: it keeps the same time span of columns of the Hankel matrix as it was originally regardless of the undersampling step k which you choose. After running the algorithm, you continue with the usual HAVOK steps.

Algorithm 4 Construction of Undersampled Hankel Matrix for HAVOK

Require: Signal $x = [x_1, \ldots, x_N]$, sampling interval t_s, undersampling step k, window size m

1: Undersample the signal: $\hat{x} = (x[t_0], x[t_0 + kt_s], x[t_0 + 2kt_s], \ldots)$
2: Preserve the total time window: $\tilde{T} \leftarrow m \cdot t_s$
3: Compute number of rows: $\tilde{m} \leftarrow \left\lfloor \frac{\tilde{T}}{kt_s} \right\rfloor$
4: Initialize Hankel matrix $\mathbf{H}_{\text{sub}}$ with dimensions $\tilde{m} \times n$
5: **for** $i = 1$ **to** $\tilde{m}$ **do**
6: **for** $j = 1$ **to** n **do**
7: $\mathbf{H}_{\text{sub}}(i, j) \leftarrow \tilde{x}_{i+j-1}$
8: **end for**
9: **end for**
10: **return** $\mathbf{H}_{\text{sub}}$

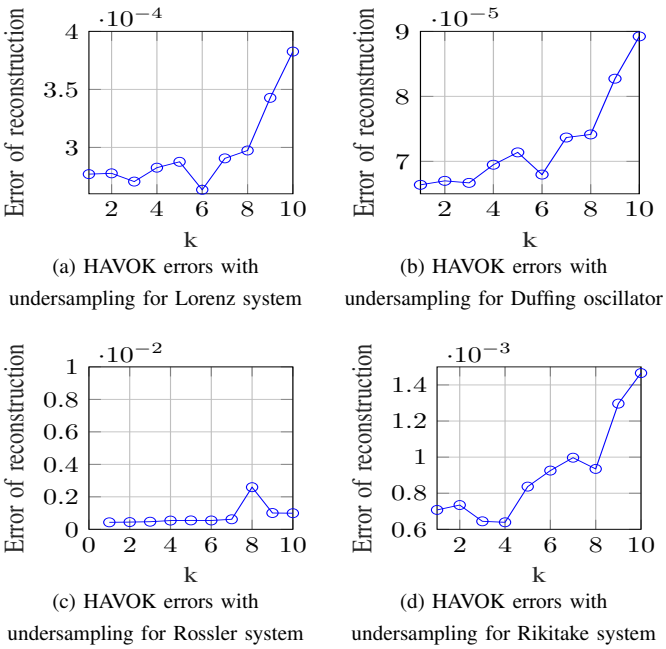
And although after running the algorithm the matrix with which you worked has apparently changed, the statistical dimension remained the same, as illustrated by Figure 8. These plots show singular value distributions for Hankel matrices constructed from the x-coordinate of the Lorenz system using Algorithm 4 with $m = 300$ and $t_s = 0.001$. The k values there are $1, 4$, and 6. As can be seen, the statistical dimension remains the same.

This gives us the method of speeding up HAVOK reconstruction while retaining the same truncation rank (Algorithm 5), assuming the truncation rank you use is smaller than the statistical dimension, as recommended in the previous section.

Algorithm 5 Accelerate HAVOK through Undersampling

Require: Signal x (time series)

1: Obtain the undersampled Hankel matrix $\mathbf{H}_{\text{sub}}$ with 4
2: Follow the Algorithm 1 from step 3 using $\mathbf{H}_{\text{sub}}$



(a) HAVOK errors with undersampling for Lorenz system

(b) HAVOK errors with undersampling for Duffing oscillator

(c) HAVOK errors with undersampling for Rossler system

(d) HAVOK errors with undersampling for Rikitake system

Fig. 9: Summary of HAVOK errors with undersampling for the four systems.

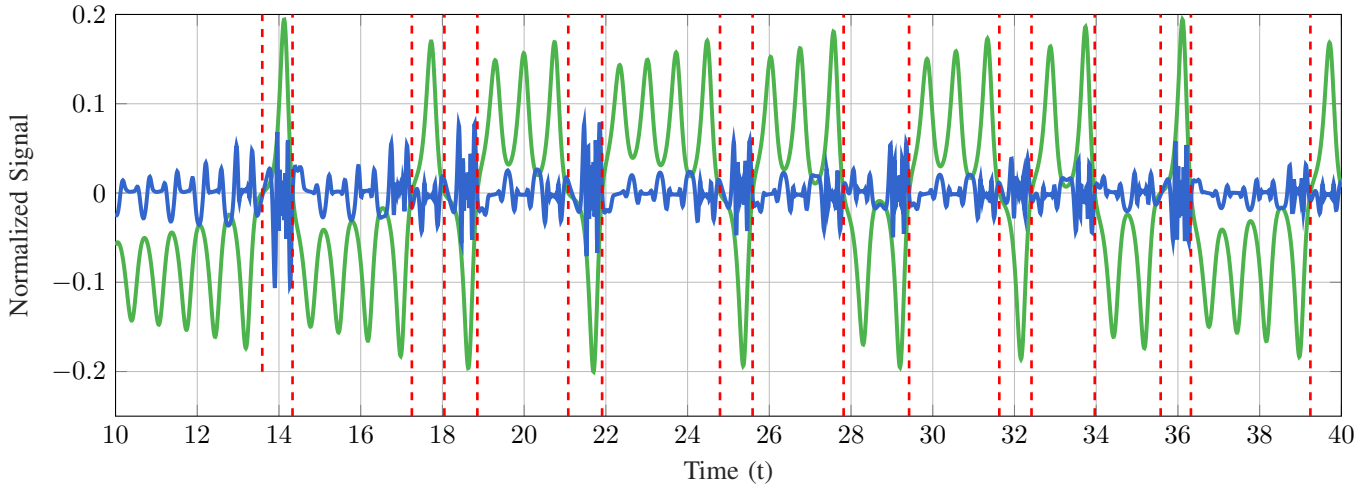


Fig. 10: Illustration of the predictive behavior of singular vectors. A singular vector v_{15} is in blue, the x-coordinate of the Lorenz system is in green, and the red lines mark the times when the lobe switches in the Lorenz system occur. The lobe switches consistently happen after some period of unusual activity in the singular vectors.

Figure 9 shows the errors of HAVOK reconstruction with the undersampling for the Lorenz system, Duffing oscillator, Rossler systems, and Rikitake system. As can be seen, the error almost everywhere stays below 10^{-3}, which we consider to be a threshold for a reconstruction almost without visual discrepancies with an original time series. And this small error persists even for the cases when we left out up to 87.5% of data. Skipping so many of the data points obviously speeds up the computations.

V. Prediction of Rare Events With SVD

In [12], along with the HAVOK model, it was mentioned that the singular vectors are predictive, because, as was observed, singular vectors that correspond to sufficiently high ranks systematically exhibit an increase in activity shortly before a rare event, like lobe-switching in the Lorenz system, starts. We study the prediction power of the singular vectors closely.

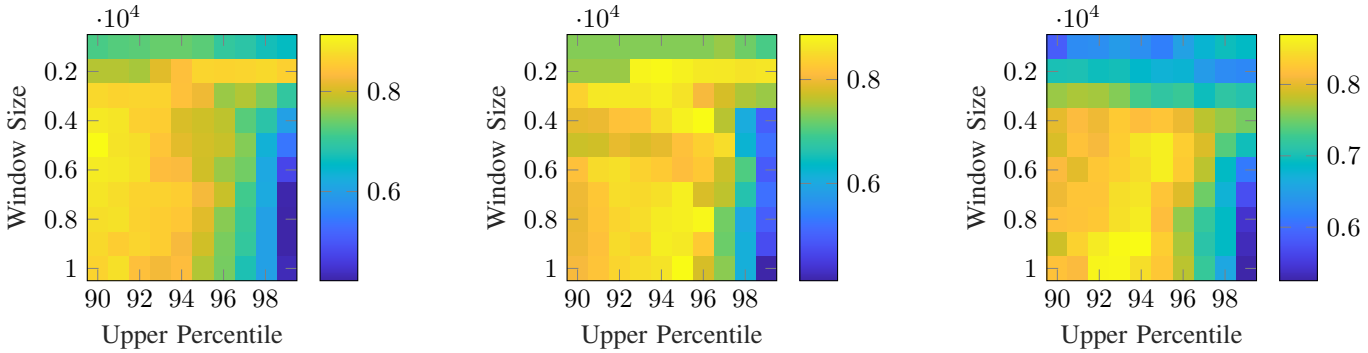
Figure 10 features singular vector v_{15} from $t_0 = 10$ to $t_1 = 40$ of the Lorenz system integrated with the parameters from Section III. The singular vectors, though, were obtained differently. To ensure that the singular vectors are not computed with the knowledge of the whole time series, the SVD computation was done progressively, meaning that we started with a small initial time series and consecutively added more points, each time computing the singular vectors of the Hankel matrix we get and storing the new state vector $\hat{v}(t) = (v_1[t], \ldots, v_{100}[t])$. The red lines on the plot mark lobe switches occurring in the x-coordinate of the system. A lobe switch is defined as a point where the x-coordinate crosses its mean. In the figure we consistently see an increase in activity of the v_{15} vector before these events.

To quantify the predictive activity in singular vectors and study the predictive power, we decided to compute percentiles of v_{15} in a running window of points and mark points where the values are larger or smaller than certain predefined percentiles. Then the closely located marked points are collapsed such that only the first point in a window is left. This is done because once a singular vector crosses one of the percentiles it continues to mark many "false" predictions for some time afterwards. So in this manner we computed the predictions on the Lorenz system from $t_0 = 15$ to $t_1 = 200$ (the time interval from 0 to 15 features the transient phase where no lobe switches yet occur) with the window of 6000 points, upper percentile equals to 95, lower percentile equals to 5, and with the collapse window of 0.5 time units. The results are summarized in Table I. The time lag part provides some statistics on the distribution of the time lags between the prediction and the events. The results in the table are very promising, featuring both high precision and recall values, even though we used a relatively simple metric for quantifying the singular vector activity.

We also decided to evaluate the method on a wider range of parameters to ensure our results are general and to identify optimal values. The optimality we seek is in terms of F1 score, which was chosen as a widespread measure that combines recall and precision values into one value, balancing both.

TABLE I: Prediction Results for Lorenz system

Metric	**Value**
Total events	105
Total triggers	115
Predicted events	96 (Recall = 91.43%)
Successful triggers	96 (Precision = 83.48%)
F1 Score	0.87
Time Lag Statistics	
Mean	0.49 sec
Standard deviation	0.10 sec
Minimum	0.09 sec
Maximum	0.61 sec



(a) Heatmap of F1 scores for Duffing oscillator (b) Heatmap of F1 scores for Rossler system (c) Heatmap of F1 scores for Rikitake system

Fig. 11: Heatmaps of F1 scores for different dynamical systems

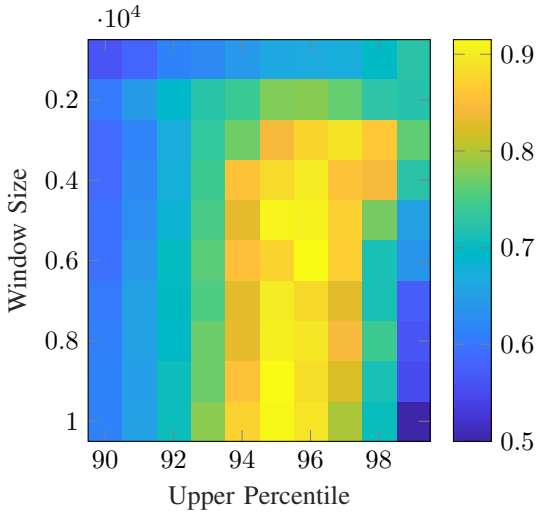


Fig. 12: Heatmap of F1 scores for Lorenz system. F1 scores are color-coded. x-coordinate is the upper percentile, the y-coordinate is the window size. The F1 score in general ranges from 0 to 1, so the values close to 1 here indicate the strong predictive ability of the method.

Since we want to evaluate how the method performs in general, without a close look at either of those two measures, the F1 score is the kind of measure we need. Figure 12 presents the F1 scores for different sizes of the running window and upper percentiles (the lower percentile is calculated by subtracting the upper one from 100). The figure shows that for a window size large enough and the upper percentile around 95, the method produces very strong results.

The Figure 11 presents the results from the section for different dynamical systems: Duffing oscillator, Rossler system, and Rikitake system.

VI. Description of the tested systems

The results in the paper were tested on four chaotic dynamical systems: the Lorenz system, the Duffing oscillator, the Rossler system, Rikitake system. The Lorenz system is already described in section III, here is the description of the other three.

A. Duffing oscillator

$$\begin{aligned} \frac{dx}{dt} &= y \\ \frac{dy}{dt} &= \gamma cos(\omega t) - \delta y - \alpha x - \beta x^3 \end{aligned} \tag{10}$$

The equations were integrated with parameters $\delta = 0.15, \alpha = -1, \beta = 1, \gamma = 0.37, \omega = 1.2$, initial conditions $x_0 = 0.1, y_0 = 0$, on time interval $[0; 1000]$ with sampling time $t_s = 0.005$. The HAVOK errors for undersampled time series were computed for a fixed $r = 10$. In the prediction part, the event which we tried to predict is a crossing of the mean by x-coordinate. The trigger points were merged in the window of 5 time units.

B. Rossler system

$$\begin{aligned} \frac{dx}{dt} &= -y - z \\ \frac{dy}{dt} &= x + ay \\ \frac{dz}{dt} &= b + z(x - c) \end{aligned} \tag{11}$$

The equations were integrated with parameters $a = 0.2, b = 0.2, c = 5.7$, initial conditions $x_0 = 1, y_0 = 1, z_0 = 0$, and on time span $[0; 1000]$ with sampling time $t_s = 0.005$. The HAVOK errors for undersampled time series were computed for a fixed $r = 10$. The events that we aimed to predict are the points where the z-coordinate crosses the threshold value 1. The trigger points were collapsed on the interval of 5 time units.

C. Rikitake system

$$\begin{aligned}\frac{dx}{dt} &= -\beta x + zy \\ \frac{dy}{dt} &= \beta y + (z-\alpha)x \\ \frac{dz}{dt} &= 1 - xy\end{aligned} \tag{12}$$

The equations were integrated with parameters $\alpha = 4, \beta = 0.98$, initial conditions $x_0 = 1, y_0 = 0, z_0 = 0$, and on time span $[0; 1000]$ with sampling time $t_s = 0.005$. The HAVOK errors for undersampled time series were computed for a fixed $r = 18$. In the prediction part, the event that we tried to predict is a crossing of the mean by the x-coordinate. The trigger points were collapsed on the interval of 5 time units.

VII. Summary

To summarize, in the paper we have shown the methods for improving the HAVOK analysis by applying the heuristic to meaningfully choose the truncation rank (section III) and the method to drastically reduce the size of the data meant to be processed through HAVOK (section III).

We also demonstrated very promising lines of future research, specifically the noise estimation method (section III), which can be further developed as a fully separate technique with applications in signal processing and time series analysis, and the rare event prediction technique (section V), which, even for the simple activity quantification method we chose here, shows that it is highly capable of short-term prediction.

The paper also poses an interesting and, at least in this context, important theoretical question about what determines the shape of the singular value distribution. Knowing more about the distribution and its causes will probably allow us to generalize the HAVOK enhancing methods we presented here to chaotic dynamical systems whose distributions might not follow the exponential-flat structure we discussed here.

We conclude the paper with a list of questions that can be further investigated from this point:

- What determines the distribution of singular values and how is it connected to noise in a signal?
- How can we generalize the truncation bound heuristic to the dynamical systems where the distribution of the singular values will not follow the discussed pattern?
- How can we generalize the noise estimation technique to these systems?
- How noise influences the distribution of singular values?
- What will be the influence of colored noise (and many other types of noise) on the distribution and how the methods from the paper can be generalized to account for this?
- What determines whether singular vectors will show the predictive behavior?
- How can we better quantify this behavior to improve the method?
- How this method compares to other techniques for rare event prediction?

References

[1] J. Bramburger, *Data-Driven Methods for Dynamic Systems*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2024.

[2] S. H. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*, 2nd ed. Boca Raton, FL: CRC Press, 2015.

[3] L. Perko, *Differential Equations and Dynamical Systems*. New York: Springer, 2013.

[4] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, pp. 130–141, March 1963.

[5] R. Devaney, *An Introduction to Chaotic Dynamical Systems*, 2nd ed. New York: Avalon Publishing, 1989.

[6] O. E. Rössler, "An equation for continuous chaos," *Phys. Lett. A*, vol. 57, no. 5, pp. 397–398, 1976.

[7] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, 1976.

[8] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence, Warwick 1980*, D. A. Rand and L.-S. Young, Eds. Berlin: Springer, 1981, vol. 898, *Lecture Notes in Mathematics*, pp. 366–381.

[9] G. Sugihara and R. M. May, "Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series," *Nature*, vol. 344, no. 6268, pp. 734–741, April 1990.

[10] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 113, no. 15, pp. 3932–3937, April 2016.

[11] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "Integrating scientific knowledge with machine learning for engineering and environmental systems," *ACM Comput. Surv.*, vol. 55, no. 4, article no. 66, 37 pages, November 2022.

[12] S. L. Brunton, B. W. Brunton, J. L. Proctor, E. Kaiser, and J. N. Kutz, "Chaos as an intermittently forced linear system," *Nature Commun.*, vol. 8, no. 1, p. 19, May 2017.

[13] D. S. Broomhead and G. P. King, "Extracting qualitative dynamics from experimental data," *Physica D*, vol. 20, pp. 217–236, June 1986.

[14] R. Vautard and M. Ghil, "Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series," *Physica D*, vol. 35, pp. 395–424, May 1989.

[15] N. Golyandina, "Particularities and commonalities of singular spectrum analysis as a method of time series analysis and signal processing," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 12, January 2020.

[16] M. Gavish and D. L. Donoho, "The optimal hard threshold for singular values is $4/\sqrt{3}$," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 5040–5053, August 2014.

[17] E. R. Deyle and G. Sugihara, "Generalized theorems for nonlinear state space reconstruction," *PLOS ONE*, vol. 6, no. 3, pp. 1–8, March 2011.

[18] D. Napoletani, D. Struppa, T. Sauer, C. Berenstein, and D. Walnut, "Delay-coordinates embeddings as a data mining tool for denoising speech signals," *Chaos*, vol. 16, p. 043116, January 2007.

[19] H. Peng, P. Chen, R. Liu, and L. Chen, "Spatiotemporal information conversion machine for time-series forecasting," *Fundam. Res.*, vol. 4, no. 6, pp. 1674–1687, 2024.