Contents

List of Figures

Consider the continuous-time LTI system:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y = Cx(t)$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & -5 & -6 & -4 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

1 Controllability and Observability Gramians

(a) Computing Gramians via Lyapunov Equations

The Controllability Gramian (W_c) and Observability Gramian (W_o) for a stable continuous-time LTI system are the unique, symmetric, positive semi-definite solutions to the following Lyapunov equations:
The **Controllability Lyapunov Equation** is:

$$AW_c + W_cA^T + BB^T = 0 \tag{1}$$

The **Observability Lyapunov Equation** is:

$$A^TW_o + W_oA + C^TC = 0 \tag{2}$$

(b) Verification of Controllability and Observability

A system of order n is fully controllable and observable if its respective Gramians are full rank (i.e., $\text{rank}(W) = n$). The state-space dimension of the given system is $n = 4$. Our results provide the rank for each computed Gramian:

```
Rank of Controllability matrix via solving Lyapunov Eqn: 4
Rank of Observability matrix via solving Lyapunov Eqn: 4
```

Since the rank of both Gramians is 4, which equals the order of the system, we can definitively conclude:

- ✓ **The system is fully controllable.** Every state is reachable from the origin through a suitable control input.
- ✓ **The system is fully observable.** The initial state of the system can be uniquely determined by observing its output over a finite time interval.

(c) Comparison of `lyap` and `gram` Commands

The results show:

```
Difference b/w both Controllability matrices:   1.0e-15 * [...]
Difference b/w both Observability matrices:     1.0e-15 * [...]
```

The differences are on the order of 10^{-15}, which is within the bounds of standard machine epsilon (floating-point precision). This is expected, as the `gram` function is a higher-level command that solves the same underlying Lyapunov equation.

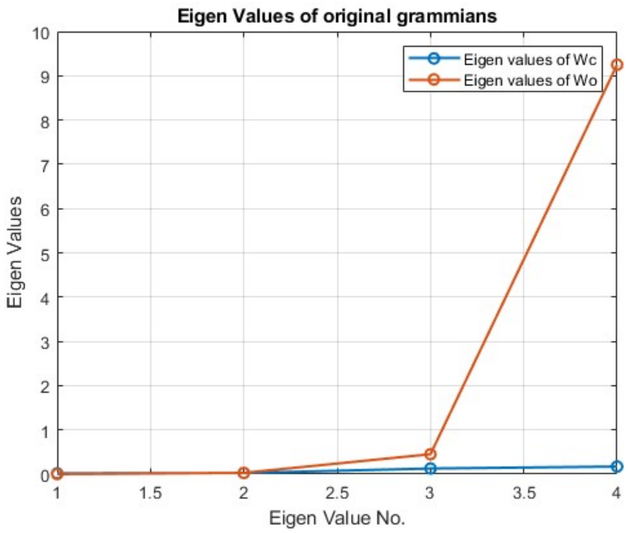

Figure 1: Eigenvalues of the original W_c and W_o gramians

2 Numerical Conditioning and Grammians

(a) Eigenvalues of Original Gramians and their Spread

Observations from the Plot:

- **Eigenvalues of W_c (Controllability):** The eigenvalues are all relatively small and tightly clustered, with the largest value being approximately 0.3. This suggests that the system states are somewhat uniformly, but not strongly, controllable. No single state or combination of states is exceptionally easy to control compared to others.

- **Eigenvalues of W_o (Observability):** The eigenvalues exhibit a very large spread. There are three small eigenvalues (close to zero) and one very dominant eigenvalue of approximately 9.3. This indicates a highly unbalanced observability. The system has one direction in its state-space (corresponding to the eigenvector of the large eigenvalue) that is *highly observable*, while the other directions are *very weakly observable*.

(b) Why a Large Spread Indicates Difficulty for Balanced Truncation

Balanced truncation is a powerful model reduction technique that relies on finding a state-space realization where the controllability and observability Gramians are equal and diagonal:

$$W_c^{\text{bal}} = W_o^{\text{bal}} = \Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n) \tag{3}$$

The diagonal entries σ_i are the **Hankel Singular Values (HSVs)**, which quantify the ”energy” of each state in the system's input-output map. States corresponding to small σ_i can often be truncated without significant loss of fidelity.

A large spread in the eigenvalues of the *original* Gramians, as seen with W_o, points to a system that is far from being balanced. This creates two potential difficulties for the balanced truncation procedure:

1. **Numerical Conditioning:** The similarity transform T required to bring a highly unbalanced system into a balanced realization can be ill-conditioned (i.e., have a very large condition number). Applying a numerically ill-conditioned transformation can introduce significant floating-point errors, potentially compromising the accuracy of the reduced-order model.

2. **Ambiguity in Truncation:** While a large spread in the Hankel Singular Values (σ_i) is *desirable* for making a clear decision on where to truncate, a large spread in the initial Gramian eigenvalues doesn't guarantee this. If the resulting HSVs do not exhibit a clear "gap" (e.g., $\sigma_k \gg \sigma_{k+1}$), it becomes difficult and subjective to choose the order of the reduced model.

(c) Effect of a Similarity Transformation

We apply a state-scaling transformation with the matrix $T = \mathrm{diag}([0.1, 0.2, 0.3, 0.5])$. Under a state transformation $x = Tz$, the Gramians transform as follows:

$$W_{c,\text{new}} = T^{-1} W_c (T^{-1})^T \tag{4}$$
$$W_{o,\text{new}} = T^T W_o T \tag{5}$$

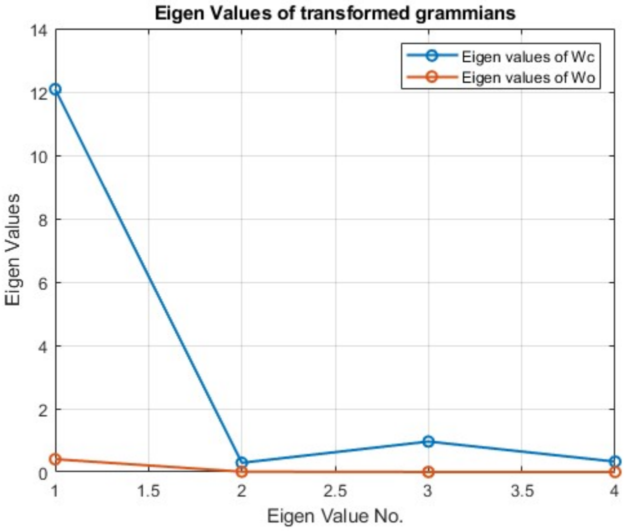

Figure 2: Eigenvalues of the transformed W_c and W_o grammians

Observations from the Transformation:

- **Eigenvalues of $W_{c,\text{trans}}$:** These eigenvalues now show an extremely large spread, with one value surging to approximately 12. The transformation matrix for controllability is T^{-1}, which has large diagonal entries. This matrix acts as an amplifier, dramatically increasing the controllability measures. The system, in these new coordinates, now has a state direction that is exceptionally easy to control.
- **Eigenvalues of $W_{o,\text{trans}}$:** Conversely, the eigenvalues of the transformed observability Gramian are now all clustered near zero. The transformation matrix here is T, whose small entries act as an attenuator. The previously high observability energy has been almost entirely suppressed. In this new coordinate system, all states are now very difficult to observe.

The most critical takeaway is what *does not* change. The Hankel singular values are the square roots of the eigenvalues of the product $W_c W_o$.

$$\begin{aligned} W_{c,\text{new}} W_{o,\text{new}} &= (T^{-1} W_c (T^{-1})^T)(T^T W_o T) \\ &= T^{-1} W_c ((T^{-1})^T T^T) W_o T \\ &= T^{-1} W_c (I) W_o T \\ &= T^{-1} (W_c W_o) T \end{aligned}$$

This shows that the product $W_c W_o$ undergoes a true similarity transformation. A core property of similarity transformations is that they **preserve eigenvalues**.
Therefore, while a state transformation can arbitrarily redistribute the "energy" between controllability and observability, it does not change the fundamental input-output properties of the system represented by the Hankel singular values.

3 Balanced Realization

(a) Manual Balanced Realization Algorithm

The process involves three key steps:

1. **Cholesky Factorization:** The symmetric, positive-definite Gramians W_c and W_o are decomposed into products of a lower-triangular matrix and its transpose.

$$W_c = R_c R_c^T \quad \texttt{(Rc = chol(Wc, 'lower'))}$$
$$W_o = R_o R_o^T \quad \texttt{(Ro = chol(Wo, 'lower'))}$$

2. **Singular Value Decomposition (SVD):** SVD is performed on the product of the Cholesky factors. The resulting singular values are the Hankel Singular Values (HSVs) of the system.

$$R_o^T R_c = U \Sigma V^T \quad \texttt{([U,S,V] = svd(Ro'*Rc))} \tag{6}$$

3. **Construct Transformation Matrix:** The balancing transformation matrix T and its inverse T^{-1} are constructed from the components of the Cholesky factorization and the SVD.

$$T = R_c V \Sigma^{-1/2}$$
$$T^{-1} = \Sigma^{-1/2} U^T R_o^T$$

(b) Comparison of Transformation Matrices

The numerical output shows the matrices calculated manually (T_1, T_1^{-1}) and those from the 'balreal' command (T_2, T_2^{-1}).

Observation: Upon careful inspection, we notice a fascinating relationship:

- The matrix 'T_{inv}' from the **manual calculation is nearly identical** to the matrix 'T' from the **balreal command** (with minor sign differences in the last column/row).
- Consequently, 'T' from the manual calculation is nearly identical to 'T_{inv}' from 'balreal'.

This suggests that your manual algorithm and the 'balreal' command found balancing transformations that are essentially inverses of each other.

Explanation: The balancing transformation matrix is **not unique**. Any valid transformation T can be followed by a transformation Q where Q is a signed permutation matrix and the system will remain balanced. The fact that one matrix is the inverse of the other is likely due to different algorithmic conventions within the 'balreal' implementation, but both are valid.

(c) Comparison of Hankel Singular Values

Observation from the Plot: The plot shows that the red circles (manual method) and the green line ('balreal' command) are **perfectly superimposed**.

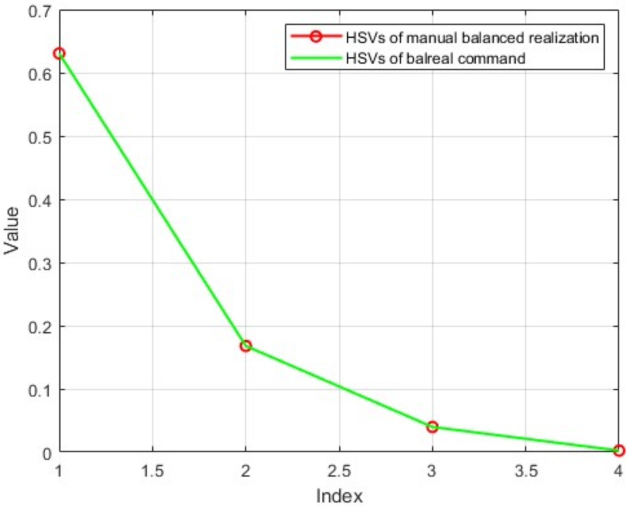

Figure 3: Comparison of HSVs from the manual algorithm and the ‘balreal‘ command

Conclusion: This confirms that despite the differences in the transformation matrices, both methods have successfully produced the **exact same Hankel Singular Values**. The numerical values are approximately:

$$\sigma = \{0.630,\ 0.167,\ 0.039,\ 0.002\}$$

4 Balanced Truncation Model Reduction

(a) & (b): Truncation and Step Response Comparison

The process of balanced truncation is straightforward once a balanced realization $(A_{bal}, B_{bal}, C_{bal})$ is obtained. To reduce the system to order r, we simply retain the first r rows and columns of the state matrix, the first r rows of the input matrix, and the first r columns of the output matrix.

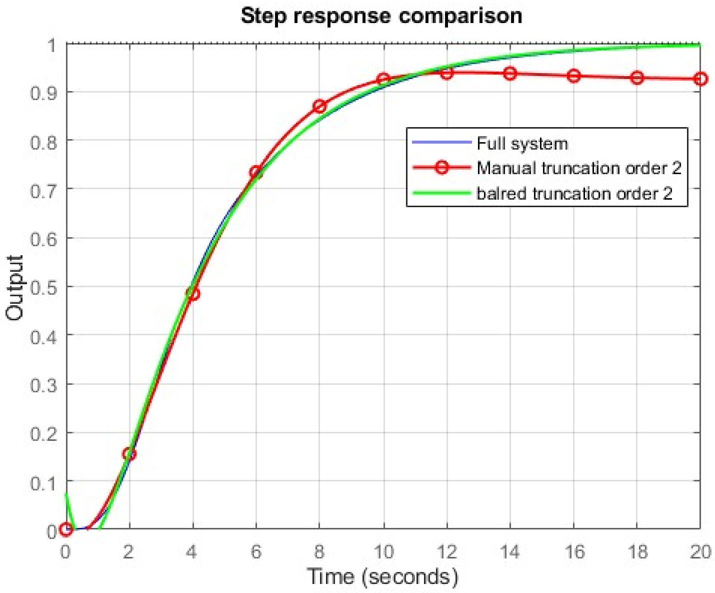

Figure 4: Step response comparison of different models

Observations from the Step Response Plot:

- **Excellent Fidelity:** The step responses of both second-order models (manual truncation and 'balred' truncation) provide an exceptionally good match to the full fourth-order system. The transient behavior, including the rise time and overall shape, is captured with high accuracy.
- **Consistency:** The responses of the manually created model and the one from the `balred` command are visually indistinguishable.
- **Minor Discrepancies:** Upon close inspection, the reduced models exhibit two small differences: a very slight initial undershoot that is not present in the original system, and a steady-state value that is slightly lower (approx. 0.93 vs 1). These minor deviations are expected and acceptable trade-offs for a 50% reduction in system complexity.

(c) Error Bounds vs. Actual Error

One of the most powerful features of balanced truncation is the existence of a provable *a priori* error bound. The H-infinity norm of the error system, $\|G(s) - G_r(s)\|_\infty$, is guaranteed to be bounded by twice the sum of the truncated Hankel Singular Values.
The relevant Hankel Singular Values from the previous analysis were approximately $\sigma = \{0.630,\ 0.167,\ 0.039,\ 0.002\}$. For a reduction to order $r = 2$, we truncate the states corresponding to σ_3 and σ_4.

Theoretical Error Bound Calculation:

$$\|G - G_r\|_\infty \leq 2 \sum_{k=r+1}^{n} \sigma_k = 2(\sigma_3 + \sigma_4) \tag{7}$$

```
Upper Bound for error: 0.084332
```

$$\|G - G_r\|_\infty \geq \sigma_{r+1} = \sigma_3 \tag{8}$$

```
Lower Bound for error: 0.039790
```

Actual Computed Error: The code then computes the actual H-infinity norm of the error between the full and reduced systems for both methods:

```
Actual H-infinity norm of error: 0.076564 (manual method)
Actual H-infinity norm of error: 0.078039 (balred method)
```

The actual errors are:

- Within bounds for both reduction methods.
- Close to the upper bound, indicating that the bound is not overly conservative in this case and provides a useful estimate of the true error.

5 Balanced Truncation for structural vibration model

The objective is to analyze and perform model reduction on a 12th-order state-space model of a six-mass-spring-damper system. The analysis covers stability, controllability, observability, balanced realization, model reduction, and error analysis. A key aspect is investigating how to handle a system that is not fully controllable.

(a) Stability Check

Eigenvalue Analysis: We first computes the eigenvalues of the state matrix A. Our result confirms that **all modes are stable**. The eigenvalues of such structural models are expected to be complex conjugate pairs with small negative real parts, corresponding to lightly damped oscillatory modes, which is physically consistent.

Suitability for Gramians: Because the system is stable, the integrals that define the controllability and observability Gramians (W_c and W_o) converge. Therefore, it is mathematically valid to compute the Gramians.

(b) Controllability and Observability

Rank Tests: The code performs rank tests on the controllability and observability matrices. The results are definitive:

- `System is not controllable`
- `System is observable`

This is the most critical finding in the entire analysis. The rank of the controllability matrix (rank=2) is less than the state dimension ($n = 12$), indicating an uncontrollable subspace. The message from the 'minreal' command, `10 states removed`, reveals that the controllable subspace has a dimension of only 2.

Physical Interpretation: This result is physically intuitive. The system is a chain of masses, but the input force is only applied to the last mass. It is not possible to independently excite all 12 system states (6 positions and 6 velocities) from this single input. The dynamics are constrained, leading to uncontrollable modes. However, since the output is the displacement of all six masses, the full state of the system is observable.

(c) Balanced Realization and Hankel Singular Values

The 'balreal' command computes a balanced realization, from which the Hankel Singular Values (HSVs) are extracted.

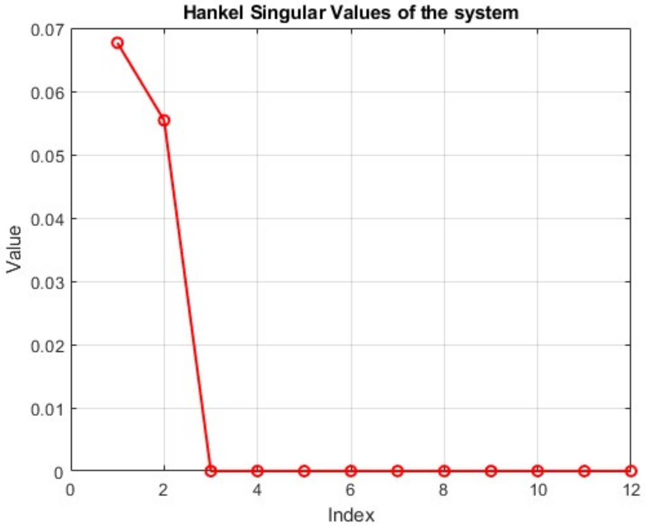

Figure 5: Hankel Singular Values of the 12th-order system.

Observations from the HSV Plot:

- There are **two dominant HSVs** (approx. 0.068 and 0.056)
- The subsequent 10 HSVs drop precipitously to near-zero values.

These 10 near-zero HSVs correspond directly to the 10 states in the uncontrollable subspace. They have effectively zero input-output energy, signifying they do not participate in the system's response to the given input.

Decision on Truncation: The significant "energy gap" occurs after the second state. The system's input-output behavior is almost entirely captured by the first two states. Therefore, the system should be reduced to order $r = 2$.

(d) & (e): Model Reduction and Comparison

'balred' wasn't able to approximate the system to order 4 due to inherent pole-zero cancellations. The standard manual balancing algorithm requires W_c to be a positive definite matrix, which is not possible in an uncontrollable system. The controllability Gramian W_c is rank-deficient (rank 2) and thus not positive definite, causing the Cholesky factorization ('chol') to fail. Thus, we first use **'minreal(sys)'** to perform a Kalman decomposition to isolate and discard the uncontrollable (and unobservable, if any) parts of the system, yielding an exact minimal realization. Manual balancing was then performed on this well-behaved 2nd-order system. So, both automated ('balred') and manual methods were used to reduce the system to order $r = 2$.

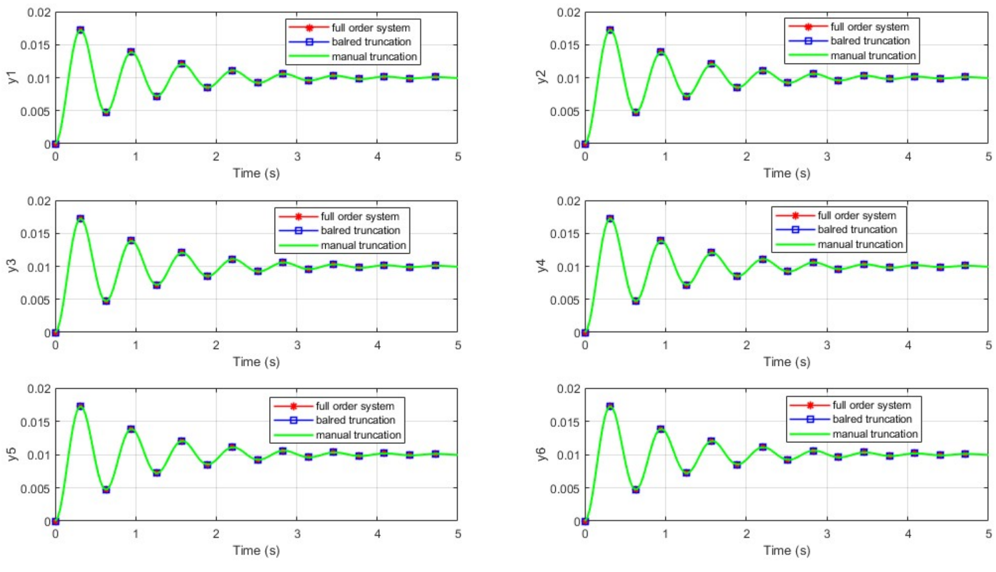

Figure 6: Step response comparison for all six mass displacements with truncated order '2'

Observations from Step Response Plots: The six subplots, showing the displacement of each mass, demonstrate that the step responses of the **full 12th-order model**, the **'balred' 2nd-order model**, and your **manual 2nd-order model** are **perfectly identical**.

Relative Error Analysis: The computed relative H-infinity error confirms the visual result:

```
Relative H_inf error between the two systems is: 5.899079e-15
```

An error on the order of 10^{-15} is effectively zero, residing at the level of machine numerical precision.

6 Dynamic Mode Decomposition

The objective is to apply Dynamic Mode Decomposition (DMD) to a dataset of vorticity snapshots from a simulation of fluid flow past a cylinder. The goals are to derive a low-dimensional linear model that describes the system's dynamics, determine an appropriate rank for the model, reconstruct the flow field using this reduced-order model (ROM), and quantitatively analyze the reconstruction error as a function of the model size.

DMD Algorithm

1. **Data Preparation:** The data is organized into two snapshot matrices, X and Y, where Y is the time-shifted version of X (i.e., $Y = [x_2, x_3, \ldots, x_m]$ and $X = [x_1, x_2, \ldots, x_{m-1}]$). This establishes the relationship $x_{k+1} \approx Ax_k$ that DMD seeks to model.

2. **Singular Value Decomposition (SVD):** The algorithm begins by performing an SVD on the initial snapshot matrix $X = U\Sigma V^T$. This finds an optimal, low-dimensional orthogonal basis (the POD modes in U) that captures the spatial structures in the flow.

3. **Low-Rank Dynamics:** The full-order system operator A is projected onto the low-dimensional basis defined by the first r columns of U. This creates a small $r \times r$ matrix $\hat{A}$ that captures the essential dynamics.

$$\hat{A} = U_r^T Y V_r \Sigma_r^{-1}$$

4. **Eigendecomposition:** An eigendecomposition of $\hat{A}$ is performed to find its eigenvalues λ and eigenvectors W. These represent the temporal frequencies/growth rates and the spatial mode weightings of the dominant dynamic patterns.

5. **DMD Mode Reconstruction:** The eigenvectors W (which are in the low-dimensional POD basis) are projected back into the high-dimensional physical space to form the DMD modes, Φ.

$$\Phi = Y V_r \Sigma_r^{-1} W$$

6. **System Reconstruction:** The full-time evolution is reconstructed as a linear combination of these DMD modes, with each mode evolving according to its corresponding eigenvalue. Continuous-time formulation is with $\omega = \log(\lambda)/\Delta t$.

Determining the Truncation Rank (r)

- **Connection to Energy:** The singular values (σ_i, the diagonal of Σ) from SVD are a direct measure of the "energy" captured by each corresponding mode. The total energy in the snapshot data is proportional to the sum of the squares of the singular values.

- **Explained Variance:** By computing the "explained variance" ($\sigma_i^2 / \sum \sigma_j^2$) and its cumulative sum, you can determine how many modes are required to capture a certain percentage of the system's total energy.

- **Criterion Used:** We seek the minimum number of modes to capture **99.9% of the variance**, which results in $r = 9$. This is an excellent choice for achieving a high-fidelity visual reconstruction, as it guarantees that nearly all of the energy (and thus the dominant flow features) is included in the model.

Flow Field Reconstruction

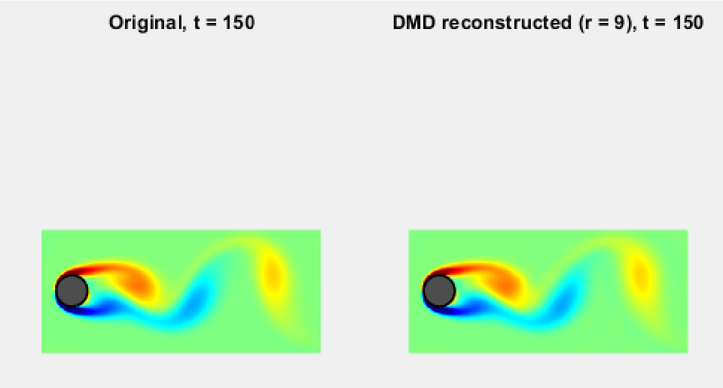

Figure 7: Side-by-side comparison of the flow field at the final time step ($t = 150$)

The provided image shows a snapshot of the original and reconstructed flow fields at the final time step. Running the full code produces an animation of the entire time evolution. The key takeaway is:

- **Extraordinary Fidelity:** With just 9 modes, the DMD model has successfully captured all the essence. The reconstructed field shows vortices with the correct size, shape, intensity, and spatial position. This visually confirms that $r = 9$ is sufficient to model the dominant coherent structures of the flow.

Model Reduction Error Analysis

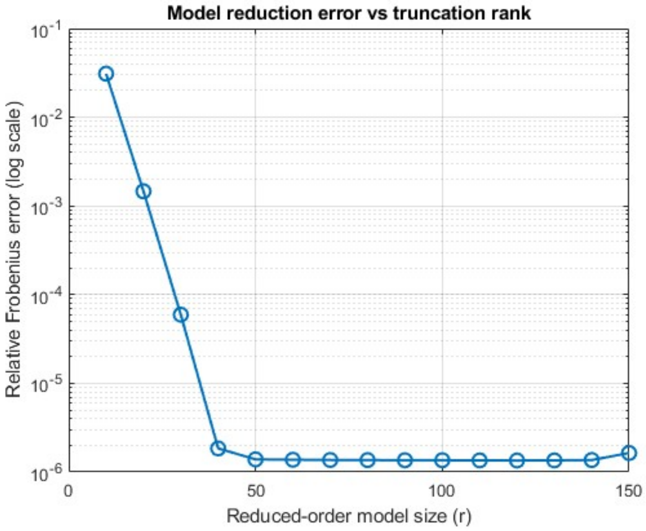

Figure 8: Relative Frobenius error of DMD reconstruction as a function of truncation rank r

Justification for the Frobenius Norm: The Frobenius norm $\|X - \hat{X}\|_F$ effectively computes the square root of the sum of squared errors over every grid point and every time snapshot. This makes it a measure of the total integrated energy of the error field. Using the relative form normalizes this by the energy of the original data, making the error metric dimensionless and comparable across different problems.

Interpretation of the Error Plot:

- **Rapid Error Decay:** The error decreases exponentially as r increases from 10 to around 30. This region corresponds to capturing the remaining low-energy but still coherent dynamic structures.
- **Error Plateau:** For $r > 40$, the error plateaus at a very low value of $\approx 10^{-6}$. This indicates that the remaining modes that could be added correspond to very fine-scale features or numerical noise. The singular values associated with these modes are so small that their inclusion does not meaningfully improve the model's accuracy.

Conclusion: What is the Right Model Size?

This analysis demonstrates that there isn't one "correct" model size, but rather a trade-off between model complexity and desired accuracy.

- For **visual fidelity and capturing the physics** of the main vortex shedding phenomenon, choice of $r = 9$ (based on 99.9% energy) is highly efficient.
- If the goal was to achieve the **maximum possible numerical accuracy**, the error plot suggests a rank of $r \approx 40$ would be more appropriate, as this is where the error floor is reached. However, this comes at the cost of a model that is over four times larger.

7 Total Least Squares DMD

The objective is to investigate and compare the performance of two Dynamic Mode Decomposition algorithms, Standard DMD and Total Least Squares DMD (TLS-DMD) in identifying the dynamics of a linear system from noisy data. A Monte Carlo simulation is employed to generate 500 noisy realizations of a mass-spring-damper system's evolution. The statistics (mean and variance) of the system eigenvalues estimated by each method are then analyzed to assess their accuracy, bias, and robustness.

Standard DMD vs. Total Least Squares DMD

The core difference between the two methods lies in their underlying assumptions about noise in the data.

Standard DMD is based on a standard least-squares regression. It seeks to find an operator A that minimizes the error in predicting the next state, assuming the current state is known perfectly. It solves the problem:

$$\min_{A} \|Y - AX\|_F$$

This implicitly assumes that the data matrix X is noise-free and all measurement error resides in Y. In reality, both matrices are noisy, which introduces a systematic error, or **bias**, into the standard DMD estimate.

Total Least Squares DMD (TLS-DMD) addresses this flaw by acknowledging that both X and Y are corrupted by noise. Instead of fitting Y to X, it finds the optimal low-rank linear relationship within the combined data space $[X; Y]$. This is a more physically realistic assumption for most measurement scenarios and is known to produce an **unbiased** estimate of the underlying system operator, especially for linear systems with Gaussian noise.

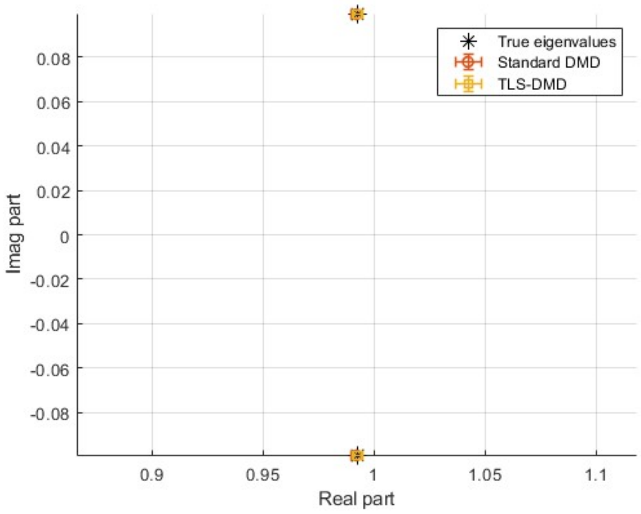

Figure 9: Errorbar plot showing the estimate and it's spread for noise = 0.02

Analysis of Monte Carlo Simulation Results

The error bar plots concisely show that for a given noise level, what is the estimate (mean) and how much it varies (std deviation) with the help of two bars showing the spread. For the given noise level, $\sigma = 0.02$, both the methods are sufficiently close to the true eigen values of the system. Now, let's increase the noise and observe what happens.

Low Noise ($\sigma = 0.02$)

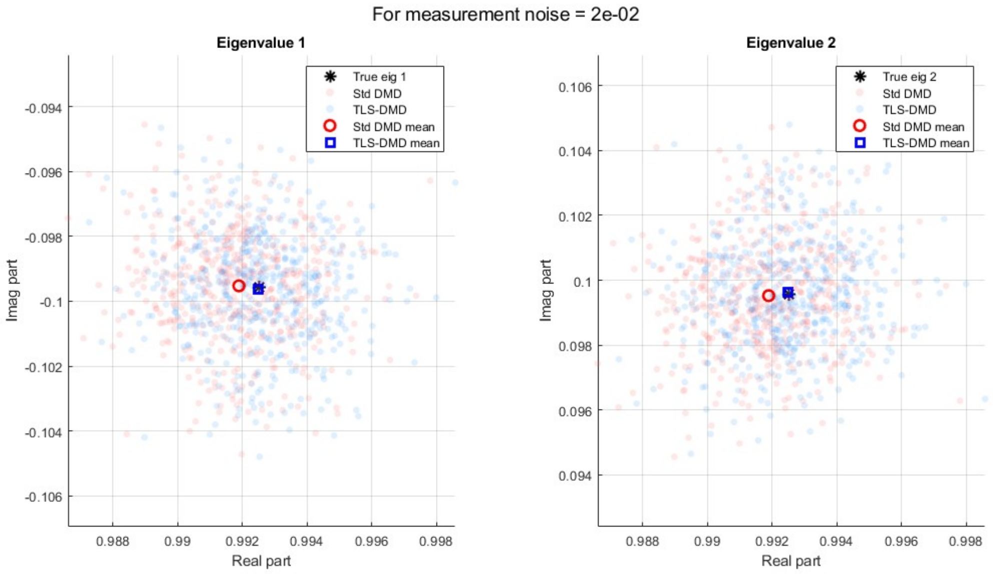

Figure 10: Distribution of estimated eigenvalues for low noise ($\sigma = 0.02$).

At a low noise level, both methods perform reasonably well. However, subtle differences are already apparent:

- The blue cloud (TLS-DMD) is slightly more compact and more centered on the true eigenvalue than the red cloud (Std DMD).

- This indicates that even with minimal noise, TLS-DMD exhibits slightly lower **variance** (tighter clustering) and lower **bias** (mean is closer to the true value).

Medium Noise ($\sigma = 0.05$)

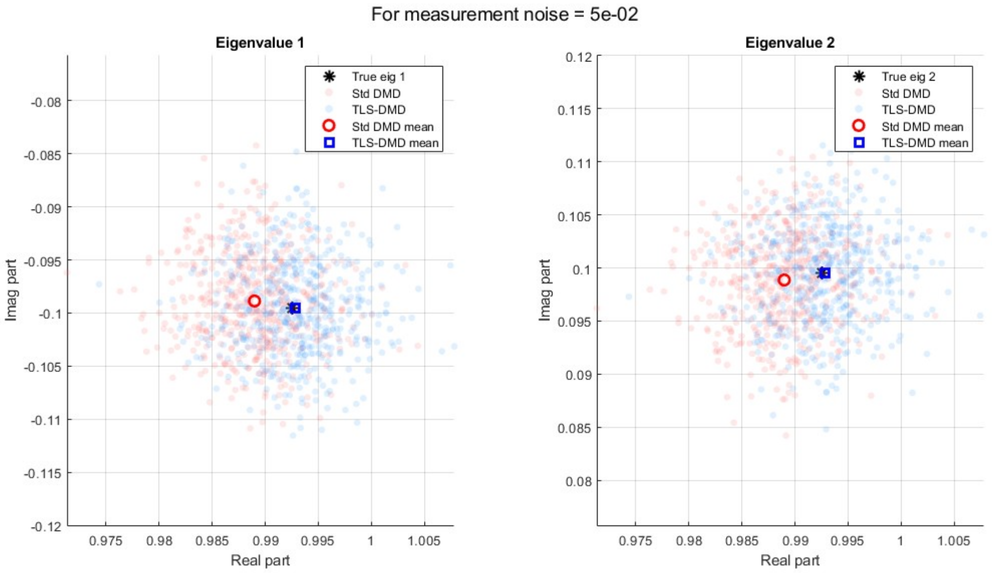

Figure 11: Distribution of estimated eigenvalues for medium noise ($\sigma = 0.05$).

As the noise increases, the superiority of TLS-DMD becomes more pronounced.

- The mean of the Standard DMD estimates has visibly drifted away from the true value, indicating a significant bias. The spread of the red cloud has also grown considerably.
- In contrast, the mean of the TLS-DMD estimates remains almost perfectly centered on the true eigenvalue, demonstrating its unbiased nature. The blue cloud is still significantly more concentrated than its standard counterpart.

High Noise ($\sigma = 0.1$)

At high noise levels, the failure of Standard DMD's underlying assumptions becomes critical.

- **Significant Bias:** The mean of the Standard DMD estimates is now far from the true value. Critically, the real part of the mean is much closer to 1.0, indicating that the noise is causing Standard DMD to systematically **underestimate the system's damping**.
- **High Variance:** The spread of the red cloud is enormous, meaning any single trial gives a highly unreliable estimate.
- **TLS-DMD Robustness:** Remarkably, the TLS-DMD mean is still an excellent, nearly unbiased estimate of the true eigenvalue, and its variance, while larger than before, is still drastically smaller than that of Standard DMD.

Some insights from the results:

1. **TLS-DMD is an unbiased estimator:** Its mean estimate of the system's eigenvalues converges to the true value, regardless of the noise level.
2. **Standard DMD is a biased estimator:** Noise introduces a systematic error that pulls the estimated eigenvalues away from their true locations.

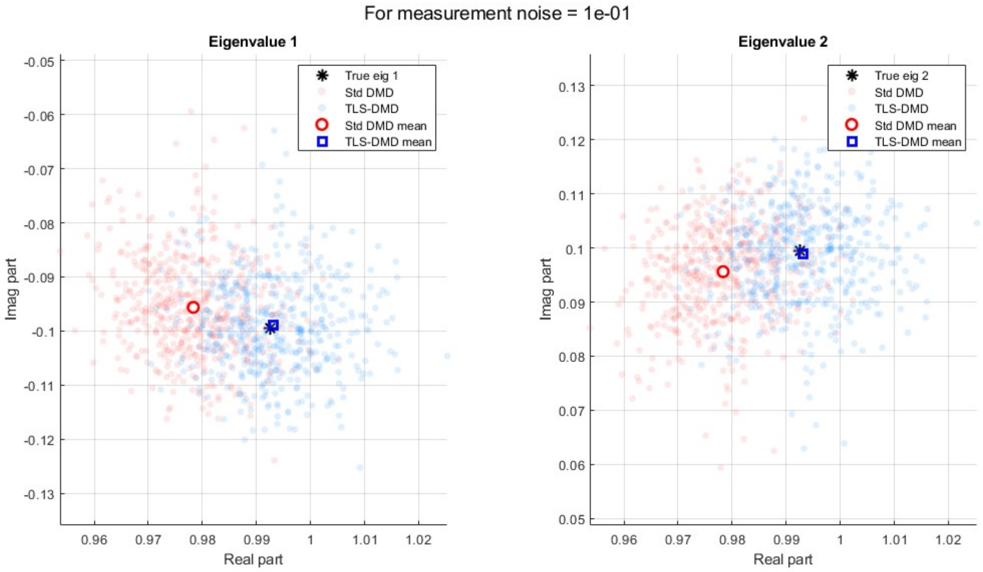

Figure 12: Distribution of estimated eigenvalues for high noise ($\sigma = 0.1$).

3. **TLS-DMD has lower variance:** The estimates from TLS-DMD are more consistent and cluster more tightly around the true value, making any single estimate more reliable.

4. **Robustness:** As noise increases, the performance of Standard DMD degrades rapidly, whereas TLS-DMD remains remarkably accurate.

Conclusion: For any real-world application where measurement noise is a factor, TLS-DMD is the fundamentally superior method. Its formulation correctly accounts for the nature of noise in physical systems, yielding estimates that are more accurate (unbiased) and more reliable (lower variance).

8 Proper Orthogonal Decomposition-based ROM

The objective is to develop, implement, and validate a Proper Orthogonal Decomposition (POD) based Reduced-Order Model (ROM) for the 1D viscous Burgers' equation. This equation features both a linear diffusion term and a quadratic convective nonlinearity, making it a canonical problem for nonlinear model reduction. The process involves simulating the full-order model (FOM) to generate data, extracting a low-dimensional basis via POD, constructing the ROM via Galerkin projection, and analyzing its accuracy and efficiency for different model sizes (r).

Full-Order Model (FOM): The spatial domain is discretized using central differences for both the first (convection) and second (diffusion) derivatives. This transforms the partial differential equation (PDE) into a large system of coupled ordinary differential equations (ODEs). The simulation is run using a simple Forward Euler time-stepping scheme to generate a "snapshot matrix" containing the system's evolution.

Proper Orthogonal Decomposition (POD) and Selecting ROM Size (r): The core of POD is the Singular Value Decomposition (SVD) of the snapshot matrix, $U = \Phi S V^T$. The left singular vectors Φ, form the POD basis, an optimal orthogonal basis that captures the spatial patterns in the data. The singular values S, quantify the "energy" associated with each basis vector.

- The total energy of the system is proportional to the sum of the squared singular values.
- The cumulative energy captured by the first r modes is given by $\sum_{i=1}^{r} \sigma_i^2 / \sum_{j=1}^{n} \sigma_j^2$.
- By setting a threshold (e.g., 99.9%), you can find the smallest rank r that captures the vast majority of the system's dynamics.

Your result is striking:

```
To capture 99.9% energy ... we need to atleast truncate it to order 2
```

This immediately implies that the dynamics of this system are fundamentally **low-dimensional**. The solution's evolution is dominated by only two principal spatial patterns. This makes it an ideal candidate for aggressive model reduction.
The ROM is constructed by projecting the full-order governing equations onto the low-dimensional POD basis Φ_r.

Linear (Diffusion) Term: The reduced diffusion operator, $A = \nu \Phi_r^T D_2 \Phi_r$ is a small $r \times r$ matrix.

Nonlinear (Convection) Term: The reduced convection operator, $N = \phi_r^T((diag(\phi_r)(D_1\phi))$ is a $r \times r^2$ matrix. This quadratic non-linearity results in a tensor.

Comparing ROM vs FOM

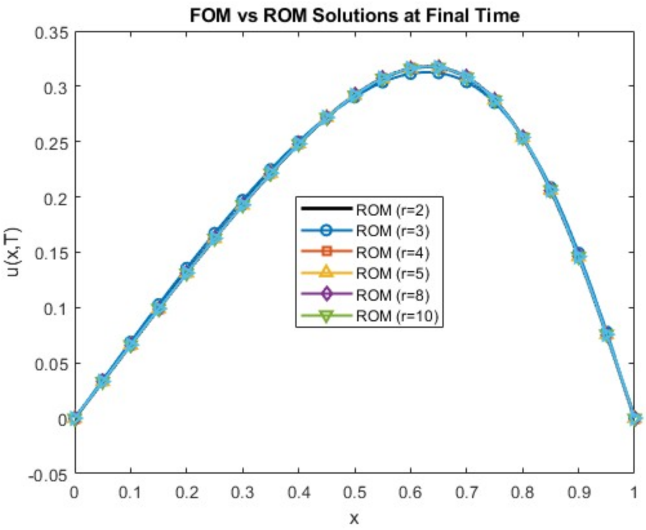

Figure 13: Solution profile at the final time ($T = 1$) for various (r)

Solution at Final Time: The plot of $u(x, T)$ shows that **all ROMs, even for $r = 2$, produce a final solution that is visually identical to the FOM**. This is a good confirmation of the energy analysis. Since just two modes capture over 99.9% of the dynamics, the reconstruction using these modes is expected to be highly accurate.

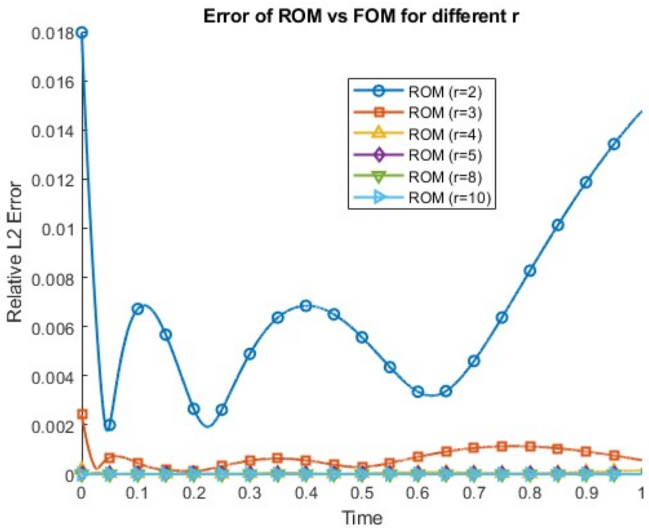

Figure 14: Evolution of the relative L_2 error over time

Error Evolution in Time:

- The ROM with $r = 2$ has a noticeable error (peaking around 1.8%) that oscillates in time. While small, it is significantly larger than the others.

- For $r = 3$ and higher, the error drops by an order of magnitude and remains exceptionally low ($< 0.2\%$) for the entire simulation. This shows that the third mode, while containing little energy, is crucial for correcting the small inaccuracies of the $r = 2$ model.

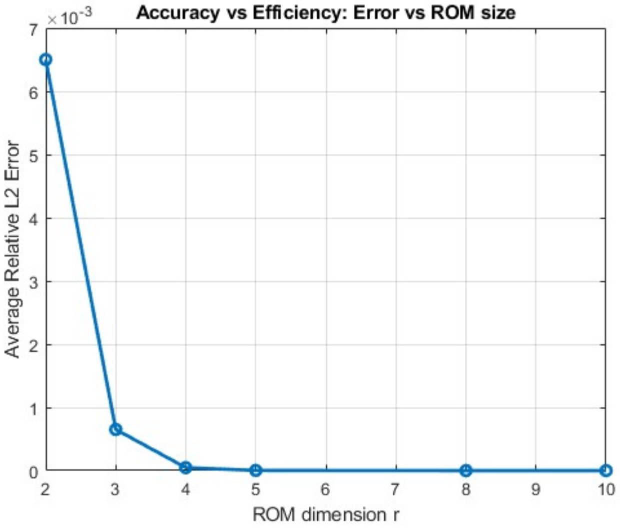

Figure 15: Trade-off between ROM size and average error

Accuracy vs. Efficiency:

- There is an **elbow** in the curve where the error drops by an order of magnitude when moving from $r = 2$ to $r = 3$.

- After $r = 4$, the improvement in accuracy is negligible. The error essentially plateaus.

Conclusion: The optimal ROM dimension is $r = 3$ or $r = 4$. This choice provides a near-perfect accuracy with minimal computational complexity. Using $r = 10$ offers no practical benefit and unnecessarily increases the cost of the ROM simulation.

Other Insights:

1. **Suitability of POD:** The smooth, decaying solution of the viscous Burgers' equation for a simple initial condition is inherently low-dimensional. The system's energy is concentrated in a few large-scale spatial patterns, which makes it a perfect application for POD.

2. **Power of Galerkin Projection:** The intrusive approach allows for the creation of a genuine, predictive dynamical system for the reduced state. Once the operators A and N are assembled, the ROM can be simulated for different initial conditions (projected onto the basis) without ever returning to the expensive FOM.

3. **Quantitative Model Selection:** The error analysis is not just a validation step; it is a design tool. The "Accuracy vs Efficiency" plot allows you to quantitatively justify the choice of a model size that balances computational cost with required accuracy. For this problem, a 101-dimensional system was faithfully replaced by a 3 or 4-dimensional one, achieving a massive reduction in complexity with virtually no loss in accuracy.