UNIVERSITY OF CALIFORNIA SAN DIEGO

Improvements in Gappy SPOD for PIV data

A Thesis submitted in partial satisfaction of the
requirements for the degree Master of Science

in

Engineering Sciences (Aerospace Engineering)

by

Ethan Brothers

Committee in charge:

Professor Oliver T. Schmidt, Chair
Professor Nicholas Boechler
Professor Boris Kramer

2024

The Thesis of Ethan Brothers is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

To those who are next.

EPIGRAPH

... so that his place shall never be with those cold and timid souls who neither know victory nor defeat.

Theodore Roosevelt, Citizenship in a Republic

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First, I would like to thank Professor Oliver T. Schmidt for giving me this opportunity; he has truly been a pleasure to work with. His support and guidance have allowed me to improve tremendously as an engineer. I am forever grateful.

Thank you to everyone in the lab during my time here. Akhil, Tianyi, Edward, Brandon, and Cong were all wonderful people to work with. I couldn't have asked for a better group.

A special thank you to Professor Boechler for hiring me, it was a pleasure to be your teaching assistant.

Finally, I would like to thank all my friends and family who supported me along the way. A thank you to Juan for encouraging me to pursue a graduate education. To Mabel for always being there for me. And lastly, to my parents who never stopped believing in me.

ABSTRACT OF THE THESIS

Improvements in Gappy SPOD for PIV data

by

Ethan Brothers

Master of Science in Engineering Sciences (Aerospace Engineering)

University of California San Diego, 2024

Professor Oliver T. Schmidt, Chair

Particle image velocimetry (PIV) is a non-intrusive optical technique to measure the velocity in fluid flows by relying on a tracer particles seeded inside of the flow, however this method has various issues that could compromise data acquisition and cause regions of missing data, or gaps: irregular seeding, light reflecting off surfaces, light path obstruction, poor particle choice, and imaging system restrictions. To reconstruct these missing regions, Nekkanti and Schmidt [8] developed the Gappy Spectral Proper Orthogonal Decomposition (Gappy SPOD), as a spectral counterpart to the commonly used Gappy Proper Orthogonal Decomposition (GPOD). This thesis presents improvements made to the Gappy SPOD that allow for application to raw PIV data, including a detailed gap finding method and processing of black zones (regions missing

in all snapshots). We then compare this method with MF GPOD, and Nearest Neighbor (NN) interpolation in terms of their turbulent kinetic energy (TKE) error in both a global and local region capacity. We find that Nearest Neighbor often outperforms Gappy SPOD or GPOD in regions containing few missing elements. This performance suffers in regions where gap size exceeds 10^3 elements and Gappy SPOD and MF GPOD provide more accurate results. The raw data sets examined here contain greater amounts of small gaps, thus the proposed hybridization of these methods in this thesis will lead to greater accuracy.

Introduction

During experimentation there is concentrated effort to obtain highly accurate information with the appropriate sensors. Often these sensors are fallible and produce inaccurate and/or incomplete data. Calibration helps to safeguard against inaccurate results through comparison between a known quantity, another reliable sensor or known phenomenon. Similarly,there is an ongoing endeavour to mitigate the effects of incomplete and missing data.

Reconstruction techniques for missing or unobserved data can be found in almost every discipline, whether that be upscaling image resolution through nearest neighbor interpolation methods [2], improved fault detection in network control systems [19], application in oceanography and other natural sciences [1].The types of algorithms developed for these purposes typically fall into four categories: spatial, spectral, temporal, and hybrids [16]. This thesis contains methods that incorporate all of these methods and even propose hybridization to help improve reconstruction techniques.

This thesis is set in the context of particle image velocimetry (PIV) data of open-cavity and high-lift device flow by Yang and Cattafesta at the college of engineering from Florida state university [21][26]. Obstruction of the light path by objects, reflections of light from the surface of objects, the inaccessibility of regions for the imaging system, and irregular seeding are a few of these possible sources that would produce missing regions in PIV data sets.

Kriging[4][9], optimal interpolation [10], and a least squares estimation conjoined with proper orthogonal decomposition (POD) known as 'gappy POD' (GPOD) [3] have been applied to open cavity flow. Efforts by by Venturi and Karniadakis [18] extended the gappy POD algorithm showing greater reconstruction ability when compared to kriging. Gappy POD has seen

use in a wide variety of fluid flow reconstruction, such as gas turbine combustors [12]. POD has a spectral counterpart known as spectral proper orthogonal decomposition (SPOD) had its framework outlined by Lumely in 1967 [6], yet despite this early conception SPOD had not seen much use; GPOD remains state of the art. The SPOD was then shown to combine the benefits of POD and dynamic mode decomposition (DMD) to show promising application in fluid flows[17] and was documented to inform the wider fluids community of its use [14]. A method of PIV data reconstruction is then introduced by Nekkanti and Schmidt [8], which involves leveraging the temporal and spatial correlation of the SPOD to perform an inversion to reconstruct and capture the underlying missing structures and energy of the fluid flow.

This thesis extends the work of Nekkanti by applying the Gappy SPOD algorithm to raw PIV data in order to compare its effectiveness against other methods, highlight potential issues, and aid in establishing best practices. The initial work relies on the introduction of artificial gaps into PIV data, recovering 97% and 80% of the flow field in presence of noise and 20% missing data; outperforming standard methods such as kriging and gappy POD. The missing data reconstructed in this paper is formed from naturally occurring issues with experimental equipment. We compare the reconstruction attempts by Gappy SPOD to Median filter (MF) GPOD, which uses an adaptive filter to determine if points should be updated during reconstruction [12], and nearest neighbor interpolation, a simple method that nevertheless appears in various forms of reconstruction [11] [2].

Chapter 1

Methodology

In this chapter we will discuss the procedures and algorithms utilized in the analysis performed in later sections. The first part of this chapter will be devoted to an explanation of the SPOD itself and the derived reconstruction algorithms. These algorithms will then be built upon with pre- and post-processes that will expand the algorithm to be more versatile when applied to raw experimental data and finally end with the error metrics definitions that we will use during analysis.

1.1 SPOD

1.1.1 Spectral Proper Orthogonal Decomposition Overview

This section will contain a condensed version of the detailed explanation of the SPOD described in the ”Guide to Spectral Proper Orthogonal Decomposition” [14].

Spectral proper orthogonal decomposition is a method that extracts the modes and structures from flow data in a way that optimizes the orthogonal basis such that a subset of modes capture a larger fraction of energy than any other orthogonal basis. This is a characteristic that it shares with ’standard’ POD, however what sets SPOD apart is that its modes vary in both space and time; being orthogonal under a space-time inner product.

The SPOD assumes that the flow is statistically stationary, mean and variance are time-invariant, with modes being harmonic in time and computed one frequency at a time from the

data. Thus, making the SPOD modes the eigenvectors of the cross-spectral density (CSD) tensor at each frequency.

The spectral portion of the decomposition complicates the method, requiring Welch's method to accurately estimate the CSD from a time series, whereas the POD portion is straight forward by only involving the eigenvectors of a matrix.

We shall define a flow quantity $\mathbf{q}$ that is a stochastic process where the expectation $E[\mathbf{q}] = 0$, satisfying the statistically stationary assumption. Should this not be fulfilled with the initial data, then we must first center it about its mean and work with the fluctuations $\mathbf{q}^{'} = \mathbf{q} - \overline{\mathbf{q}}$. To remain consistent with the referenced notation we will assume this step has been taken and exclude the prime notation. A discrete form of the SPOD procedure is provided in the next lines.

To start we notate our ensemble with n_t snapshots as

$$\mathbf{q}_i = \mathbf{q}(t_i) \;\; i = 1, 2, ..., n_t \tag{1.1}$$

The first steps to then implement Welch's method is to segment the data into n_{blk} overlapping blocks, each containing n_{fft} snapshots, with n_{ovlp} describing the number of overlapping snapshots. We then denote $\mathbf{q}^k(t_j)$ as the k-th realization of the observation vector at discrete time t_j. Assuming that each realization is available at regular time intervals $t_j = t_0 + jT/n_{fft}$ over time period T, where $j = 1, 2, ..., n_{fft}$. Since the flow is assumed to be stationary the starting time t_0 is arbitrary. We then take a Fourier transform as shown in 1.2.

$$\hat{q}^k(f_m) = \sum_{j=0}^{n_{fft}-1} q^k(t_{j+1}) e^{-i2\pi jm/n_{fft}} \;\; k = -n_{fft}/2 + 1, ..., n_{fft}/2 \tag{1.2}$$

Where f_m are the discrete frequencies, $f_m = m\, n_{fft}/T$ and $m = -n_{fft}/2, ..., n_{fft}$. The corresponding inverse Fourier transform can be stated as

$$q^{(k)}(t_{j+1}) = \frac{1}{n_{fft}} \sum_{m=-n_{fft}/2+1}^{n_{fft}/2} \hat{q}^{(k)}(f_m) e^{i2\pi jm/n_{fft}} \;\; j = 0, ..., n_{fft} - 1; \tag{1.3}$$

Once transformed we can then make an ensemble one frequency at a time by constructing the data matrix

$$\hat{Q} = [\hat{q^1}\ \hat{q^2} \dots \hat{q^N}], \quad \hat{Q}\ \varepsilon\ \mathbb{C}^{M \times N} \tag{1.4}$$

With N realizations and M degrees of freedom. We then write the CSD as

$$\hat{C} = \frac{1}{N-1}\hat{Q}\hat{Q}^H \tag{1.5}$$

Which can then be solved either as

$$\hat{C}W\hat{\Phi} = \hat{\Phi}\Lambda \tag{1.6}$$

or by the method of snapshots.

$$\hat{Q}^H W \hat{Q}\hat{\Psi} = \hat{\Psi}\hat{\Lambda} \qquad \hat{\Phi} = \hat{Q}\hat{\Psi} \tag{1.7}$$

In figure1.1 a graphical representation of this process is displayed.

The weight matrix W in equations 1.6 and 1.7 determines the orthogonal and optimal properties of the modes. An important property is the ability of the weight matrix to apply the SPOD to a subset or subdomains of the variables in the solution vector. We can exclude portions of the domain by assigning them zero weight. This allows them to remain in the part of the spatial coherence for the modes and permits their physical interpretation without affecting the SPOD spectrum. We use this to our advantage when performing SPOD reconstruction to ensure only missing areas are reconstructed.

In order to accurately estimate the CSD matrix shown in equation 1.5 we need to choose appropriate parameters. Sampling parameters are n_t, and Δt and spectral parameters are n_{fft} and

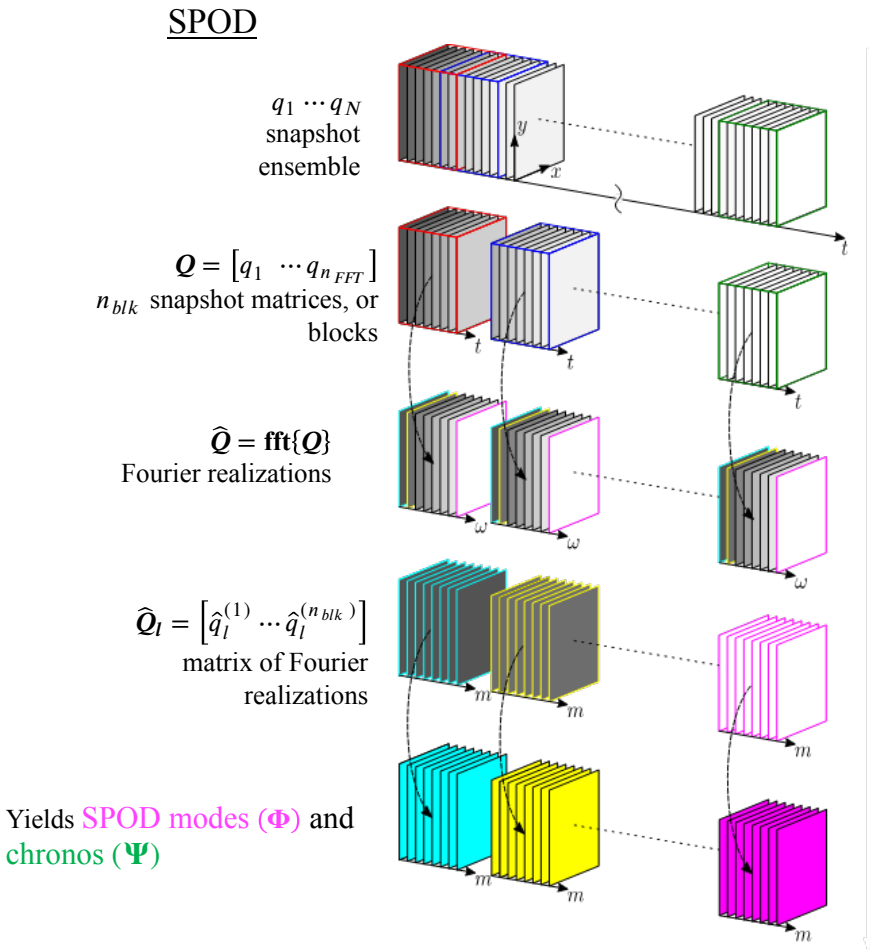


Figure 1.1. Depiction of SPOD procedure as displayed in [13]

n_{ovlp}. There has been much work on standards and best practices done on finding the optimal parameters, which can be found in most signal processing textbooks. We leave such details for the reader to explore. These parameters attempt to solve issues that could introduce bias or error in the reconstruction: sampling bias, windowing, Confidence bounds, and data storage.

Biasing is the error that occurs due to aliasing effects by not sampling at a rate that satisfies the Nyquist criterion. Common strategies are to over-sample or implement low-pass filters to attenuate the frequencies above the Nyquist frequency. In terms of our parameters we can see the relations in the following equations.

$$f_s = \frac{1}{\Delta t} \tag{1.8}$$

$$f_{max} = \frac{f_s}{2} \tag{1.9}$$

where f_s is the sampling rate determined by the temporal step size 1.8. This allows us to determine the max resolvable frequency by applying the Nyquist–Shannon sampling theorem. 1.9.

The period of each block of the Fourier transform is determined by n_{fft} as seen in equation

$$T = n_{fft}\Delta t = \frac{n_{fft}}{f_s} \tag{1.10}$$

which leads to the bin size (frequency resolution) given as

$$\Delta f = \frac{1}{T} = \frac{f_s}{n_{fft}} \tag{1.11}$$

if Δf isn't made to be small enough then distinct content will smeared across the bins. We can choose a large n_{fft} to make this bias small, however choosing a large n_{fft} will reduce the number of segments in the ensemble and prevent statistical convergence. To mitigate the uncertainty that occurs from a small bin size we can segment the data into overlapping blocks, thereby increasing the number of segments. We denote the number of overlapping points with n_{ovlp} and determine the number of block using the following equation

$$N_{blk} = \left\lfloor \frac{n_t - n_{ovlp}}{n_{fft} - n_{ovlp}} \right\rfloor \tag{1.12}$$

where $\lfloor \cdot \rfloor$ represents a floor operator.

Since we are applying a discrete Fourier transform there is a possibility for spectral leakage, when the data is non-periodic sharp peaks will "leak" into other bins causing the peaks

to be attenuated. Windowing is often used to mitigate this effect. SPOD uses the Hamming window as its default windowing method.

This will conclude the discussion on the basic SPOD procedure. This section was meant to provide the necessary context for the upcoming discussion; if the reader would like to read in more detail about the SPOD we shall refer them to [7][14] [8].

1.1.2 Inversion of the SPOD

Now that the standard form of the SPOD has been described it is time to present the inversion of the SPOD, which lies at the core of the reconstruction algorithm. Now there are two methods of reconstruction presented in [8] we will only present the relevant SPOD inversion method here and leave the oblique projection method up to the readers discretion to seek out.

The first step in inverting the SPOD is ensuring the SPOD basis is weighted correctly. This can be accomplished by adjusting the expansion coefficients and using them to reconstruct the Fourier realizations at each frequency.

$$\hat{Q}_k = \Phi_l A_l \tag{1.13}$$

Where A_k is the matrix of expansion coefficients:

$$A_l = \sqrt{n_{fft}}\Lambda^{\frac{1}{2}}\Psi_l^H = \Phi_l^H W \hat{Q}_k \tag{1.14}$$

From this equation we can see that the expansion coefficients can be found during the SPOD computation or recovered later. We will drop the l notation on A for the following, since the SPOD eigenvalue problem is solved at each frequency separately.Taking 1.14 into matrix form it can be shown that the columns of A contain the expansion coefficients that allow for the reconstruction of a particular Fourier realization.

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,nblk} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,nblk} \\ \cdots & \cdots & \ddots & \cdots \\ a_{nblk,1} & a_{nblk,2} & \cdots & a_{nblk,nblk} \end{bmatrix} \tag{1.15}$$

Similarly the coefficients in A can be used to expand specific SPOD modes in terms of the Fourier realizations. This can be seen by rearranging equation 1.13

$$\Phi_l = \frac{1}{n_{blk}} \hat{Q}_l A_l^H \Lambda_l^{-1} \tag{1.16}$$

Following 1.16we can write the Fourier transform reconstruction of the k-th block as

$$\hat{Q}^{(k)} = \left[\left(\sum_i a_{ik} \Phi^{(i)} \right)_{l=1}, \left(\sum_i a_{ik} \Phi^{(i)} \right)_{l=2}, \cdots, \left(\sum_i a_{ik} \Phi^{(i)} \right)_{l=n_{fft}} \right] \tag{1.17}$$

We can then recover the original data $Q^{(k)}$ by using an inverse Fourier transform.

$$q_j^{(k)} = \frac{1}{w(j)} \mathscr{F}^{-1} \{ \hat{q}_j^{(k)} \} \tag{1.18}$$

These steps are visually summarized in figure 1.2 .

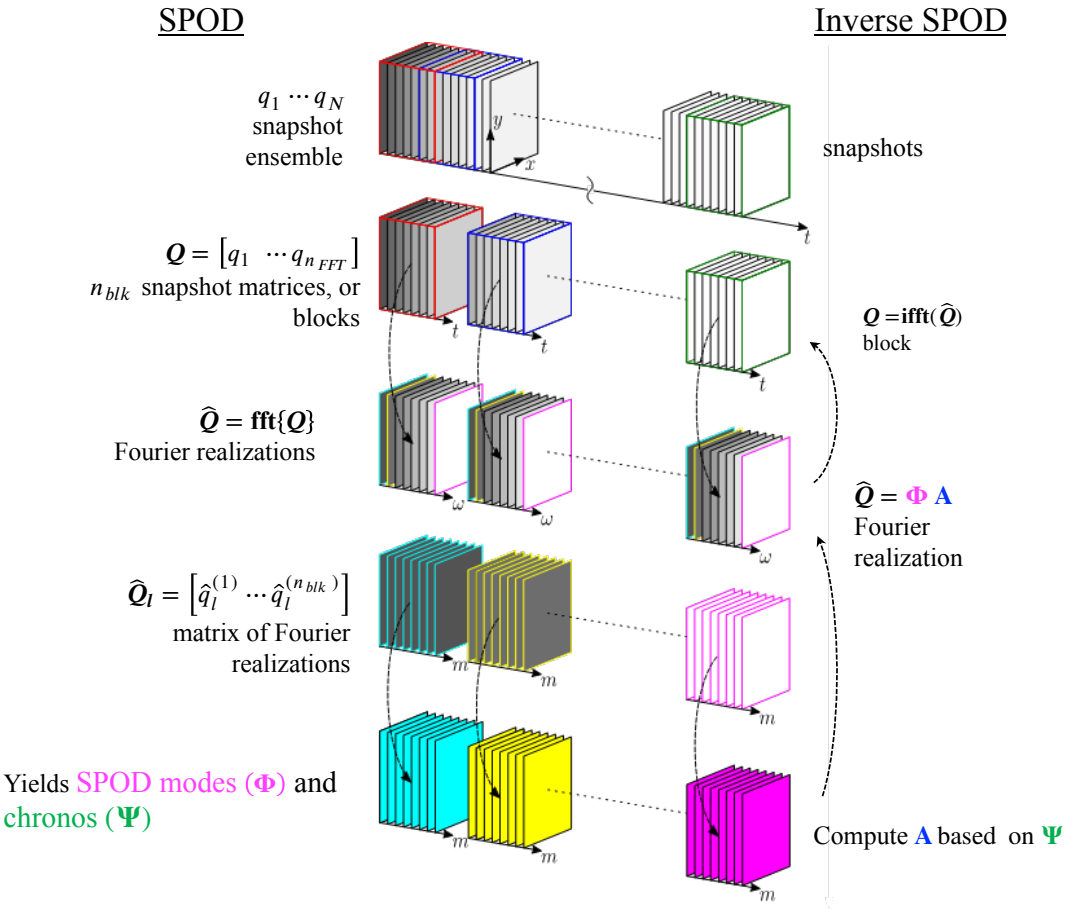


Figure 1.2. Depiction of the SPOD and inverse SPOD process modified from [13]

1.2 SPOD Reconstruction Algorithms

We shall now discuss the algorithm that uses the SPOD for reconstruction. There are two ways in which this is implemented: gap-wise and block-wise. We will present the algorithms here.

1.2.1 Gap-wise Reconstruction

The gap-wise implementation is the original version of the algorithm used in previous works. Unlike the block-wise method, gap-wise reconstructs only one individual gap at a time, slowly iterating through the data until convergence has been obtained for all gaps. In order to only reconstruct gappy regions and prevent alterations to the original reference data, we can

set zeros for locations that do not need to be reconstructed in the weight matrix during SPOD inversion. The algorithm adapted from [8] is as follows:

1. Segment the time series into overlapping blocks and compute the temporal Fourier transform of each block (if not previously computed).

2. Proceed to the n-th gap and choose all the realizations of the Fourier transform that are not affected by this gap.

3. Compute the SPOD from all the Fourier realizations that are not affected by this gap (equations 1.7 and 1.16) and save the corresponding expansion coefficients (equation 1.14).

4. Compute the SPOD expansion coefficients for blocks affected by the n-th gap by projecting their Fourier transforms onto the SPOD basis

5. Reconstruct the affected blocks by inverting the SPOD (equation 1.17) from the expansion coefficients computed in (3) and (4); replace the regions affected by the n-th gap.

6. (local loop) Go to (4) to update the expansion coefficients now that the data is reconstructed in the affected regions until convergence criterion based on change of reconstruction of n-th gap is met.

7. (inner loop) Set n $\leftarrow$ n+1 and go to (1) until all gaps are reconstructed.

8. (outer loop) Set n = 1 and go to (1); repeat until convergence criterion based on change of reconstruction between outer loop iterations is met.

We will discuss convergence and error metrics in a later section.

1.2.2 Block-wise Reconstruction

The Block-wise reconstruction algorithm is very similar to the gap-wise, but makes one notable difference. Instead of reconstructing each gap individually it reconstructs entire blocks

from the ensemble at a time. This allows for multiple gaps to be reconstructed at a time, making it easier to implement and have greater performance time when reconstructing data with a large number of individual gaps. The following is the procedure adapted from [15]

1. Segment the time series into overlapping blocks and compute the temporal Fourier transform of each block. ‘

2. Proceed to the n-th block and identify all FFT blocks whose fraction of shared gaps is below $\text{gap}_{ovlp,max}$. If fewer than r_{min} meet this criterion, select the r_{min} blocks with the lowest shared gap percentage.

3. Compute the SPOD using equation 1.7 from the selected FFT blocks. Restrict the SPOD basis to r_{max} modes per frequency.

4. Compute the expansion coefficients for n-th block by projecting its FFT onto the new SPOD basis using equation 1.14.

5. Reconstruct the n-th block by inverting the SPOD using equation 1.17 based on these expansion coefficients. Then replace the data in the missing regions of the n-th block with the reconstruction.

6. (Inner loop) Update the FFT of the n-th block and go to (4) until the relative change between the current and previous inner iteration, measured in the L2-sense, falls below the tolerance, tol_{inner}, or until the maximum number of allowed inner iterations, $n_{inner,max,}$ is reached.

7. Let n $\leftarrow$ n+1 and go to (2) until all blocks are reconstructed.

8. (Outer loop) Let n $\leftarrow$ 1 and go to (1) for n_{outer} times.

1.3 Pre-Processes and Post-Processes

In this section we will cover the important processes that are implemented to apply the reconstruction method to raw experimental data. The first being an gap finding routine that will be run before the reconstruction, and secondly a discussion of black zones.

1.3.1 Gap Finding Algorithm

Let us now address the motivation behind this entire thesis, missing values present in experimental data, which we will call gaps.

It is necessary to find the locations in which these gaps occur in order to reconstruct them and them alone. There are notable differences between the gaps we work with here and those in the previous work done by Dr. Nekkanti. Namely, these are naturally occurring gaps that do not have a patterned form; they can freely occur within any space in the FOV. As mentioned previously, there are many reasons these gaps could exist in PIV data, however we are not concerned with their origin in this thesis. The artificial gaps previously tested with were all rectangular prisms. Their locations were randomized but their shapes were not. This made it easy to find the locations of these gaps and to reconstruct them. Since raw data does not have such a conspicuous pattern we must implement a robust method that will meticulously record the index location of every gap. This may seem unnecessary considering the context of block-wise reconstruction, however such information is relevant to the function of the gap-wise method and will present itself useful in analysis and future applications.

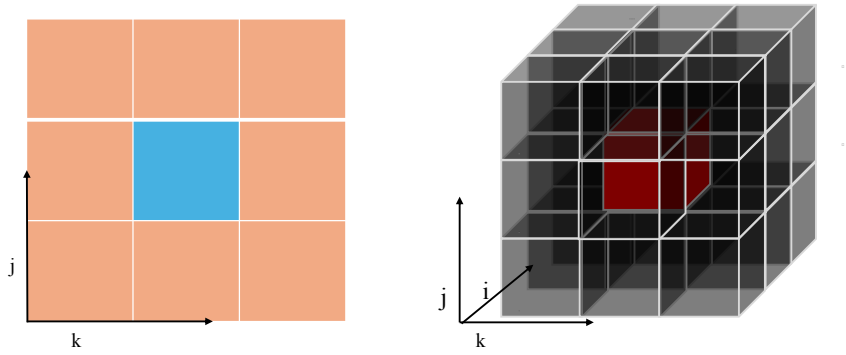
The method we decided to pursue was a simple search algorithm based on a 3-D Moore's neighborhood as shown in figure 1.3. This allows us to search for all connections possible in 3-D and track gap shapes with complete accuracy. The types of gaps this method could group together is displayed in figure 1.4, highlighting a possible a drawback that will discussed in chapter 4

The first step is to recognize how they are treated inside of the data itself. In the context

of MATLAB they appear as Not a Number (NaN), which is consistent across many programming languages. NaN is a unique object inside of MATLAB and has many built-in tools to handle the processing of NaNs that we will be taking advantage of. The following procedure outlines the working of the algorithm:

1. Loop through data until you find the location of a NaN, create a new list
2. check Moors neighborhood around current NaN for other NaNs
3. save found NaN locations in list
4. repeat steps 2-3 for each NaN in list
5. Once all NaNs in list have been checked, continue looping through data and repeat steps 1-4 until all NaNs have been accounted for.

This procedure will create a list for each gap containing the locations of all NaNs comprising that gap. These gaps can then be stored into a cell array such that every cell represents a single gap. Due to the large size of the data (16kHz and 11kHz) it is necessary to ensure the algorithm is performing as efficient as possible to reduce run time. With that in mind we run this algorithm in logical arrays and only store index locations in single float. An important thing to note is that the black zones should be removed before performing this search, else you will suffer significant increases in run time with this method. More will be discussed on black zones in the next section.



(a) 2 -D (b) 3 -D

Figure 1.3. Visual representation of the Moor's neighborhood in index notation. 2-D (a) and 3-D (b) are shown. Centers about which the algorithm searches are shown in blue and red respectively

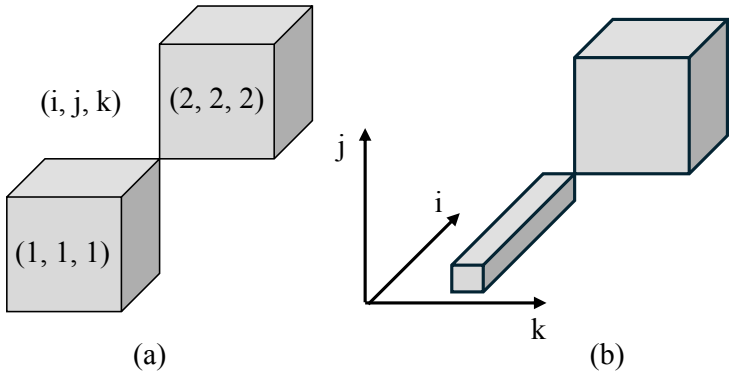


Figure 1.4. Example of possible connections. The left side (a) shows the individual elements being connected with index locations given. The right (b) illustrates possible large gap connections.

1.3.2 Black Zones

Black zones are a unique case among missing data. The diction comes from Gunes in their paper "Gappy data: To Krig or not to Krig?" [4], where a black zone is defined as a region of the domain in which data is missing at all points in time.

Since these points are missing at all instances in time there is no reference data to compare them to. In the previous work the domain was truncated to exclude these areas, a valid method

to deal with them. However part of the goal of this work is to provide accurate reconstruction methods that will enable to improved experimentation by supplementing the limitations of high speed cameras. One way in which we envisioned this was to support the use of multiple smaller cameras to capture a larger flow field, by ”stitching” the PIV data from multiple sources together. Black zones are an inherent part of this experimentation process, so we sought ways provide a coherent strategy to handle these zones.

We concluded on a method to extrapolate to these zones during the post-processing. This can be done in a number of ways: Kriging in the flow field, Kriging the SPOD modes and inverting them to reconstruct these locations, or simply use an algorithm like nearest neighbor. As we do not have reference data in these locations we default to nearest neighbor and allow the user to determine whether they would like to perform this extrapolation or not.

The structure of blackzones makes them easier to find without the gap finding algorithm detailed above, so they are removed before finding the other gaps to reduce run time.

1.4 Nearest Neighbor Algorithm

A major component of this paper is the comparison between reconstruction method. We have selected a nearest neighbor interpolation as one of these methods to compare against as it is significantly different in its approach.

Nearest Neighbor interpolation algorithms, traditionally created to solve the traveling salesman problem [5], simply replaces the query point with the nearest point. The merits to this type of method are its low cost and ease of implementation, however these algorithms are not known for their optimality or accuracy.

In this work we utilized the recent MATLAB 2023 $fillmissing2$ function, which replaces NaN in a 2-D matrix with a user selected function; Nearest Neighbor being the function in this case. We reconstructed the 2-D flow field of individual snapshots with this method, thus making the reconstruction reliant along the local spatial dimensions.

Through testing we were able to determine that the MATLAB nearest neighbor filling functions operates based on a directional priority inside of the data matrix: Right $\rightarrow$, Down $\downarrow$, Up $\uparrow$, Left $\leftarrow$

It will always choose the closest euclidean distance but if two points are equidistant it will choose the one according to priority. This can be seen in the following simple examples:

Example 1: given a choice of (1 left) or (1 right, 1 down) it will take the value 1 left.

Example 2: given a choice between (1 right, 2 down) or (1 right, 2 up); it will choose the (1 right, 2 down) value.

1.5 Error Metrics

The error metrics defined here are similar to the work in done by Nekkanti and Schmidt [8], but are extended to include absolute error.

1.5.1 Error Definitions

We define G as the index set corresponding to all gappy regions and G_n,$B_k \subset$ G, where G_n is the subset of indices corresponding to the n-th gap and B_k is the subset of snapshots comprising the k-th block. We use both absolute and relative error to evaluate the efficacy of the methods. These errors reflect errors in the 2-D turbulent kinetic energy (TKE), $TKE = \frac{1}{2}\left(\overline{(u')^2} + \overline{(v')^2}\right)$ with units of joules per kilogram $[J/kg]$.

Absolute Error(AE):

$$AE_i = \frac{1}{2N}\|q_i^g - \tilde{q}_i^g\|_2^2 \text{ (absolute error of the i-th snapshot)} \tag{1.19}$$

$$AE_n = \frac{1}{2N}\sum_{(i,j,k)\in G_n}\|q_{(i,j,k)}^g - \tilde{q}_{(i,j,k)}^g\|_2^2 \text{ (absolute error of n-th gap)} \tag{1.20}$$

$$AE_k = \sum_{i \in B_k} AE_i \text{ (absolute error of k-th block)} \tag{1.21}$$

$$AE_{global} = \sum_{i \in G} AE_i \text{ (Global absolute error)} \tag{1.22}$$

Where $\|\cdot\|_2$ represents the L2-norm. $\tilde{q}^g$ is the reconstruction in the gappy regions, q^g is the reference data in the gappy regions, and N is the number of elements in the vector q^g. Then to create a relative error we simply divide the snapshot error by the TKE of the original snapshot reference data.

Relative Error(RE):

$$RE_i = \frac{\|q_i^g - \tilde{q}_i^g\|_2^2}{\|q_i^g\|_2^2} \text{ (relative error of the i-th snapshot)} \tag{1.23}$$

$$RE_n = \sum_{i \in G_n} RE_i \text{ (relative error of n-th gap)} \tag{1.24}$$

$$RE_k = \sum_{i \in B_k} RE_i \text{ (relative error of k-th block)} \tag{1.25}$$

$$RE_{global} = \sum_{i \in G} RE_i \text{ (Global relative error)} \tag{1.26}$$

the following shows the convergence metric for both the gap wise and block wise methods.

$$C_n = \sum_{i \in G_n} \frac{\|\tilde{q}^j - \tilde{q}^{(j-1)}\|_2^2}{\|\tilde{q}^{(j-1)}\|_2^2} \text{ (convergence of n-th gap)} \tag{1.27}$$

$$C_k = \sum_{i \in B_k} \frac{\|\tilde{q}^j - \tilde{q}^{(j-1)}||_2^2}{\|\tilde{q}^{(j-1)}\|_2^2} \text{ (convergence of k-th block)} \tag{1.28}$$

Where j is the iteration number. This convergence is only applied to the Gappy SPOD and not nearest neighbor or Median Filter (MF) GPOD, and does not require reference data unlike the error calculations.

1.5.2 Gap Shifting Procedure

In order to apply these error metrics we required a reliable reference for comparison. Often, Direct Numerical Simulation (DNS) data is used for these purposes, however such data is not always available. DNS prioritizes accuracy, often making the computation of DNS data a slow process.

Instead we have opted to expand on the idea of artificial gaps by shifting original gaps into subsequent snapshots. We would like to emphasis that we are not adding any arbitrary missing points to the data; we are simply shifting the existing gaps in the time domain by an arbitrary 50%, maintaining the original gap shape. We chose 50% believing it provided a sufficient shuffling of the gaps to leave a distinct enough change to the gap topology within the snapshots.

While this eliminates costly computations it does come with the necessity of a greater level of detail when implementing. The biggest issue is dealing with overlaps between the original gap locations and the newly shifted gap locations. To ensure that we are computing true error, we must make sure that the overlapping gap locations are not included in our error computations; this is simply done by zeroing their weight during error computation.

An example of this procedure is shown in figure 1.5 where we start with the raw data set (a) and record the original gap locations (dashed outlined in magenta) and fill in those gaps with Nearest Neighbor interpolation to prevent future processing errors. Then for the same snapshot we introduce gaps from a different snapshot, as seen in (b) with the shifted gaps solid outlined in red and the original gaps dashed outlined in magenta still. The overlapping portions are then weighted out, presenting regions where a true error can be taken. (c) displays the original snapshot before interpolation with the regions used for error analysis outlined in black.

Of course this procedure can only be done with in-depth knowledge of gap locations, creating a dependence on a gap finding algorithm. Another trade-off to be taken into consideration, is that due to overlap the total number of points that can be used for error analysis is ultimately

reduced.

This ends our discussion of the background information to contextualize the upcoming results. Any deviations of these methods during application or unique considerations will be discussed as needed with their accompanying results.

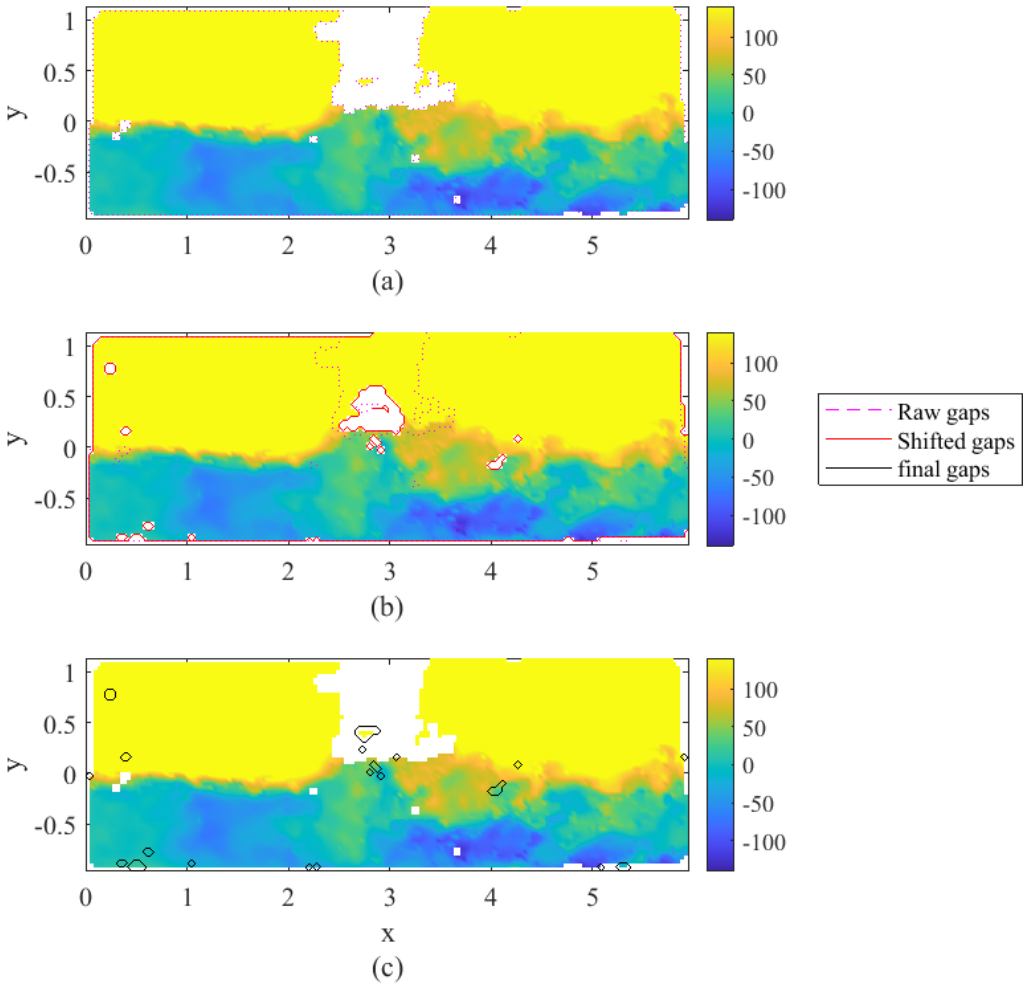


Figure 1.5. Gap shifting procedure. Original gap locations (a) (magenta dashed) are shifted by $n_t/2$ snapshots (b) (red solid).Then overlapping gaps are removed (c) (black solid).

Chapter 2

Results: Open-Cavity Flow

This chapter covers the results from the open-cavity flow time resolved (TR)-PIV data provided to us by Zhang and Cattafesta from the Florida Center for Advanced Aero-Propulsion at Florida State University. A more detailed breakdown of their experimental setup can be found in [20][21][26], which is the source of the information presented here.

A blow-down windtunnel produces High-speed subsonic flow (Mach 0.6) over a rectangular open-cavity model with dimensions of $L/D = 6$ and $W_c/D = 3.85$ where $D = 26.5mm$. A photonics DM dual-head laser and a Phantom V2012 high-speed camera are used to take snapshots at a sampling rate of 16 kHz for one second in double-frame mode with an image resolution of 1280×464 pixels [26]. A schematic of this setup is shown in figure 2.1.

In table 2.1 the parameters used for SPOD reconstruction is provided.

Table 2.1. Parameters for SPOD reconstruction

case	Mach	n_x	n_y	n_t	n_{fft}	n_{ovlp}	n_{blk}	tol_{inner}	$gap_{ovlpmax}$
raw	0.6	160	57	16000	128	64	249	10^{-8}	20%
Artificial gaps	0.6	156	55	16000	128	64	249	10^{-4}	20%

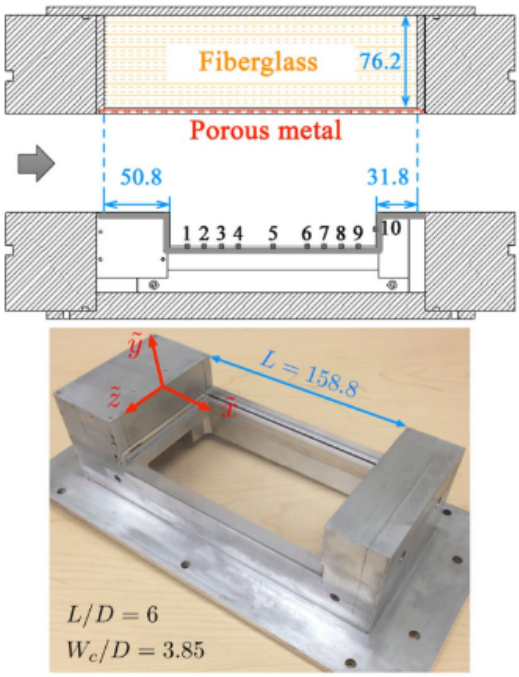


Figure 2.1. Setup schematic of open-cavity flow experiment taken from [26],[25].

2.1 Raw Gaps

This section will cover the data used that involves only the gaps present in raw data provided. We will first provide gap statistics to provide greater insight into the areas of reconstruction and how characteristics of those areas will influence the accuracy of reconstruction. An example of the reconstruction can be seen in figure 2.2 with gappy areas outlined in black.

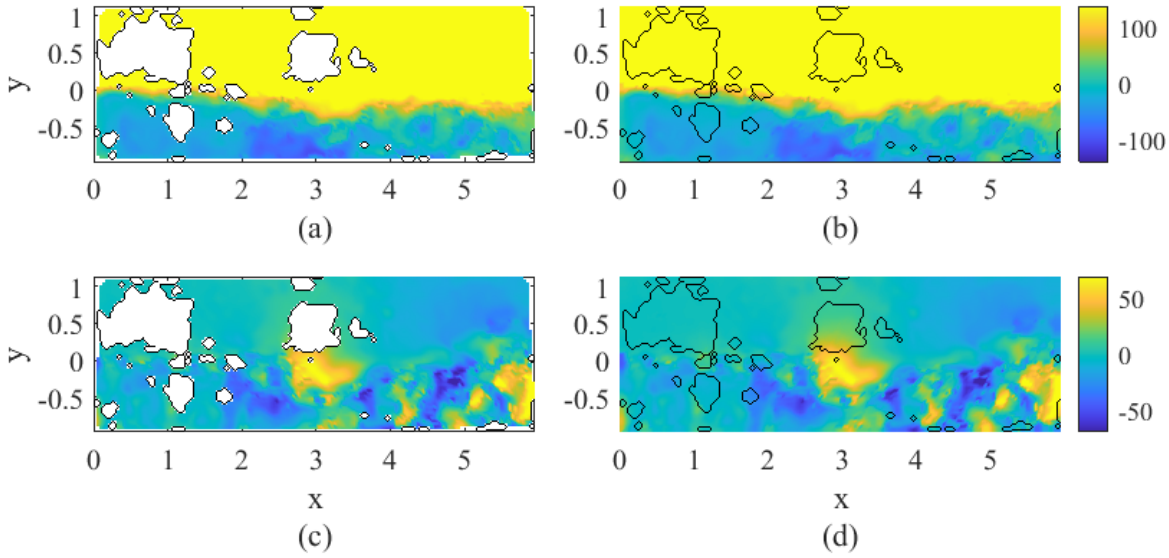


Figure 2.2. Block-wise reconstructions for raw open-cavity flow data. Shifted regions are shown for U (a) and V (c) velocity. Reconstructions (b) and (d) have color maps of $\pm 140 m\backslash s$ and $\pm 40 m\backslash s$ respectively

2.1.1 Gap Statistics

The first step in our reconstruction was to implement of the gap finding method detailed in the methodology section. Here we shall present the results provided to us by this method, which will be used to contextualize the results presented in the next section. It is important to note that the gaps are consistent in both u- and v- velocity components of the cavity data. As such we only search one component to reduce computational costs and to prevent inflating the gap statistics. Black zones have been removed for the detailed statistics but are still discussed appropriately.

Table 2.2. Breakdown of gappyness percentage

	total # of elements	# of missing elements	% gappyness
Non-Black zones	1.4592×10^8	2.4674×10^6	1.69%
Black zones	1.4592×10^8	5.7920×10^6	3.97%
Overall	1.4592×10^8	8.2594×10^6	5.66%

First we would like showcase the overall gap presence exhibited in this data set. From

Table 2.3. Raw open-cavity gap sizes and percentage of overall gaps and total elements

gaps size	% of gaps	% of total elements
10^0	84.31%	0.16%
10^1	14.16%	0.24%
10^2	1.29%	0.24%
10^3	2.4%	0.42%
10^4	5.7396×10^{-3} %	0.0504%

table 2.2 we can see that 5.66% of the data is missing with around 70% of that missing data comprised of blackzones. This means that almost 4% of the data is comprised of black zones, only leaving 1.69% to be reconstructed with the gappy algorithms. Once we perform the shifting procedure this results in a non-black zone element reduction of 749297 leaving 1718103 or approximately 1.12% of total number of elements available for reconstruction comparison.

In figure 2.3 we show a breakdown of the gap sizes encountered and their frequency of occurrence post overlap removal to remain consistent with the error comparisons, Since these are the gaps used during during error computations. A breakdown of the same form for the raw gaps pre-shifting procedure can be viewed in A.2. To help contextualize the gap distribution shown in figure 2.3, we provide percentages of the gap sizes orders of magnitudes with respect to the total number of gaps found and with respect to the total number of elements in the data set in table 2.3. The largest occurring type of gap are single element gaps. With 40520 occurrences this comprises 38.76% of the 104537 recorded gaps. When looking at all gaps on the order of 10^0, we see that they comprise 84.31% of the recorded gaps, whereas the largest gaps, on order 10^4, constitute only 5.7396×10^{-3} % of the recorded gaps. In terms of # of elements 10^0 comprise 0.16%, the 10^4 gaps make up 0.0504%, clearly displaying a tendency to containing smaller gap size. This provides important context when viewing the absolute error results in the next section.

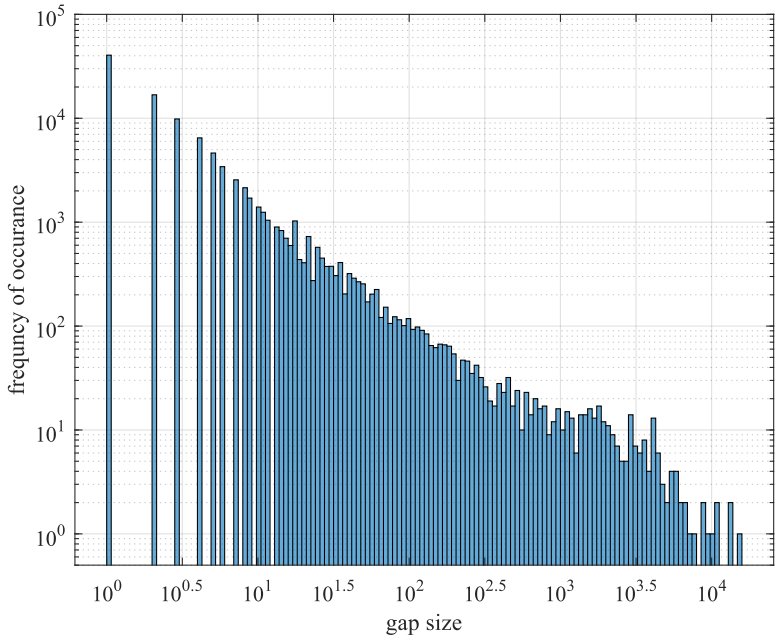


Figure 2.3. Histogram of gap sizes versus their frequency plotted on a log-log plot. Gap size is determined by the number of elements that comprise a single gap.

2.1.2 Method Comparison

In this section we will present and discuss three methods of reconstruction on the current data set: Nearest Neighbor interpolation, Gappy SPOD, and MF GPOD. Due to the large number of gaps that need to be reconstructed, we we use the block-wise methods during these reconstructions. Our collaborators Cattefesta and Zhang used their MF GPOD method to reconstruct the missing data after the shifting procedure had been performed. This was conducted as a blind test, as they were only given the data set to reconstruct without much information behind it.

Table 2.4. Absolute global errors of Raw Open-Cavity Flow

Method	Absolute Global error
NN	1.4114×10^6
GPOD	2.5598×10^6
SPOD	4.0710×10^6

In table 2.4 we present the absolute global error according to equation 1.22 of all

three methods. We observe that nearest neighbor outperforms when compared to the other methods. GPOD has error between SPOD and NN but is slightly closer to NN. This presents an unanticipated result of a typically less accurate method providing a more accurate reconstruction then the more commonly employed methods. In light of this result, it becomes imperative that we provide a more detailed breakdown of the reconstructions to gain a better understanding of how these methods interact with the data set.

Gappy SPOD Block-wise Reconstruction

Here we present the convergences and iteration error relationships during reconstruction.In figure 2.4 the relative global errors per iteration are displayed calculated according to equation 1.26 and normalized according to the first block-wise iteration result. We see there is a large increase in error from the NN initialization point, this is expected in the context of table 2.4 where NN has an overall lower relative global error. We see convergence early on making the tol^{-8} and the 4 outer iterations (as shown by the red vertical lines) overly strict and costly with little gain.

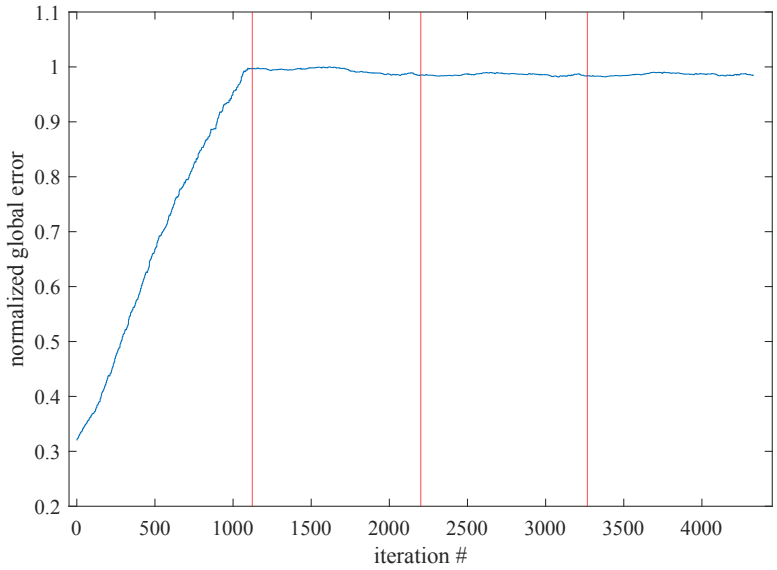


Figure 2.4. Global error per iteration of block-wise reconstruction, normalized about the maximum point. Red vertical lines represent outer iterations

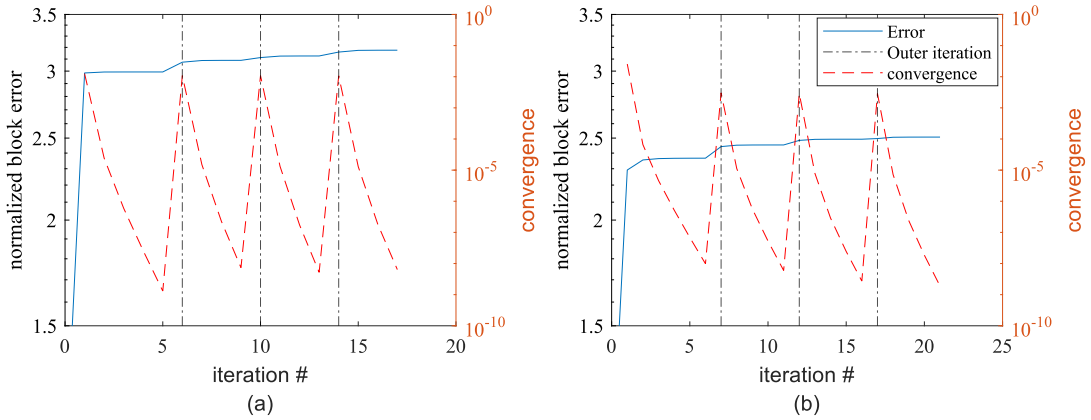


Figure 2.5. Block reconstructions with relative error and convergences vs iteration count. (a) is the block containing the fewest amount of missing points (4785) and (b) is the block containing the most (36360).

The block error and convergence were computed according to equations 1.25 and 1.28 respectively, using the error at initialization a normalizing constant. This can be seen in figure 2.5. We show to samples: a block containing the fewest amount of points (a) and one containing the most (b). We can see that the relative error in the block with the most amount of missing points does not increase as much as the block with fewer points. In both instances we see convergence reached after only a couple iterations, but immediately peaks during the beginning of a new outer iteration.

These results show a poorly performing algorithm, which is counter to the results displayed by Nekkanti and Schmidt[8]. The main difference between the works is the data set in use. We shall discuss it more in-depth soon, but the data used in their work included greater amounts of missing data, notably up to 20% missing and include large regions of missing data. This data set contrasts greatly with that by being biased towards smaller gaps, and we can see that the relative error is somewhat less awful in areas containing more missing points. With this in mind we then investigate the gap size in comparison to the error present in those gaps for each method.

Gappy SPOD, MF GPOD and Nearest Neighbor Comparison

The reconstructions from each methods can be seen in figures 2.6 and 2.7 for the u-velocity and v-velocity respectively. The snapshots presented contain the largest amount of missing data in the shear layer of any snapshot. Qualitatively these reconstructions look similar, with the exception of GPOD which has regions of zero velocity on the edge of the FOV in areas where black zones were originally located. It can be seen that while these gaps are the largest in the shear layer, the bulk of them reside above that region with with a few portions stretching downwards into that layer. This is a common characteristics among these larger gaps.

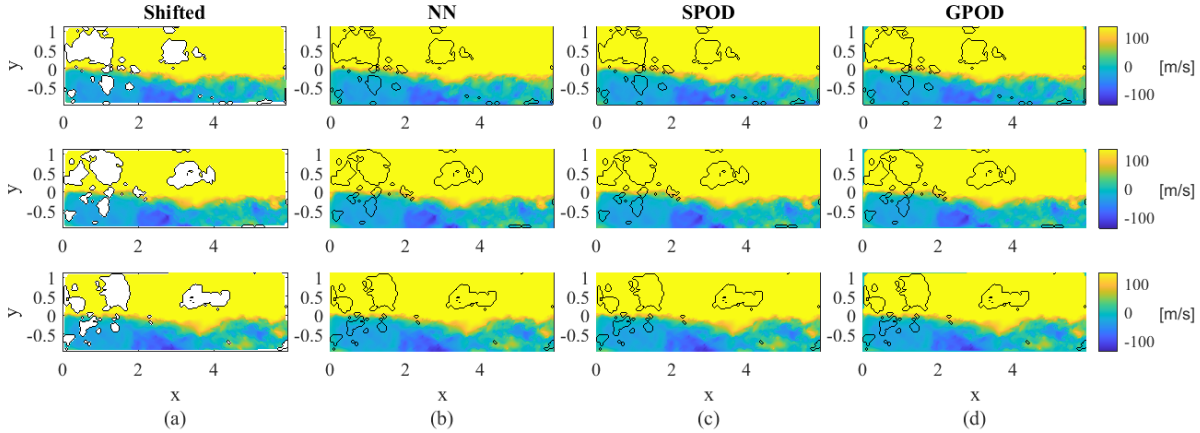


Figure 2.6. U-velocity snapshots of raw data reconstructions where large gaps are present in the shear-layer region (i=5646 [top] ,5647 [middle] ,5648 [bottom]). Shifted data (a), NN (b), SPOD (c) and GPOD (d) are shown with gappy regions outlined in black.

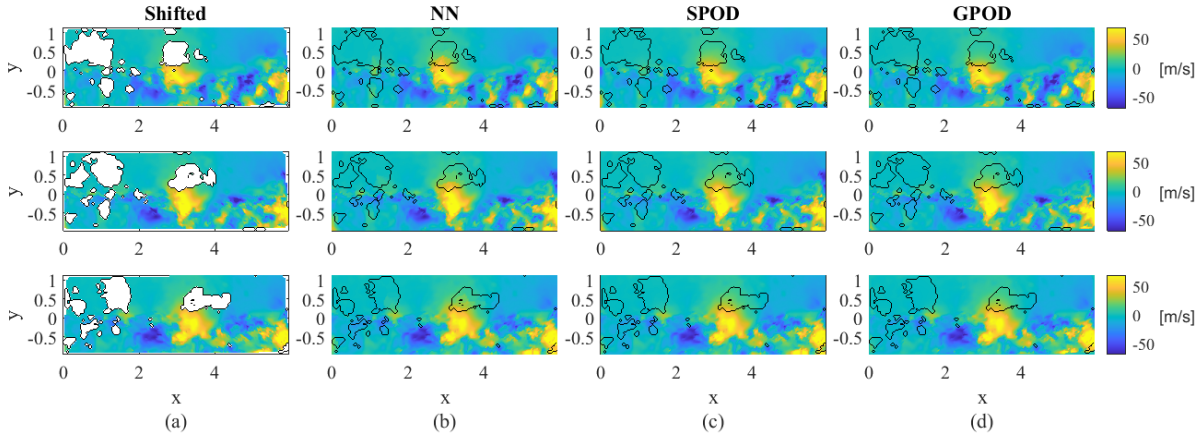


Figure 2.7. V-velocity snapshots of raw data reconstructions where large gaps are present in the shear-layer region (i=5646 [top] ,5647 [middle] ,5648 [bottom]). Shifted data (a), NN (b), SPOD (c) and GPOD (d) are shown with gappy regions outlined in black.

In order to provide a quantitative analysis of the methods, we compute the absolute error of every individual gap, as described by 1.20. Figure 2.8 displays this error as a scatter plot, with absolute error per gap on the y-axis and gap size on the x-axis. When looking at error as a function of gap size we can see that NN tends to be lower than the other methods, with SPOD being higher. Of particular note is the small cluster of extremely small errors in the lower end of the gap sizes, which is only exhibited by NN. There is significant variation in the errors for a given gap size for all methods, however we can see that the minimum error tends to be lower for NN and the maximum error higher for SPOD. While difficult to see in figure 2.8 as the gaps size increases to 10^3 and 10^4 this trend no longer continues and GPOD and SPOD are shown to have lower errors.

To better highlight this trend, we take the average error for each gap size and report the difference for all 3 combinations of the methods as shown in the following figure.

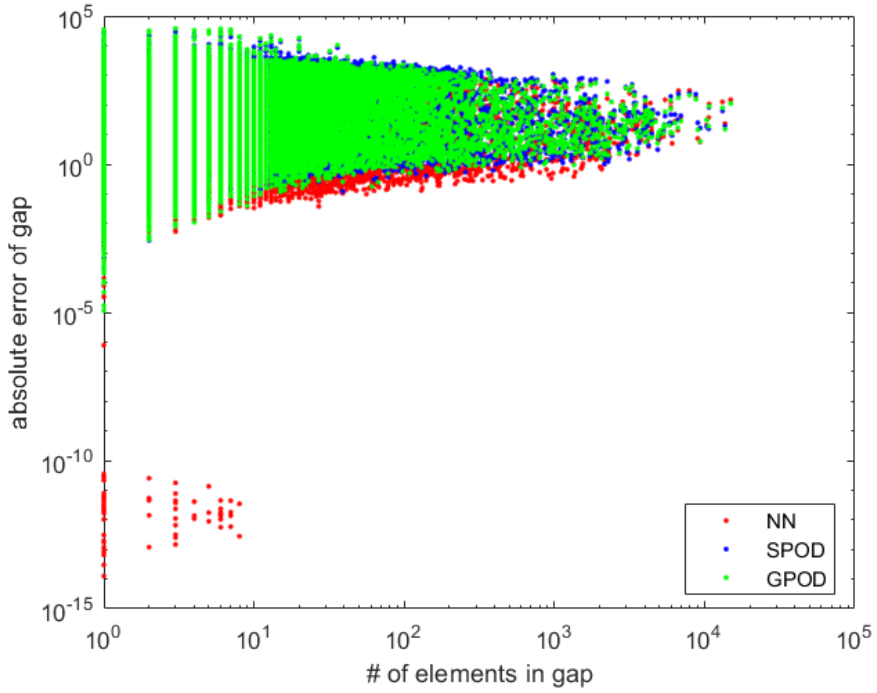


Figure 2.8. NN (red), SPOD (blue) and GPOD (green) absolute TKE errors are displayed for each gap in this scatter plot. Lower errors represent better reconstructions.

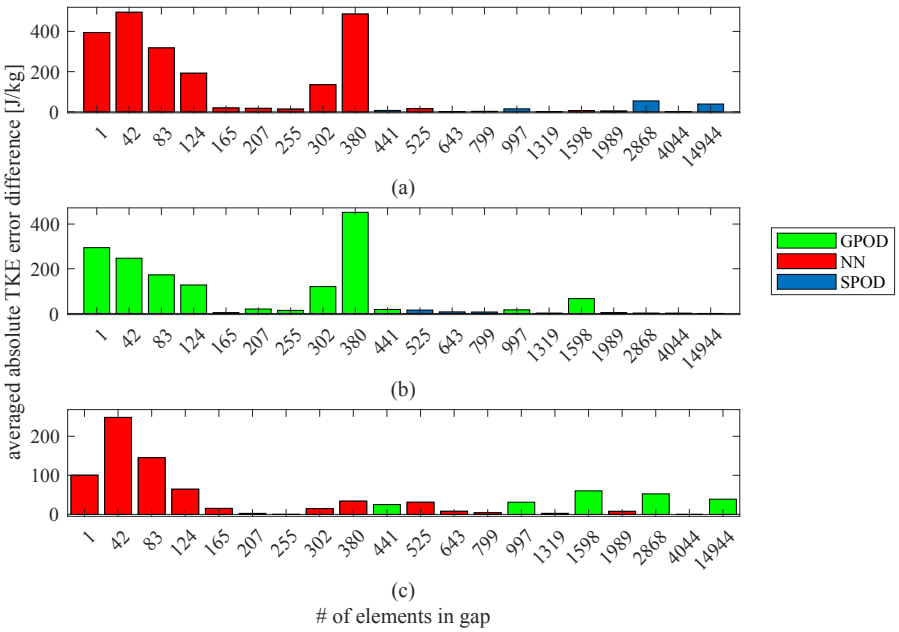


Figure 2.9. Difference in average absolute TKE error per gap size between each of the three methods: (a) NN/SPOD , (b) GPOD/SPOD and (c) NN/GPOD. Color indicates which method performs better: GPOD (green), NN (red), SPOD (blue).

Figure 2.9 (a) shows that the NN performs better in comparison to SPOD for most gap sizes, with a large difference in favor of NN for smaller gap sizes. NN can be seen to have errors in the range of 400 J/kg better than SPOD. However at larger gap sizes we see the difference between the two shrink and SPOD outperforming at the largest gap sizes. Similar trends are observed in (c) where NN outperforms MF GPOD, however the difference is less severe, with the greatest difference being slightly over 200 J/kg. In (b) we see MF GPOD outperform SPOD at most of the sizes shown, having error differences similar to those shown in (a). It should be noted that gap sizes represented in figure 2.9 are skewed towards those less than 10^3. These sizes for comparison were chosen by evenly sampling the recorded 780 unique gap sizes, where 534 or 68% of the unique gap sizes fall into the under 10^3 range. Next we shall look specifically at the larger gap sizes to better characterize the full extent of these methods.

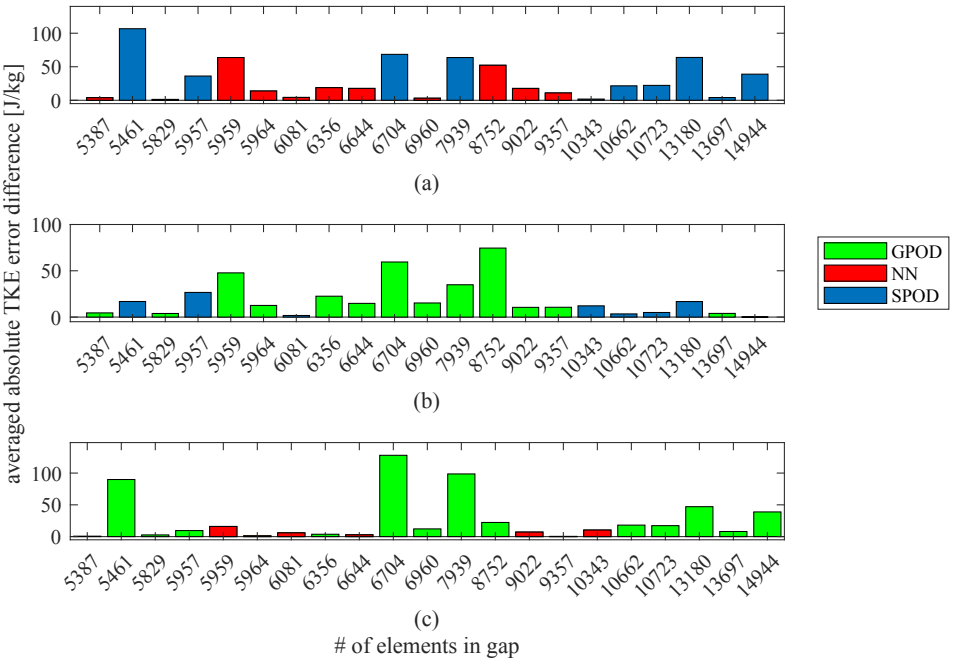


Figure 2.10. Largest 20 gap sizes with difference in average TKE error between the three methods: (a) NN/SPOD , (b) GPOD/SPOD and (c) NN/GPOD. Color indicates which method performs better: GPOD (green), NN (red), SPOD (blue)

In figure 2.10 the difference of the averages of the largest 20 gaps are shown. In contrast to the figure 2.9 we see NN to have errors that are closer to the other methods in these larger gaps. In (a) we can see SPOD having less average error then NN in some gap sizes, as noted by the blue bars. This is more consistent in gaps greater than 10^4, where SPOD is always shown to have lower error. It should be noted that the error differences here are smaller than those in the previous figure,as the largest difference is about 100 J/kg in SPOD's favor. When looking at (b) of figure 2.10 MF GPOD tends to outperform SPOD in large gaps until the gap size again grows to above 10^4 number of elements, Where SPOD enjoys a slight advantage. MF GPOD performs better in most of these larger gap sizes, however in many of the gaps there is shown to be only a slight difference, as seen in (c).

When looking at these results in combination we see NN has instances in which it outperforms both SPOD and MF GPOD by a wide margin, especially in the smaller gap sizes. NN also displays the lowest absolute global error. Since smaller gaps are the dominant type of gap seen in this data set,this begins to present a correlation between gap size and algorithm performance; where NN performs well in smaller gaps, while SPOD and GPOD performs better in larger gaps. To further explore this correlation we will need to examine the results from other data sets with different gap distributions. One such available data set is the previously test open-cavity flow with artificial gap insertion.

2.2 Artificial Gaps

In this section we return to the data set involving the artificial gaps used in[8], to compare the differences of the raw data set. This set has reference data that does not contain any gaps, making the gap shifting procedure unnecessary here. This section does not have a GPOD reconstruction comparison, but does include a comparison between the Gappy SPOD block-wise and gap-wise methods. An example of block-wise reconstruction for this data and the type of gaps present is shown in figure 2.11.

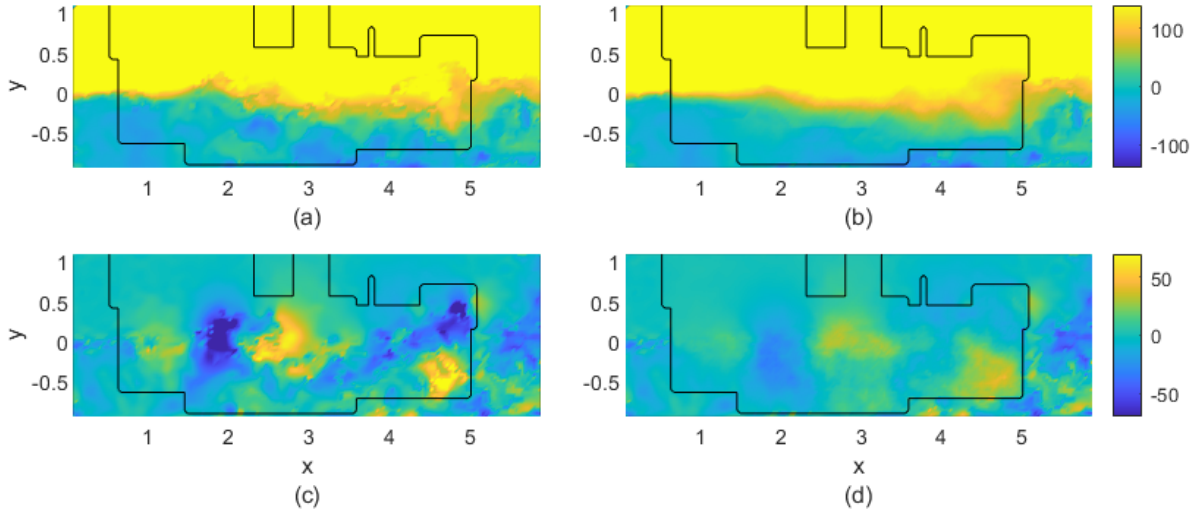


Figure 2.11. Block-wise reconstructions for open-cavity flow with artificial gaps. Shifted regions are shown for U (a) and V (c) velocity. Reconstructions (b) and (d) have color maps of $\pm 140 m\backslash s$ and $\pm 40 m\backslash s$ respectively.

2.2.1 Gap Statistics

A major difference between the artificial gaps and the raw gaps comes down to differences in size and occurrence. In the previous section we have shown that the raw data has 104537 gaps with the most common gap being single elements. There are only 43 artificial gaps that occupy 21.37% of the total number of elements available. There are no blackzones in this setup, as the domain has been truncated. The detailed breakdown is shown in figure 2.12. We see that the lowest gap size is on the order of 10^3 and the largest is on the order of 10^6, with the most common being in the 10^5 range; a vast departure from the raw gap characteristics.

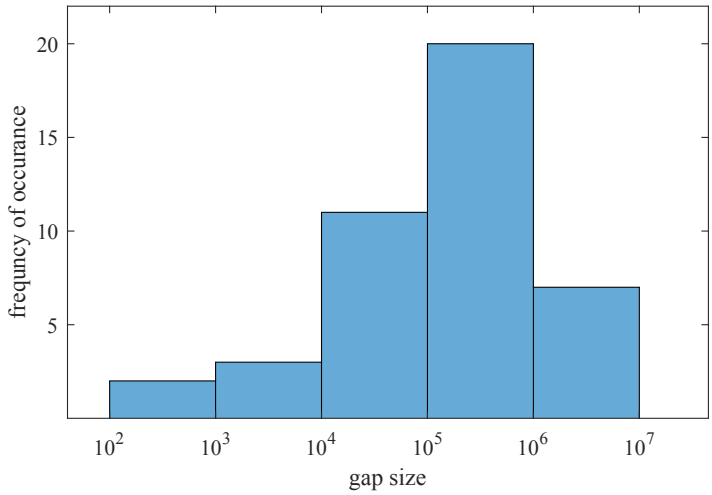


Figure 2.12. Histogram of artificial gap sizes. Frequency of occurrence is plotted along the vertical direction and gap size in number of elements is plotted along the horizontal direction.

2.2.2 Method Comparison

We would now like to present a comparison between Nearest neighbor, and gappy SPOD with both block-wise and gap-wise. Reconstruction for all these methods are displayed in a side-by-side comparison in figures 2.13 and 2.14. The large sections of missing data are made apparent in the artificial data frames, significantly larger and more structured than the raw gaps from the previous section. We then see the NN reconstruction struggle to produce qualitatively good reconstructions, with large streaks and artifacts from reconstruction obvious in the large gap regions. We then see very slight variations in the reconstruction between block-wise and gap-wise, where gap-wise tends to display a smoother shear line. This is most prevalent in snapshot 14319.

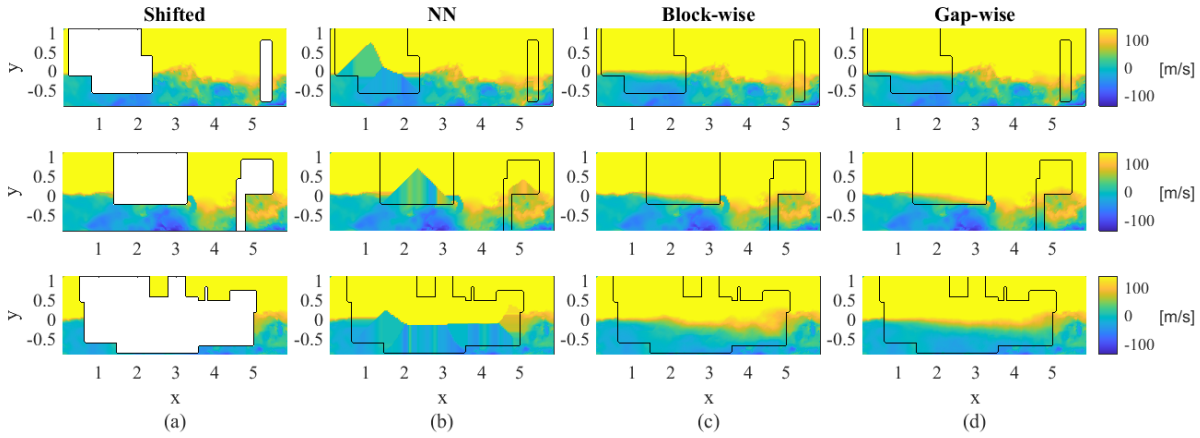


Figure 2.13. U-velocity snapshots of artificial gap reconstructions, where large gaps are present in the shear-layer region (i=1463 [top] ,8565 [middle] ,14319 [bottom]). Shifted data (a), NN (b), Block-wise (c) and Gap-wise (d) are shown with gappy regions outlined in black.

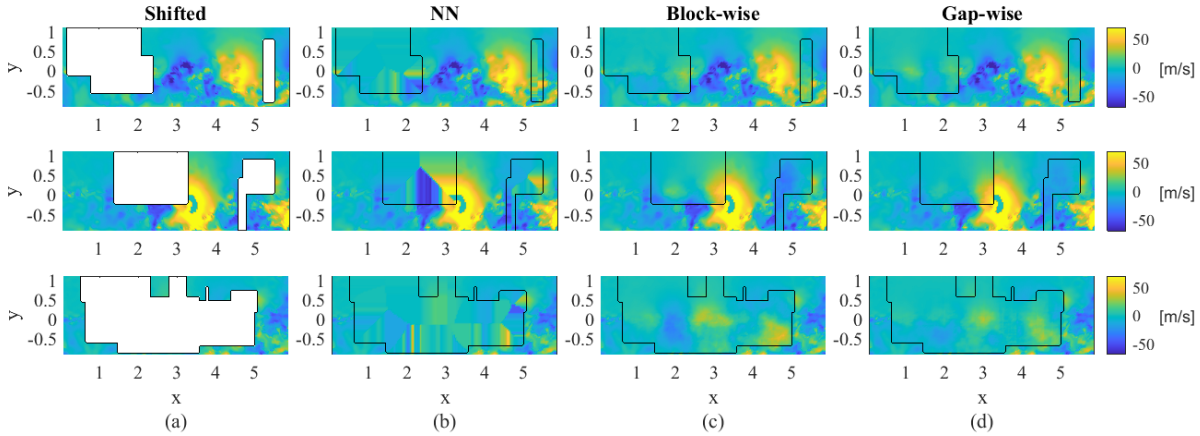


Figure 2.14. V-velocity snapshots of artificial gap reconstructions, where large gaps are present in the shear-layer region (i=1463 [top] ,8565 [middle] ,14319 [bottom]). Shifted data (a), NN (b), Block-wise (c) and Gap-wise (d) are shown with gappy regions outlined in black.

In the same 14319 snapshot, we see all methods struggle to reconstruct the v-velocity component, presenting less variations in the velocity about the shear line; as seen by the more muddled and less distinct coloring. Of course, this snapshot provides little information in the center that could be used for reconstruction, as the majority of the region is missing. This especially true of the gap-wise reconstruction. The same effect can be seen in the other snapshots but to a lesser degree.

Gappy SPOD block-wise reconstruction

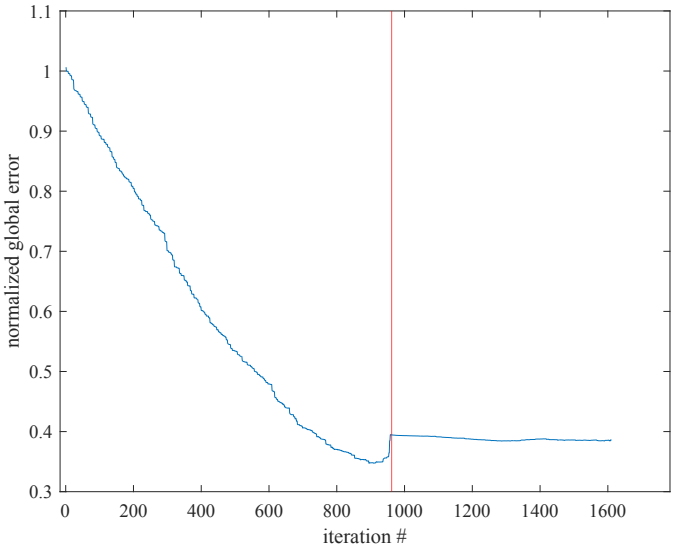


Figure 2.15. The normalized global error per iteration of block-wise reconstruction with NN initialization. The Red vertical line represents the next outer iteration

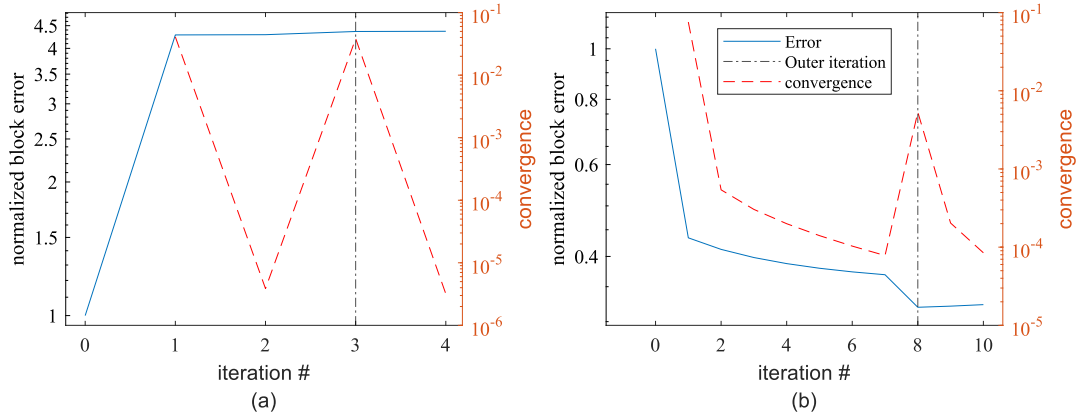


Figure 2.16. Block reconstructions with relative error and convergences vs iteration count. The block containing the fewest missing points (a)(1088) and the most (b)(633021) are shown.

In figure 2.15 we can observe a decreasing trend after the initial couple of iterations. In the first couple iterations we see the error increase from the nearest neighbor initialization but then steadily decrease, where it settles close to 40% of the initial error. There is a small observable increase in the error at the beginning of the 2^{nd} outer loop.

In figure 2.16 reconstruction convergence and relative errors are shown for the blocks containing the most (b) and least (a) amount of gaps of all blocks. The left example (a) contains the least amount of gaps; we can see the block-wise method struggle to reconstruct this region more accurately then the NN initialization, ultimately converging to an error that is $4\times$ the initial. This convergence criteria is achieved after one iteration in both global loops, as denoted by the dotted-dashed line.

The block containing the most missing elements (b) exhibits the opposite trend, showing a significant error reduction from the initialization. We see that it requires more iterations to achieve this, but much of this improvement occurs during the first outer loop. Gap-wise reconstruction errors and convergence can be found in A.1.

In comparison to the raw gaps we see a better performance from the block-wise method in the global relative error and relative block error. To complete the comparison we now look at the individual gap errors.

Blockwise, Gapwise and Nearest Neighbor comparisons

Table 2.5. Absolute global errors of Artificial gaps

Method	Absolute Global error
NN	9.6215×10^6
Gapwise	2.9807×10^6
Blockwise	2.5148×10^6

We present the global errors of each method in table 2.5, where it is shown that block-wise reconstruction results in the overall lowest global error, with Nearest neighbor interpolation exhibiting the worse performance. Gap-wise presents and error that remains very close, though slightly worse, to the block-wise method.

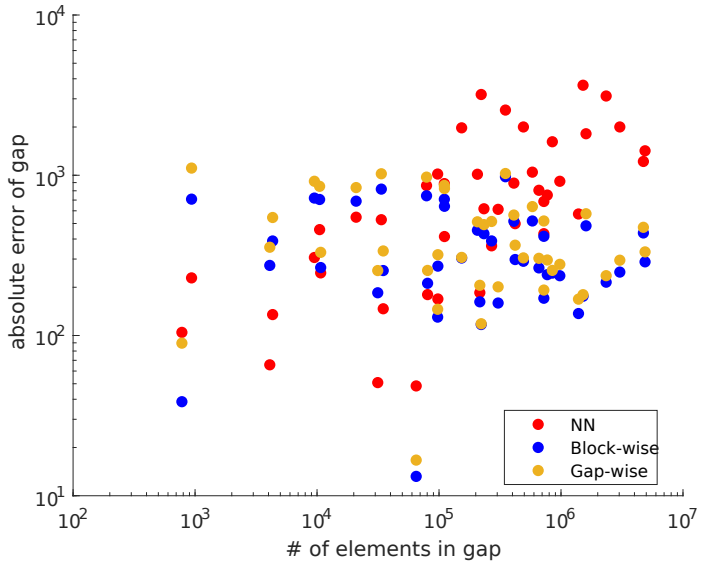


Figure 2.17. Scatter plot of relative gap error versus gap size in # of elements. Nearest Neighbor (red), Block-wise SPOD (blue) and gap-wise SPOD (orange) are all shown.

In order to know more about how gap size affects reconstruction, we once again look at a scatter plot of error vs gap size for these artificial gaps to view any relevant trends. In figure 2.17 we that the gap wise method and block wise method remain relatively close in error with block wise having a slight advantage. Nearest Neighbor remains slightly below both of these methods until around 10^5 elements where it starts to have consistently higher errors than either method.

In figure 2.18 a bar chart for Nearest neighbor and block wise SPOD is shown in a similar fashion to the charts shown earlier. It displays the average absolute TKE error per gap size, however this is a bit of misnomer as there was not enough repetition in gap size to provide an average. We see that block wise SPOD has a lower error most samples, especially as the gap size increases.Note that the absolute errors provided here are magnitudes larger here than in the previous section, however the gappyness of this data is about 19.7% greater than the previously presented data.

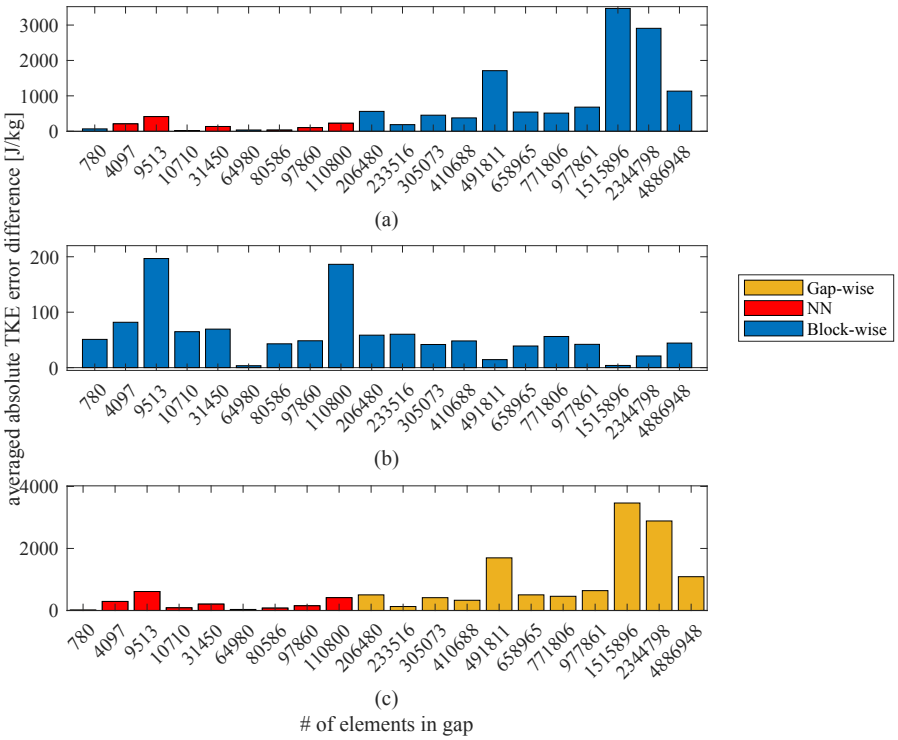


Figure 2.18. Difference in average absolute TKE error per gap size between each of the three methods: (a) NN/Block-wise , (b) Gap-wise/Block-wise and (c) NN/Gap-wise. Color indicates which method performs better: NN (red), Block-wise (blue), Gap-wise (orange)

It is also important to note the scale of these errors, where the SPOD methods produce errors that are better by thousands of J/kg in the largest gap sizes. This is a magnitude greater than the errors displayed in the raw open-cavity data. There is once again a correlation present between the individual methods and the size of the reconstruction area, with SPOD working better in larger regions.

This will conclude our analysis of the open-cavity data sets and we shall now transition to the high-lift data sets to determine if these trends continue.

Chapter 3

Results: High-lift flow

We now move into the final data set available for comparison. This data was produced by Zhang and Cattafesta using a flap from an airfoil in windtunnel testing and two Phantom V2012 high-speed cameras equipped with 180 mm Tamron SP Di Macro lenses, Scheimpflug adapters, and 532 nm band-pass filters to acquire the PIV measurements [24]. An image of the schematic taken from this paper can be seen in figure 3.1, and a schematic of the airfoil is available in A.4.

Parameters used for reconstruction in table 3.1. Figure 3.2 presents an example of this reconstruction, showing both U- and V- velocities. .

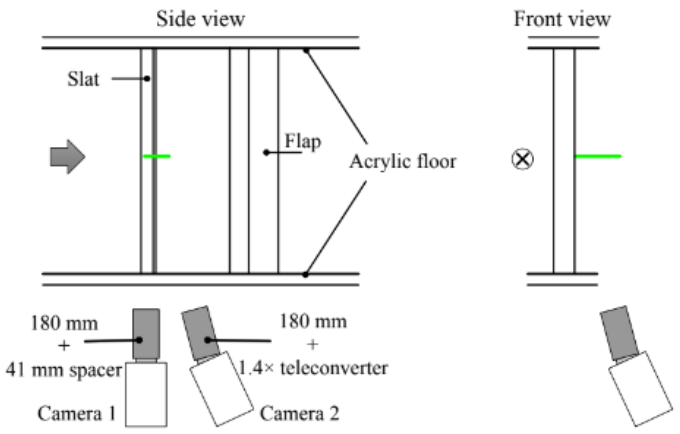


Fig. 6 Schematic of PIV setup (not to scale).

Figure 3.1. Schematic of High lift experimental setup taken from [24],[22]

Table 3.1. Parameters for reconstruction

N_x	N_y	N_t	Δt	n_{fft}	n_{ovlp}	n_{blk}	tol_{inner}	$gap_{ovlpmax}$
238	135	11000	9×10^{-5}	128	64	170	10^{-4}	20%

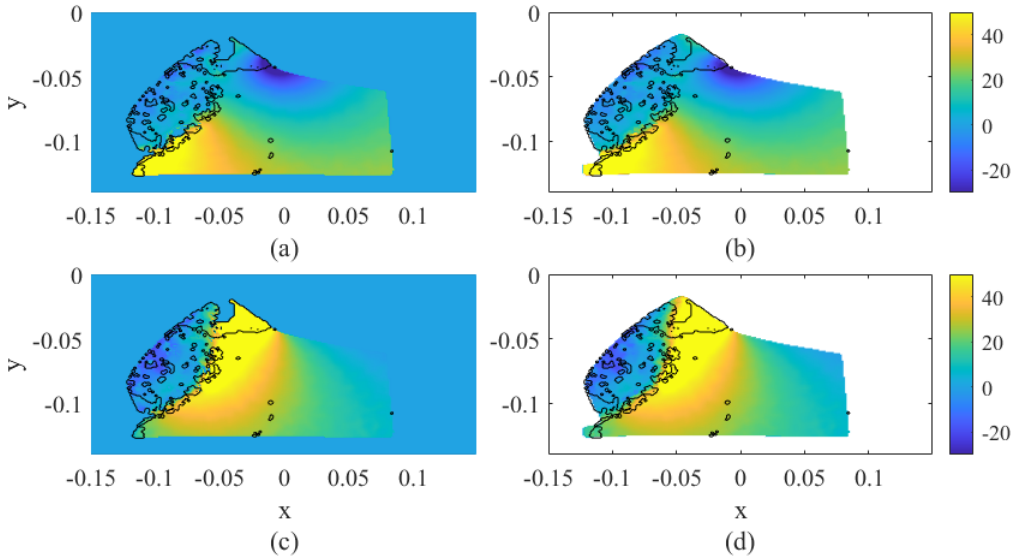


Figure 3.2. Example of block-wise reconstructions for data with artificial gaps. Reference data is shown in (a) and (c) for U- and V-velocity respectively.(b) shows U-velocity and (d) shows V-velocity reconstruction with color maps of $-30, +50$

3.0.1 Gap Statistics

Table 3.2 gives a breakdown of the types of gaps existing in this data set. Immediately we can see that this data contains a much large percentage of missing values at 70.35%. However 92.77% of these missing values are black zones leaving only 7.23% of these values, or 5.08% of the total data, for reconstruction. This amount then reduces even further after shifting the gaps, leaving us with 1.2977×10^7, 3.42% of all data, as points for reconstruction comparison.

Table 3.2. Breakdown of gappyness percentage

	total # of elements	# of missing elements	% gappyness
Non-Black zones	3.7961×10^8	1.7965×10^7	5.08%
Black zones	3.7961×10^8	2.3067×10^8	65.27%
Overall	3.7961×10^8	2.4863×10^8	70.35%

Once again we present a histogram detailing a breakdown of the frequency of gap sizes in figure 3.3. Gap sizes of single elements are the most common as with the raw open-cavity data, however the largest gap in this data set remains slightly below 10^7; a gap size much larger than the one shown in the raw open-cavity flow.

The breakdown of these sizes versus the data set and other gaps are displayed in table 3.3. This table does not account for black zones or losses from shifting gaps. We can see that the largest gap accounts for 4.7% of the total data set yet makes up a very small percentage of the total 230629 enumerated gaps. With this information in mind we shall now proceed to the method comparison section.

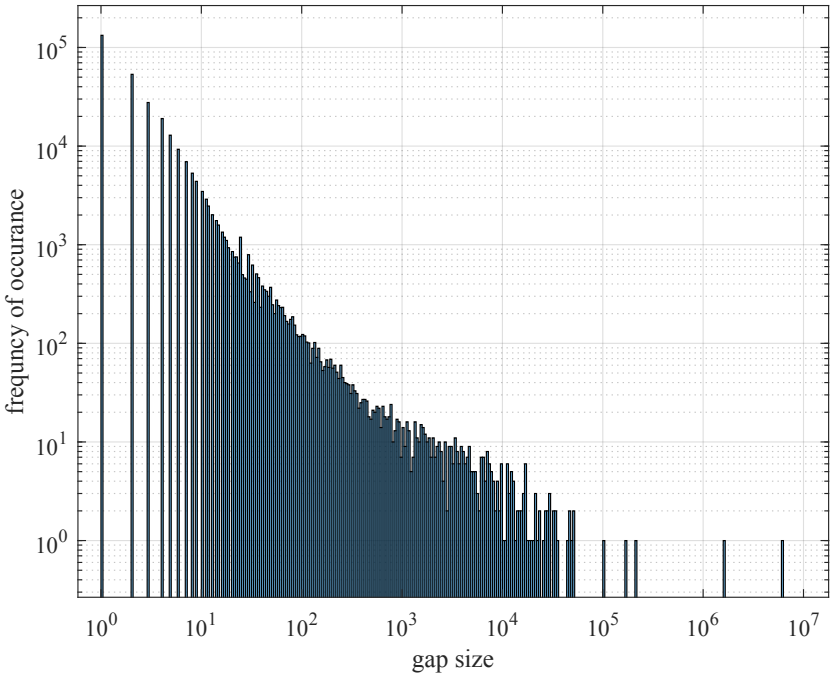


Figure 3.3. Histogram of high-lift gap sizes versus their frequency plotted on a log-log plot after gap shifting. Gap size is the number of elements that comprise a single gap.

We contextualize the presence of these gaps by displaying their percentages against the total number of elements in the overall data set and against the number of gaps found in table 3.2. We see that while small gaps of the order 10^0 to 10^1 make up almost 90% of the gaps found, yet do not represent a large portion of the data set. The large gaps make up 2.23% of the entire

data set, despite being the fewest types of gaps found. Let us continue on to examining the errors of the methods to determine if there are any trends present that are similar to previous results.

Table 3.3. Percentage of gap sizes in terms of number of gaps and total elements

gaps size	% of gaps	% of total elements
10^0	43.33%	3.502^{-2} %
10^1	46.45%	0.15%
10^2	9.35%	0.19%
10^3	0.72%	0.15%
10^4	1.3%	0.33%
10^5	2.119×10^{-4}%	0.36%
10^6	9.780×10^{-4}%	2.21%

3.0.2 Method Comparison

Table 3.4. Global Absolute error for high lift data set

Method	Absolute Global error
NN	1.1133×10^5
GPOD	8.1063×10^4
SPOD	1.3161×10^5

We shall begin by examining the global error from each of the three methods as presented in table 3.4. MF GPOD appears to have the lowest overall global error, and SPOD having the overall highest, with NN and SPOD presenting similar error values. In figures 3.4 and 3.5 we see the reconstructions of the high lift device flow field with all methods. The blackzones have all been filled with zeros in the GPOD reconstructions, causing the blue background in the images. We have decided to not extrapolate to blackzones in these reconstructions, to make the region of reconstruction clearer. There aren't any obvious qualitative differences in the reconstruction images.

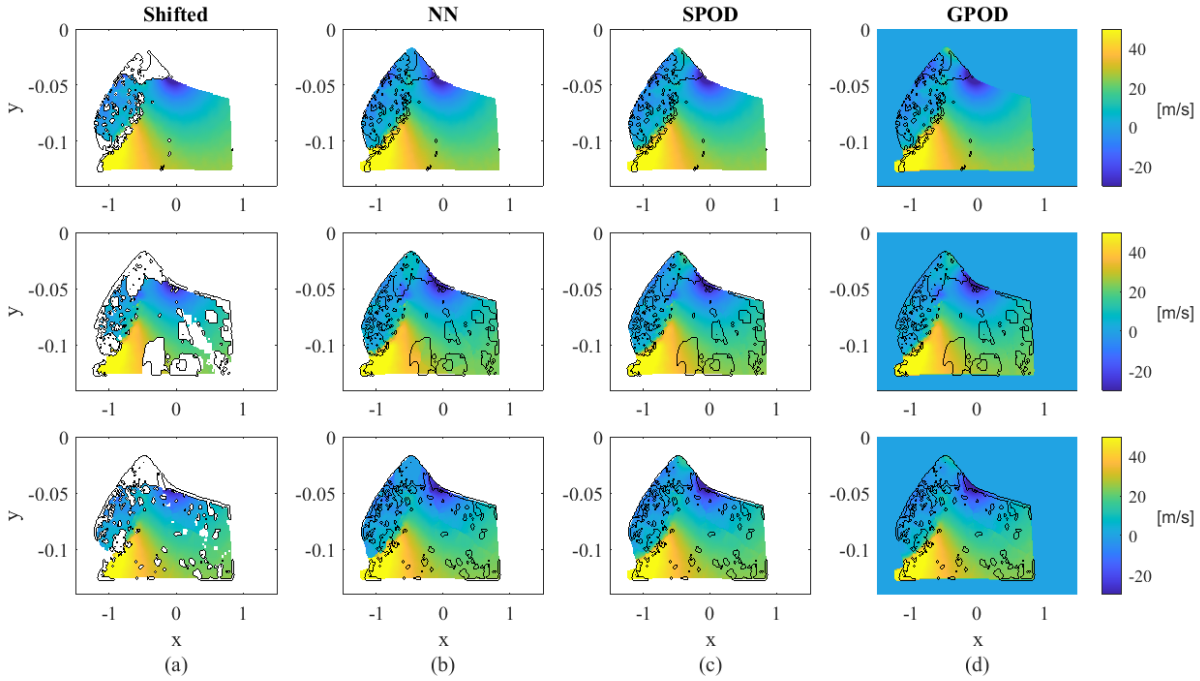


Figure 3.4. U-velocity snapshots of high-lift data reconstructions where large gaps are present in the shear-layer region (i=1019 [top] ,3224 [middle] ,5014 [bottom]). Shifted data (a), NN (b), SPOD (c) and GPOD (d) are shown with gappy regions outlined in black.

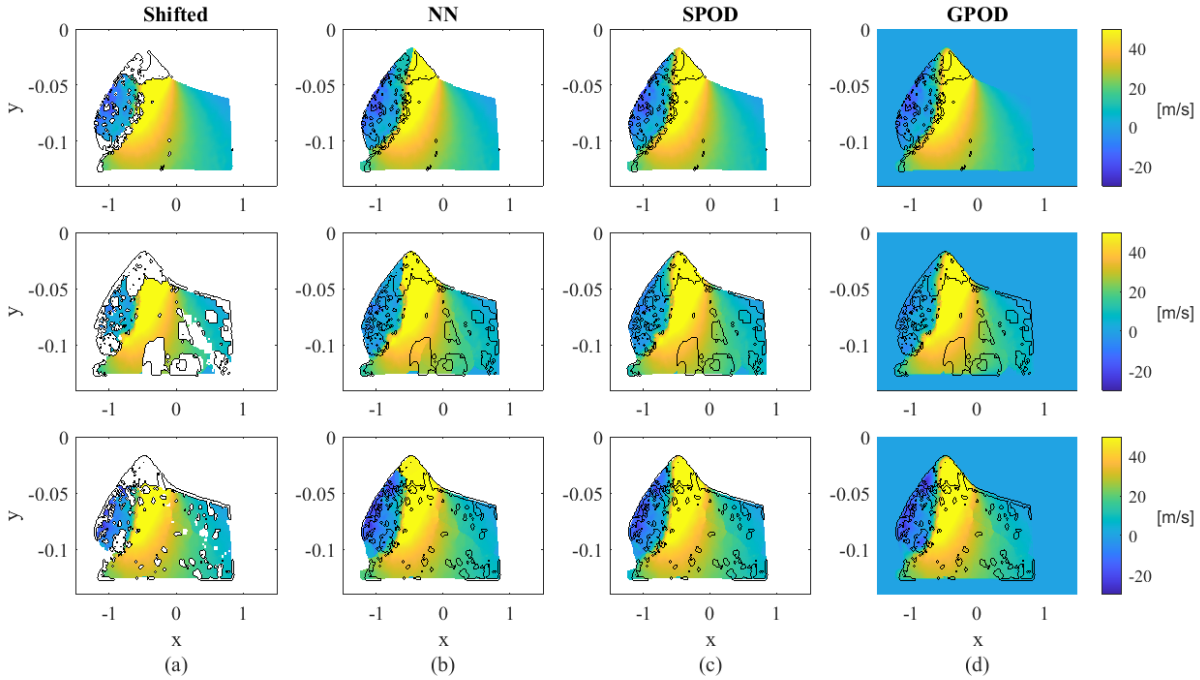


Figure 3.5. V-velocity snapshots of high-lift data reconstructions where large gaps are present in the shear-layer region (i=1019 [top] ,3224 [middle] ,5014 [bottom]). Shifted data (a), NN (b), SPOD (c) and GPOD (d) are shown with gappy regions outlined in black.

Gappy SPOD Blockwise reconstruction

We present the errors per iteration and block convergence examples to help quantify the performance of the block-wise algorithm on this data set. In figure 3.6 we can see the error plotted against iterations. This figure displays an overall increasing trend for the error as iteration continues on. The final error settles near the maximum error the data is normalized around. The error begins to drop once the 2^{nd} outer iteration has begun but begins to plateau, experiencing a slight increase during its final iterations. This trend tells us of the overall advantage NN reconstruction has in this data set, based on the initialization value. Since the previous work on the open-cavity flow did not present any benefits for multiple outer loop iterations we reduced the number of outer loops from 4 to 2 for this data set. This reduces the number of total iterations, but convergences criteria are still reached, as shown in the next figure. We reduced the tolerance requirement here for the same reasons.

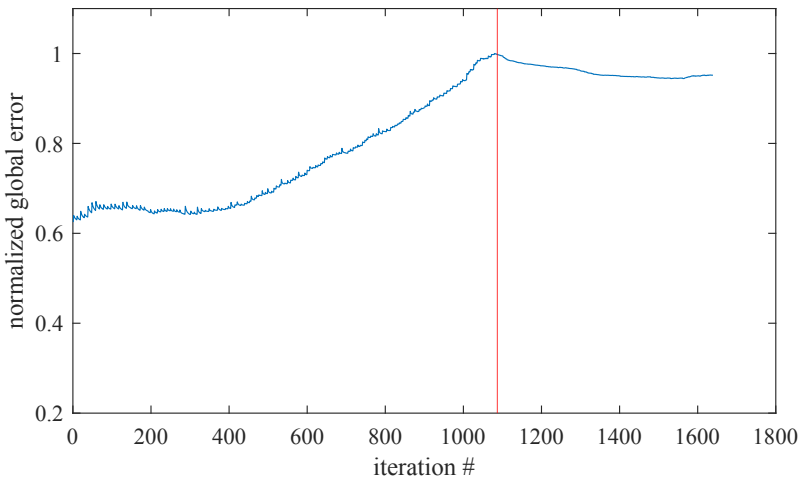


Figure 3.6. Error of block wise reconstruction vs iteration number for high lift data set. The red vertical line represents the outer loop.

In figure 3.7 we see the error and convergences of two blocks that represent the minimum and maximum number of missing points found in a single block. In the block with minimum missing points (a), the error plateaus almost immediately at double the initialization error. We can see a sharp peak in the convergence of this block, similar to those seen in previous sections,

which corresponds to the the outer loop iteration.

For the maximum number of missing points (b), the error quickly rises to above four times the initial and then slowly decreases down to around 2 for the first outer iteration. During the second outer iteration the error is reduced well below two before reaching the tolerance criteria.

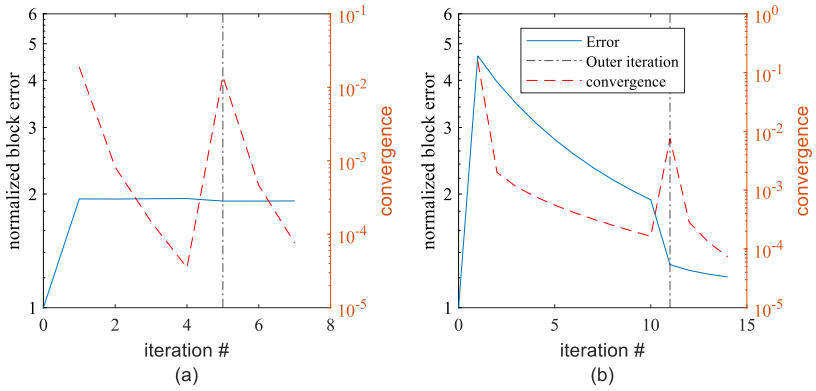


Figure 3.7. Examples of individual block reconstructions. (a) is the block containing the fewest amount of missing points (39027) and (b) is the block containing the most (455967)

This follows the trends of SPOD having better performance in blocks containing larger amounts of gappy data shown in the previous sections.Next we shall look at the error in each individual gap size.

Gappy SPOD, MF GPOD and Nearest Neighbor comparision

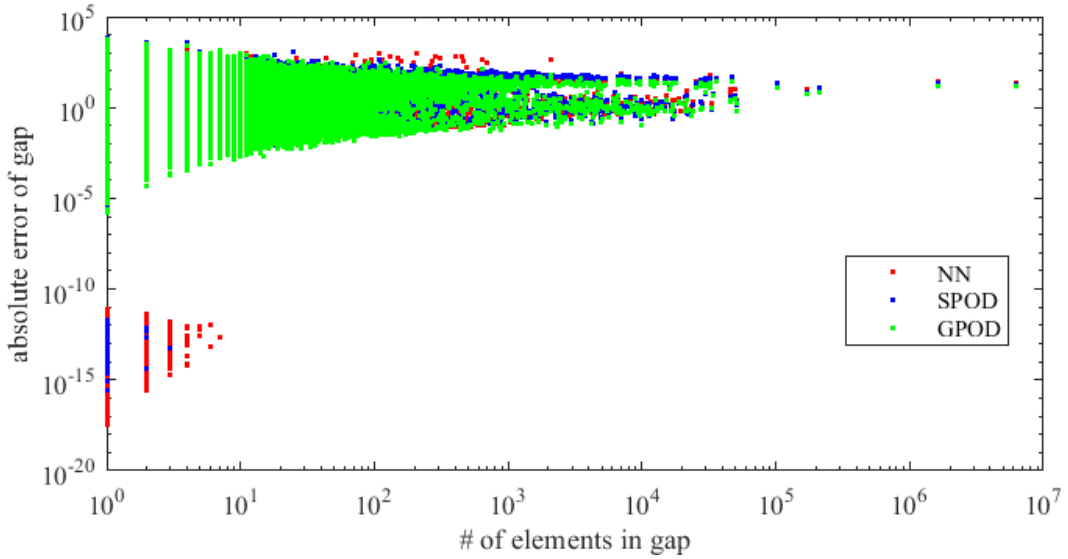


Figure 3.8. Scatter plot of gap error vs. gap size in number of elements. SPOD (blue), NN (red) and GPOD (green) are shown.

Figure 3.8 shows the errors for each method in terms of absolute error per gap size. MF GPOD is consistently lower than the other two methods, except for the clustering of low errors in the 10^0-10^1 range. This clustering appeared before in the raw open-cavity data, but exhibited this trend only for NN. In this case, SPOD errors can be seen in this cluster as well, though to a smaller degree. NN is actually shown to have higher errors in 10^2 to 10^3 unlike in previous data sets.

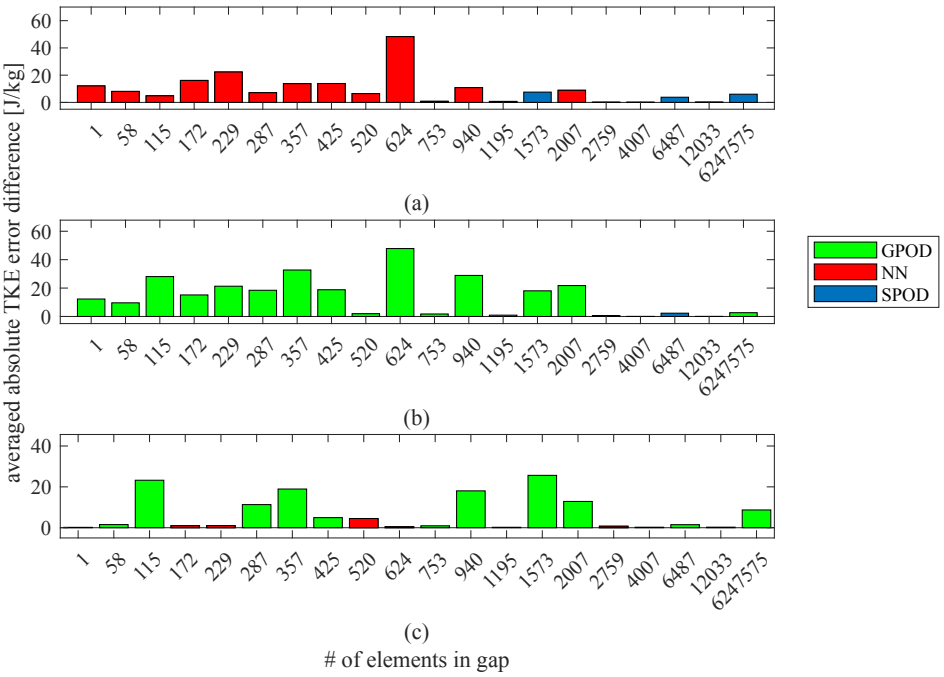


Figure 3.9. Difference in average absolute TKE error per gap size between each of the three methods: (a) NN/SPOD, (b) GPOD/SPOD, and (c) NN/GPOD. Color indicates which method performs better:GPOD (green), NN (red), SPOD (blue).

To help distinguish performance according to gap size, we once again take an average of the error as shown in figure 3.9. This figure shows GPOD to be more accurate in almost every gap size in comparison to both SPOD and NN. The difference between GPOD and SPOD is greatest at about 50 J/kg in favor of GPOD; whereas the greatest difference between GPOD and NN is about 30 J/kg in favor of GPOD. These errors are all smaller then those present in the open-cavity flow. The most common type of gap, single missing element gaps, is shown to have almost no difference between GPOD and NN as seen in (c). When comparing NN and SPOD (a), NN performs better in the smaller sizes, but soon loses this advantage in the larger size. Once again smaller gaps make up a greater portion of the overall recorded gap sizes and is evident when evenly sampling through the list of recorded gap sizes. To balance our analysis we will once again compare the 20 largest gaps.

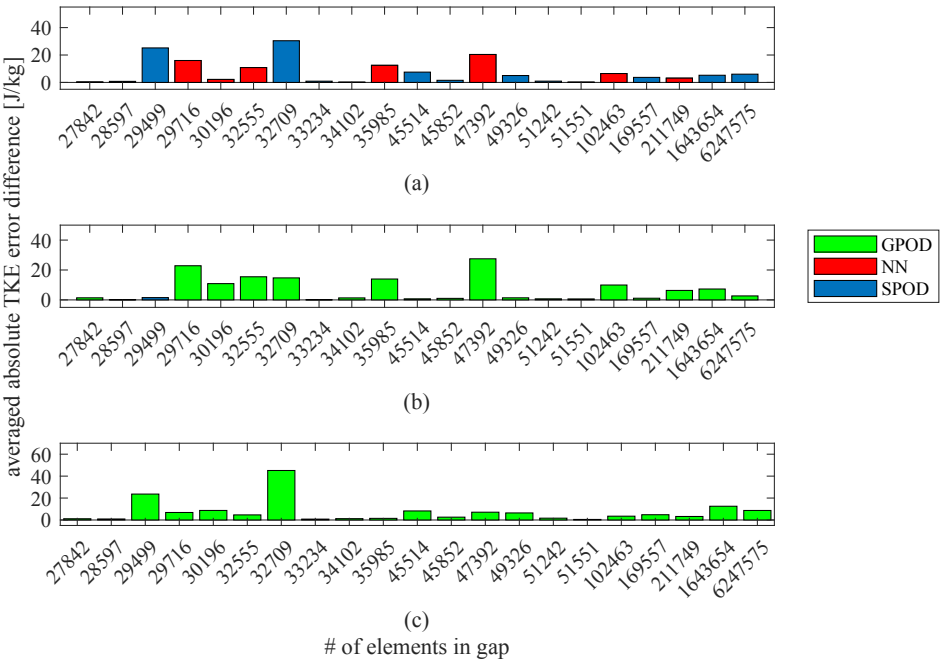


Figure 3.10. Largest 20 gap sizes with difference in average TKE error between the three methods: (a) NN/SPOD , (b) GPOD/SPOD and (c) NN/GPOD. Color indicates which method performs better: GPOD (green), NN (red), SPOD (blue)

In figure 3.10 the differences in the errors between these methods do not exhibit obvious trends as before, with the error differences only slightly favoring one method over the other in many gap sizes. We do see GPOD in (b),(c) having the most notable favorable difference in error in these larger gaps. The comparison between SPOD and NN (a) is not as decisive as was shown in previous data.

The high-lift data does not follow many of the trends shown in the open-cavity flow data and sees much better reconstruction by MF GPOD on every level, despite some areas of close similarities with the other methods. This concludes our analysis of all data sets and we shall begin a more in-depth discussion of these results in the next chapter.

Chapter 4

Discussion

Reviewing the results presented in this thesis we see NN perform significantly better than expected in all data sets. It performs best in the raw open-cavity data but worst in the artificial gaps. For SPOD, the opposite is true. The sees the best reconstruction with MF GPOD, while NN and SPOD are shown to have similar reconstructions. This does present a possible correlation between the size of the gap and performance of the reconstruction method. A possible consideration for this correlation is the size of the FOV. For the cavity flow data the FOV consists of point grid sized at 160×57 which equates to $9,120$ distinct points for consideration. It is unlikely that a gap completely covers this entire area during a single snapshot, so larger gaps will tend towards extending throughout multiple snapshots. Since a major facet of SPOD is to compute correlated modes in space and time, it could have an advantage over other algorithms reconstructing gaps found spread over the time domain.This relies on a few assumptions: the method in which we groups is accurate and meaningful in determining the reconstruction performance, and that gap shape and temporal/spatial distribution is a key factor during reconstruction.

However the high-lift data does not support this reasoning as much as the open-cavity, otherwise we would expect to see SPOD have the best reconstruction error in the largest gap sizes. This would imply that there are other factors at play, possibly involving the location of the gap and the surround flow properties. For instance, many of the gaps tend to be located

higher above the shear line. This regions has a more uniform flow characteristic , as opposed to the more turbulent and chaotic nature of the flow around the shear line, which could make reconstruction easier in that location.

When deciding the method in which to group gaps we selected a simplistic approach allowing any connection to serve as justification for grouping points together. As shown in the 1.3.1 with figure 1.4 this could group points together with weak connections (i.e single diagonal connections), which could be inflating gap size statistics.

Black zones could also contribute this idea if not handled properly. Black zones are not discussed widely throughout this paper, mainly due to the lack of reference data. When performing the reconstruction we have only removed true black zones that are missing in all frames, however there is still the possibility of ”approximate” black zones in the data set, essentially missing values that appear in almost every snapshot. It is entirely possible for a spatial point to missing in all but one snapshot. These do not affect the errors since initial missing points are filtered out when comparing the shifted data, however they could connect gaps that would otherwise be unconnected. The data sets used in this paper do not contain such close approximate black zones, however areas missing in 98% of snapshots do exist which equates to missing in 15680 snapshots. It is best left to the researcher to determine the validity of the points occurring in these mostly missing locations of their data sets and to handle these approximate black zones accordingly. It would require more data sets and testing to determine if these factors mentioned greatly influence the gappy SPOD reconstruction capability

When it comes to the differences between gap-wise and block-wise gappy SPOD we can observe the block-wise method being consistently more accurate then the gap wise. So it is my recommendation to use block-wise as it is more versatile in reconstructing data, by requiring less time to reconstruct a large amount of gaps then gap-wise would.

A surprising outcome was the effectiveness of Nearest Neighbor. Nearest Neighbor has outperformed GPOD and SPOD more often then expected. We see Nearest neighbor perform better globally in the raw open-cavity flow data and better than the SPOD in the high lift device

flow. The only data set Nearest neighbor was the worst performer, occurred in the artificial gaps placed in the open-cavity flow. The better global performance of Nearest Neighbor for the raw open cavity flow, could be attributed to its greater percentage of small element gaps. Gap size on the order of 10^0 comprised 9.07% of the data, whereas gaps on the order 10^4 only comprised 2.98% of the total element count. This contrasts with the high lift where 10^0 sized gaps only comprises 0.14% of total elements. An important aspect to note is that in both of these data sets 10^0 gaps were the majority of regions to be reconstructed, making up 84.31% and 89.85% of the gappy regions in each data set respectively. This implies that Nearest Neighbor performs better when the majority of reconstruction targets are small. As mentioned earlier the location of these gaps could influence reconstruction performance as well. It is more likely that NN performs better in uniform flow where velocities are similar in a local region. We also see that Nearest neighbor performs worse during large gap reconstruction with obvious artifacts from reconstruction, which is observable in the artificial gap reconstructions. This makes Nearest Neighbor a prime candidate to be used in conjunction with more complex reconstruction methods to help offset runtime and performance in small gap region reconstruction.

MF GPOD served as a middle ground in terms of the reconstruction methods. In the open cavity it performed worse than Nearest Neighbor in both global error and in most of the small gap reconstructions. In many instances of the same data it performed better than SPOD in smaller gaps and in some larger gaps, however it consistently struggled to outperform SPOD once the gap size reached greater than 10^4, sometimes the difference being small. This performance difference could be related to the differences in the spectral and temporal relations of the methods. However the high-lift data is best reconstructed by MF GPOD and this could be due to the flow physics inside of the data itself and how it influences the gappy regions. One important difference to mention is the ability to handle black zones. The SPOD code has been modified to account for them, but the GPOD code does not appear to be, as those regions appear as zeros.

As of now it is difficult to predict the factors that will determine when one methods will outperform the others. There are some trends correlating gap size and methods performance, but

this is only one of possibly many confounding factors. More research into these relationships will be required to fully understand and improve these reconstruction algorithm applications. For now we take the information presented here and will propose some best practices until more research is conducted.

4.0.1 Conclusions

Throughout this thesis the missing points from raw PIV data has been characterized and their effects on reconstruction methods have been analysed. Through this analysis we have found that the majority of missing points are small gaps and MF GPOD and gappy SPOD do not perform as well as Nearest Neighbor interpolation in these areas, however gap majority does not mean missing element majority. As such, in sets where large gaps contain the majority of missing elements, methods (GPOD and gappy SPOD) that are better at reconstructing large gaps have lower global error. Based on the results presented in this thesis We would now recommend a new procedure for PIV data reconstruction:

1. Remove black zones
2. remove approximate black zones if determined necessary
3. run a gap-finding algorithm that stores gap sizes
4. initialize the data set with Nearest Neighbor for reconstruction
5. Use block-wise gappy SPOD to reconstruct gaps of size 10^3 and greater for open-cavity flow and use MF GPOD for flow similar to the high-lift device
6. Once finished with reconstruction choose to interpolate black zones or leave them as missing.

This procedure is based on a limited number of data sets and more testing to further understand the best operating conditions for each method presented would enable greater reconstruction capabilities for the future.

Appendix A

Appendix

A.1 Gap-wise Reconstruction of Open-Cavity Flow

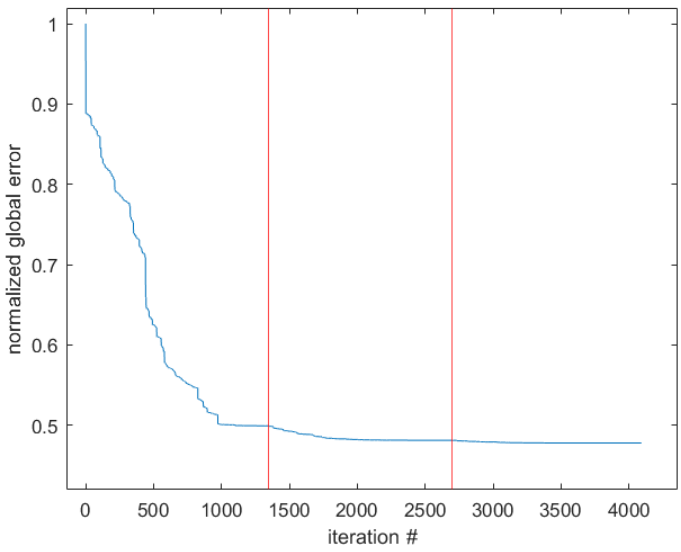


Figure A.1. The normalized global error per iteration of block-wise reconstruction. The error is normalized the maximum value. Red vertical lines represent outer iterations

Figure A.1 displays the error as a function of iteration number for the gap wise reconstruction of the artificial gaps in the open cavity flow. This does not directly match the results produce in Nekkanti and Schmidt [8]. The reason for this is a difference in parameters and initialization methods. Gaps were originally initialized using the mean flow, but nearest neighbor produces a more accurate reconstruction. This causes the value used for normalization to be

closer to the final error value, hence having a higher ending error percentage.

A.2 Raw Cavity Data

Below in figure A.2 we can see a breakdown of the gap sizes versus frequency before shifting and overlapping gaps are taken into account. We can the gap sizes having a higher range with a single gap having 10^6 elements in comparison to the 10^4 maximum present after shifting.

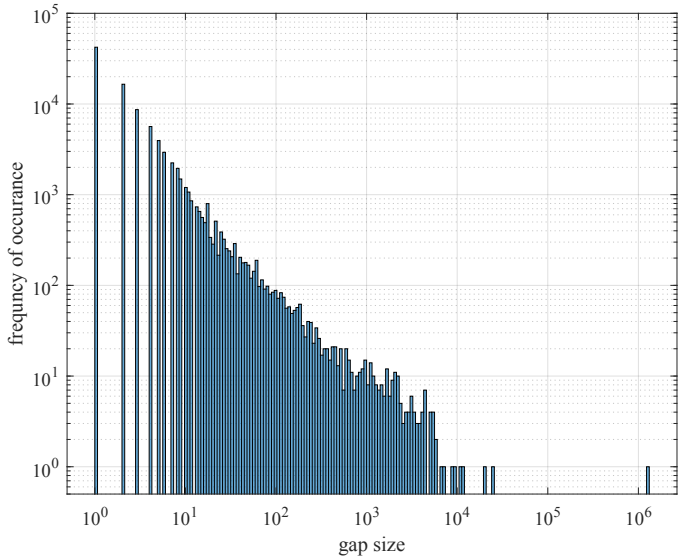


Figure A.2. Histogram of raw gap before shifting. Gap size is determined by the number of elements that comprise a single gap

A.3 High-Lift Data

A.3.1 Gap Information

Figure A.3 shows a breakdown of the gap sizes that occur in the raw high lift data set before shifting and overlap removal. These gaps have more extreme sizes, with larger maximum gap sizes and more 10^0 gaps

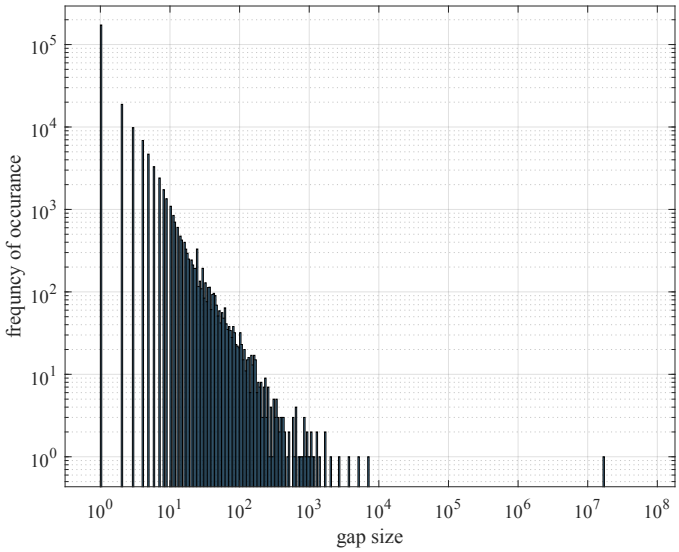


Figure A.3. Histogram of high-lift before shifting. Gap size is determined by the number of elements that comprise a single gap

Following this figure is a breakdown of the gappy percentages in table A.1

Table A.1. gap sizes and percentage of overall gaps and total elements

gaps size	% of gaps	% of total elements
10^0	89.85%	0.14%
10^1	9.82%	0.14%
10^2	0.32%	4.323×10^{-4}%
10^3	9.9727×10^{-5}%	1.5185×10^{-4}%
10^5	4.3360×10^{-6} %	3.8254×10^{-4}%
10^7	4.3360×10^{-6} %	4.70%

A.3.2 Airfoil

In figure A.4 the parameters of the airfoil used to produce the data is shown along with a schematic of the airfoil itself.

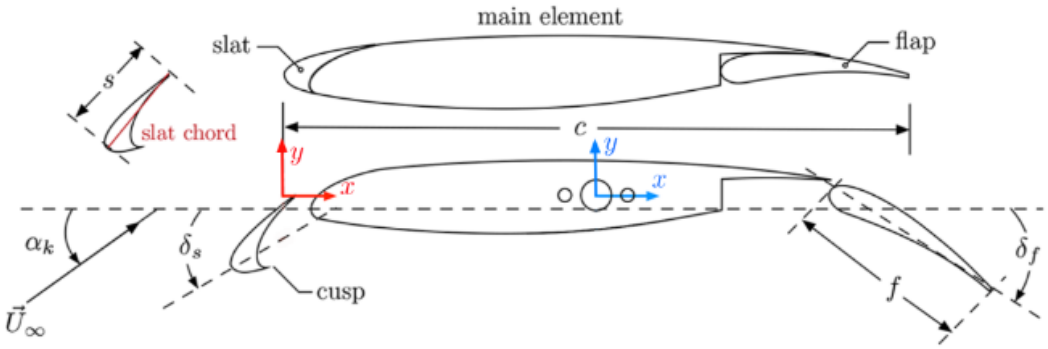


Fig. 1 Schematic of the 30P30N multielement airfoil.

Table 1 Primary dimensions of 30P30N airfoil.

Stowed chord	c	0.457 m
Span length	b	0.914 m
Slat chord	s	$0.15c$
Flap chord	f	$0.3c$
Slat deflection angle	δ_s	30°
Flap deflection angle	δ_f	30°

Figure A.4. Picture of airfoil and its parameters that is used to produce the High-lift flow data. This image was taken from [24],[23]

Bibliography

[1] Jean-Marie Beckers and Michel Rixen. Eof calculations and data filling from incomplete oceanographic datasets. *Journal of Atmospheric and oceanic technology*, 20(12):1839–1856, 2003.

[2] Manoj Diwakar, Pratibha Sharma, and Niranjan Lal. Edge detection using moore neighborhood. *International Journal of Computer Applications*, 61(3):0975–8887, 2013.

[3] Richard Everson and Lawrence Sirovich. Karhunen–loeve procedure for gappy data. *JOSA A*, 12(8):1657–1664, 1995.

[4] Hasan Gunes, Sirod Sirisup, and George Em Karniadakis. Gappy data: To krig or not to krig? *Journal of Computational Physics*, 212(1):358–382, 2006.

[5] Gregory Gutin and Abraham P. Punnen. *The traveling salesman problem and its variations*. Springer, 2007.

[6] LUMLEY JL. The structure of inhomogeneous turbulent flow. *Atmospheric turbulence and wave propagation*, 1967.

[7] A. Nekkanti and O. T. Schmidt. Frequency–time analysis, low-rank reconstruction and denoising of turbulent flows using spod. *Journal of Fluid Mechanics*, 926:A26, 2021.

[8] A. Nekkanti and O. T. Schmidt. Gappy spectral proper orthogonal decomposition. *Journal of Computational Physics*, 478:111950, 2023.

[9] Margaret A Oliver and Richard Webster. Kriging: a method of interpolation for geographical information systems. *International Journal of Geographical Information System*, 4(3):313–332, 1990.

[10] Richard W Reynolds and Thomas M Smith. Improved global sea surface temperature analyses using optimum interpolation. *Journal of climate*, 7(6):929–948, 1994.

[11] Olivier Rukundo and Hanqiang Cao. Nearest neighbor value interpolation. *arXiv preprint arXiv:1211.1768*, 2012.

[12] Pankaj Saini, Christoph M Arndt, and Adam M Steinberg. Development and evaluation of gappy-pod as a data reconstruction technique for noisy piv measurements in gas turbine combustors. *Experiments in Fluids*, 57:1–15, 2016.

[13] O. Schmidt. Spectral modal decomposition of turbulent flows. Presented at UCLA Mechanical Aerospace Engineering Fluid Mechanics Seminar Series, 2022.

[14] Oliver T. Schmidt and Tim Colonius. Guide to spectral proper orthogonal decomposition. *AIAA Journal*, 58(3):1023–1033, 2020.

[15] OT Schmidt, E Brothers, A Nekkanti, Y Zhang, and LN Cattafesta. Gappy spod for tr-piv data of stationary flows.

[16] Huanfeng Shen, Xinghua Li, Qing Cheng, Chao Zeng, Gang Yang, Huifang Li, and Liangpei Zhang. Missing information reconstruction of remote sensing data: A technical review. *IEEE Geoscience and Remote Sensing Magazine*, 3(3):61–85, 2015.

[17] Aaron Towne, Oliver T Schmidt, and Tim Colonius. Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis. *Journal of Fluid Mechanics*, 847:821–867, 2018.

[18] Daniele Venturi and George Em Karniadakis. Gappy data and reconstruction procedures for flow past a cylinder. *Journal of Fluid Mechanics*, 519:315–336, 2004.

[19] Yu-Long Wang, Tian-Bao Wang, and Qing-Long Han. Fault detection filter design for data reconstruction-based continuous-time networked control systems. *Information Sciences*, 328:577–594, 2016.

[20] Yang Zhang, Lou Cattafesta, and Lawrence Ukeiley. Identification of coherent structures in cavity flows using stochastic estimation and dynamic mode decomposition. In *10th International Symposium on Turbulence and Shear Flow Phenomena, TSFP*, volume 2017, page 3, 2017.

[21] Yang Zhang, Louis Cattafesta, and Lawrency Ukeiley. A spectral analysis modal method applied to cavity flow oscillations. In *11th international symposium on turbulence and shear flow phenomena (TSFP11), UK*, 2019.

[22] Yang Zhang, Louis N. Cattafesta, Kyle A. Pascioni, Meelan M. Choudhari, David P. Lockard, Mehdi R. Khorrami, and Travis Turner. *Fig. 6 Schematic of PIV setup (not to scale). [schematic].*

[23] Yang Zhang, Louis N. Cattafesta, Kyle A. Pascioni, Meelan M. Choudhari, David P. Lockard, Mehdi R. Khorrami, and Travis Turner. *Fig.1 Schematic of the 30P30N multielement airfoil/Table 1 Primary dimensions of 30P30N airfoil. .[schematic].*

[24] Yang Zhang, Louis N. Cattafesta, Kyle A. Pascioni, Meelan M. Choudhari, David P. Lockard, Mehdi R. Khorrami, and Travis Turner. *Slat Noise Control Using a Slat Gap Filler*.

[25] Yang Zhang, Louis N Cattafesta, and Lawrence Ukeiley. Fig. 1 schematic of the cavity model (units are in mm). coordinate system x, , and z are non-dimensionalized by d [photograph/schematic]. *Experiments in Fluids*, 61:3, 2020.

[26] Yang Zhang, Louis N Cattafesta, and Lawrence Ukeiley. Spectral analysis modal methods (samms) using non-time-resolved piv. *Experiments in Fluids*, 61:1–12, 2020.