POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Matematica

Master's Degree Thesis

Entropy Stable Collocation Reduced Order Modeling for Nonlinear Conservation Laws

Supervisors

Prof. Davide Carlo Ambrosi
Dr. Michele Giuliano Carlino

Candidate

Giorgia Lanciotti

Academic Year 2025-2026

Abstract

The evolution of mathematical modeling and numerical simulation has provided increasingly accurate analyses of complex phenomena but increased computational cost. To address this challenge, reduction methods are being developed that maintain physical consistency while reducing complexity.
In this work we study a class of structure-preserving collocation reduced order models, which integrate structural constraints to preserve fundamental properties. Particular attention is paid to conservation of entropy, which is essential to ensure physical consistency. It focuses on the Burgers equation, analyzing the construction of an entropy stable reduced model using proper orthogonal decomposition and validating its effectiveness with numerical tests. Finally, the hyper-reduced case is considered using a collocation strategy based on Nonnegative Least Squares, highlighting its limitations and possible future developments.

Contents

Introduction

In the last few decades, the rapid evolution of mathematical modeling and numerical simulation has profoundly transformed the way complex physical systems are studied. The ability to reproduce nonlinear dynamics, discontinuities and multi-scale phenomena has made numerical methods an indispensable tool for scientific and engineering applications. In areas such as aerodynamics, fluid dynamics, heat transfer and structural mechanics, the development of increasingly accurate computational algorithms has enabled remarkable progress in both theoretical research and technological design.

However, the pursuit of greater accuracy and robustness in numerical solvers has inevitably led to an increase in computational complexity. High-fidelity simulations, referred to as Full Order Models, abbreviated as FOMs, require extremely fine spatial and temporal discretizations, or very high-order approximations to capture the relevant physical scales of the problem. As a result, the computational cost, in terms of CPU time and memory usage, can become prohibitive, particularly when a wide range of parameters must be explored. These limitations are especially severe when simulations must be performed repeatedly, as in optimization or real-time control tasks.

To address these challenges, the scientific community has developed a wide range of Model Order Reduction techniques, abbreviated as MOR, whose main goal is to reduce the computational effort while maintaining an acceptable level of accuracy and physical consistency. The idea is to construct a reduced model that captures the dominant dynamics of the system by exploiting the fact that, even though the full-order model may involve many degrees of freedom, the underlying behavior can be captured by a lower number of degrees of freedom, so the essential information of the solution can be represented using a limited number of parameters or modes.

Among the various families of model order reduction techniques, a particularly promising class is that of collocated-Model Order Reduction, abbreviated as c-ROMs. These methods evaluate the reduced dynamics at a small number of selected spatial points, called collocation points. The selection of these collocation points can be carried out using different strategies; one of the most effective is based on the Nonnegative Least Squares (NNLS).

An additional and increasingly important aspect of model reduction is the preservation of the structure of the original problem. Classical numerical schemes are often designed

to satisfy fundamental physical constraints, such as the conservation of mass, momentum and energy, or the non-decreasing behavior of entropy. When these properties are not maintained at the reduced level, the resulting model may exhibit unphysical instabilities or fail to reproduce the correct qualitative behavior of the system. For this reason, the development of structure-preserving reduced models, capable of ensuring stability, positivity and entropy consistency, has become an active field of research.

In this thesis, particular attention is devoted to the notion of entropy stability. For nonlinear hyperbolic equations, such as the Burgers equation, entropy provides a crucial measure of the physical admissibility of solutions. The work presented in this dissertation aims to explore the construction, analysis and numerical validation of entropy stable collocation-based reduced models. The focus is placed on understanding how the reduction and hyper-reduction processes affect the fundamental properties of the full-order system and how suitable formulations can be designed to retain entropy consistency within a reduced dimensional setting. This thesis explores the development of entropy stable collocation-based models, starting from high-fidelity discretisations and proceeding toward structure preserving formulations for nonlinear conservation laws.

In particular, the first chapter starts with a review of high-fidelity numerical methods for the discretisation of partial differential equations. High-fidelity schemes for transport and diffusion equations are presented, highlighting their accuracy and limitations associated with computational cost. The discussion then moves on to the theoretical formulation of the hyper-reduction technique based on a collocation approach, which significantly reduces the number of spatial evaluation points while maintaining a high level of accuracy compared to the full model.

In Chapter 2, these concepts are applied to the inviscid Burgers equation. The equation is solved numerically using the Lax-Friedrichs scheme, which is adopted as the high-fidelity reference method. We first present the theoretical analysis of entropy stability for the Lax-Friedrichs scheme, emphasizing the fundamental role of entropy conservation in ensuring the physical admissibility of numerical solutions. After verifying that the Lax-Friedrichs scheme satisfies the entropy stability condition, we focus on its applications to the inviscid Burgers equation and on its use as a high-fidelity solver for the construction of reduced order models.

In Chapter 3, we propose a reduced order model for the Burgers equation and investigate whether the entropy stability property can be preserved within this reduced formulation. Numerical tests are then performed to assess the accuracy and to verify the entropy stability of the reduced order model.

Finally, Chapter 4 extends the discussion to the hyper-reduced regime with the choice of collocation points via the Nonnegative Least Squares algorithm, as it provides positive weights and a consistent selection of representative cells. The chapter presents the mathematical formulation of this hyper-reduced model and examines its entropy stability, concluding with a discussion of the limitations of the method and possible directions for future improvements.

Chapter 1

Numerical discretization methods

The development of model order reduction techniques comes from the need to efficiently approximate the dynamics of large-scale systems. When these systems are discretised using classical high-fidelity numerical schemes, they lead to very large algebraic problems, the solution of which can be computationally expensive or even unfeasible in many practical applications.
This chapter begins by recalling the high-fidelity discretisation strategies that form the bases of reduced formulations. We introduce commonly used high-fidelity schemes for approximating transport and diffusion equations, highlighting their accuracy and limitations in terms of computational cost. Next, we present the collocation-based reduced order model, which, unlike standard projection methods that require the evaluation of all points in the domain, it limits the evaluation to a subset of selected points in the spatial domain, reducing computational costs without compromising accuracy.

1.1 High-Fidelity Methods

In high-fidelity methods for the numerical solution of partial differential equations, the accuracy of the solution is guaranteed by high-order extrapolation of the discrete solution, or by a fine discretisation of the domain, with an error that typically scales as $O(h^{\alpha})$, where h denotes the grid spacing and α is the order of accuracy of the numerical scheme. The high computational cost of these approaches arises both from the fine spatial and temporal resolution required to capture detailed features of the solution and from the use of high-order approximation spaces, for example in finite element methods that employ Lagrange basis functions with very high polynomial degrees. These methods are particularly suitable for accurately modeling complex phenomena, but their computational expense often results in prohibitively long simulation times.
This work initially uses a high-fidelity numerical method based on the upwind scheme for the transport equation, Cranck-Nicolson for the diffusion equation and Lax-Friedrichs for the Burger's equation.

1.1.1 Upwind method for linear transport equation

The finite volume method is adopted for the spatial discretisation of conservation laws. The computational domain $[0, L]$ is divided into N uniform cells of width Δx and the cell-averaged solution is defined as:

$$u_k(t) = \frac{1}{\Delta x} \int_{x_{k-\frac{1}{2}}}^{x_{k+\frac{1}{2}}} u(x,t)\, dx. \tag{1.1}$$

Integrating the conservation law in time and space yields the semi-discrete formulation:

$$u_k^{n+1} = u_k^n - \frac{\Delta t^n}{\Delta x} \left(f_{k+\frac{1}{2}}^n - f_{k-\frac{1}{2}}^n \right), \tag{1.2}$$

where $f_{k+\frac{1}{2}}^n$ denotes the numerical flux at the cell interfaces [4].

For example, in the case of the linear advection equation, the Upwind flux is obtained by setting:

$$f_{k+\frac{1}{2}} = a\, u_k, \qquad f_{k-\frac{1}{2}} = a\, u_{k-1}. \tag{1.3}$$

The corresponding space-time discretisation reads:

$$u_i^{n+1} = u_i^n - \frac{a\, \Delta t}{\Delta x}(u_i^n - u_{i-1}^n), \tag{1.4}$$

where a is the transport velocity and Δx, Δt denote the spatial and temporal steps, respectively [11].

As the method is explicit, the stability condition is governed by the Courant-Friedrichs-Lewy number, abbreviated as CFL:

$$CFL = \frac{a\, \Delta t^n}{\Delta x} \leq 1. \tag{1.5}$$

In the simulations, a value of $CFL = 0.9$ is adopted and the time step is computed as:

$$\Delta t^n = CFL \cdot \frac{\Delta x}{|a|}. \tag{1.6}$$

Numerical results

The first physical model considered is the linear transport equation, expressed as:

$$\begin{cases} \dfrac{\partial u}{\partial t} + a \dfrac{\partial u}{\partial x} = 0, & (x,t) \in [0, L] \times [0, T], \\ u(x,0) = f(x), & x \in [0, L], \end{cases} \tag{1.7}$$

where $u(x,t)$ denotes the transported quantity and $a \neq 0$ is the advection velocity, which here is considered positive. The analytical solution is the following:

$$u(x,t) = f(x - at), \tag{1.8}$$

where $f(x)$ represents the initial condition.

In this analysis, a compact support function is considered as the initial condition:

$$u(x,0) = f(x) = \begin{cases} A \cdot e^{-\alpha(x-x_0)^2}, & \text{if } |x - x_0| \leq R, \\ 0, & \text{otherwise,} \end{cases} \tag{1.9}$$

where A is the amplitude, x_0 the centre, R the radius and α the decay coefficient.

The numerical simulations are performed with advection velocity $a = 1$, domain length $L = 4$ and $N = 4000$ uniform cells. A compact support function centered at $x_0 = 1$ with parameters $R = 1$, $A = 1$ and $\alpha = 10$ is adopted as the initial condition. The Courant number is set to $CFL = 0.9$, ensuring stability of the explicit Upwind scheme. The final simulation time is $T = 1.5$.

in Figure 1.1 is shown the time evolution of the numerical solution, compared to the exact analytical translation $u(x,t) = f(x - at)$. The transported pulse maintains its overall shape while moving to the right with constant speed $a = 1$. A slight numerical diffusion can be observed, which is characteristic of the first-order Upwind scheme: the front of the pulse becomes smoother over time and the amplitude decreases gradually.

To assess the accuracy of the method, the L^2 norm of the error with respect to the exact solution is computed at 40 equally spaced snapshot times. In Figure 1.2 is shown the evolution of the numerical error. The L^2 error remains on the order of 10^{-4} throughout the simulation, confirming that the Upwind method accurately reproduces the physical advection of the initial profile. This behavior is consistent with the characteristics of first-order finite volume Upwind schemes, which are first-order accurate in space.

1.1.2 Crank-Nicolson method for diffusion equation

The second mathematical model considered is a parabolic linear diffusion equation:

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2},$$

where $u = u(x,t)$ represents the state variable, while $D > 0$ is the diffusion coefficient. We assume a spatial domain $x \in [0,1]$ and a final time $T = 1.5$. The problem is subject to homogeneous Dirichlet boundary conditions, i.e. $u(0,t) = u(1,t) = 0$, while the initial condition is defined as a combination of sinusoidal functions:

$$u(x,0) = \sin(\pi x) + \sin(2\pi x).$$

The exact solution of the problem is given by:

$$u(x,t) = \sin(\pi x)e^{-\pi^2 Dt} + \sin(2\pi x)e^{-4\pi^2 Dt}.$$

For temporal discretisation is used the implicit Crank-Nicolson scheme [5]. This method is based on a time-centered discretization, where the spatial derivative is evaluated as the

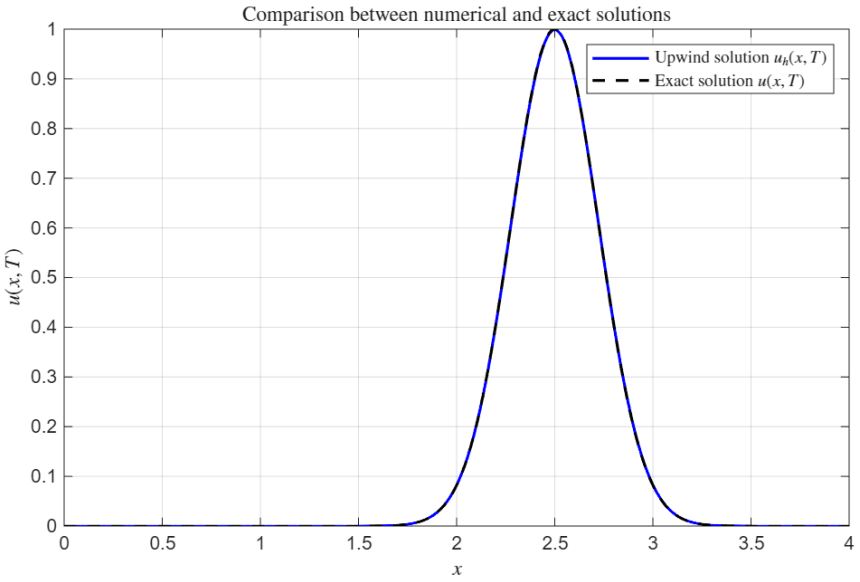


Figure 1.1: Snapshots of the solution obtained with the Upwind scheme for the linear transport equation. The numerical solution (blue) is compared with the analytical solution (black dashed).

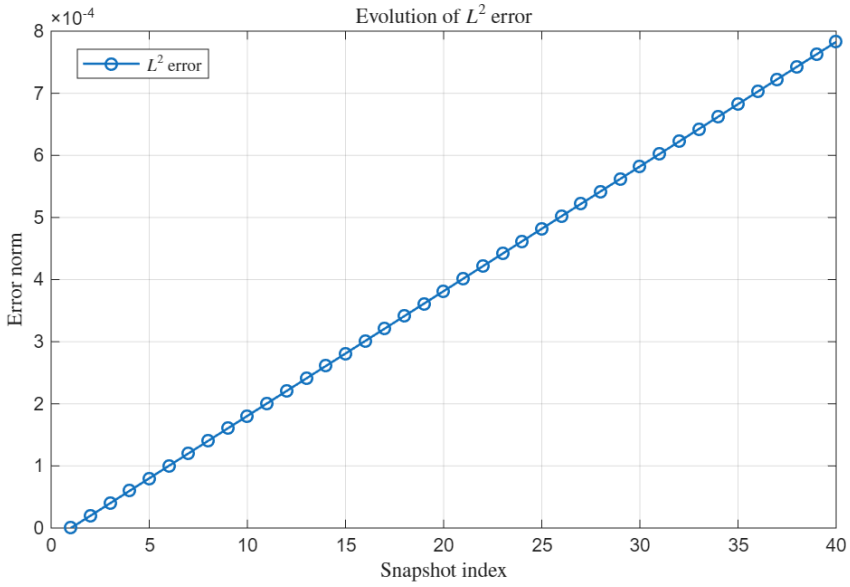


Figure 1.2: Evolution of the L^2 error during the simulation. The Upwind method shows stable and monotonic error growth consistent with first-order accuracy.

average between time levels n and $n+1$, leading to the following update formula:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{D}{2\Delta x^2}\left(u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1} + u_{i+1}^n - 2u_i^n + u_{i-1}^n\right).$$

Spatial discretisation is obtained using the second-order central finite difference method [4]. The resulting set of equations is solved by means of a linear system, obtained by reformulating the relationship in matrix form:

$$\mathbf{A}\,\mathbf{u}^{n+1} = \mathbf{B}\,\mathbf{u}^{n}, \tag{1.10}$$

where the matrices $\mathbf{A}$ and $\mathbf{B}$ are tridiagonal and depend on the parameter

$$r = \frac{D\Delta t}{2\Delta x^2}.$$

Specifically, their structure is given by:

$$\mathbf{A} = \begin{bmatrix} 1+2r & -r & & & 0 \\ -r & 1+2r & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 1+2r & -r \\ 0 & & & -r & 1+2r \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 1-2r & r & & & 0 \\ r & 1-2r & r & & \\ & \ddots & \ddots & \ddots & \\ & & r & 1-2r & r \\ 0 & & & r & 1-2r \end{bmatrix}.$$

To ensure high spatial resolution, the domain was divided into $N = 998$ internal points, with $\Delta x = \frac{1}{N+1}$. The time step was chosen to be proportional to Δx^2, according to:

$$\Delta t = \frac{\Delta x^2}{2D},$$

in order to ensure stability [6].
The numerical error was monitored with respect to the exact solution using the L_2 norm. The results reported in Table 1.2 show that the Crank-Nicolson scheme provides a numerically accurate solution over time, with errors remaining small and consistent throughout the simulation.

Numerical results

The numerical simulations are performed with diffusion coefficient $D = 1$, domain length $L = 1$ and $N = 1000$ grid points. The temporal step is chosen as $\Delta t = \frac{\Delta x^2}{2D}$, ensuring the numerical stability of the implicit Crank-Nicolson scheme. The final integration time is $T = 1.5$ and the homogeneous Dirichlet boundary conditions $u(0,t) = u(1,t) = 0$ are imposed.

The initial condition is defined as the superposition of two sinusoidal modes:

$$u(x,0) = \sin(\pi x) + \sin(2\pi x),$$

for which the analytical solution is known and given by:

$$u(x,t) = \sin(\pi x)e^{-\pi^2 Dt} + \sin(2\pi x)e^{-4\pi^2 Dt}.$$

Figure 1.3 illustrates the time evolution of the numerical solution obtained with the Crank-Nicolson method compared to the exact analytical solution.

The numerical solution accurately reproduces this behavior and shows good matching with the analytical profile.

The accuracy of the scheme is quantified through the L^2 norm of the error between the numerical and exact solutions, computed at 40 equally spaced snapshot times. Figure 1.4 reports the evolution of the error over time. The L^2 error remains on the order of 10^{-6}, confirming the second-order accuracy in both space and time characteristic of the Crank-Nicolson discretization.

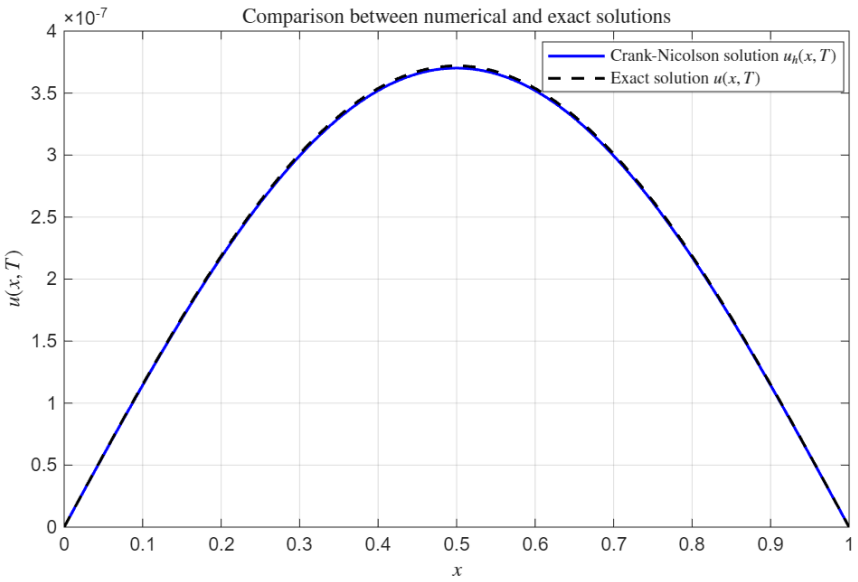


Figure 1.3: Snapshots of the solution obtained with the Crank–Nicolson method for the diffusion equation. The numerical solution (blue) is compared with the exact analytical solution (black dashed).

1.2 Model Order Reduction tecniques

Before delving into the specific methodologies for model order reduction, it is important to recall that high-fidelity simulations, while accurate, often involve a very large number of degrees of freedom and consequently require substantial computational resources. This becomes particularly limiting in the case of complex problems with many parameters. MOR techniques aim to alleviate these limitations by constructing a lower dimensional model that accurately preserves the dynamics of the system while drastically reducing the computational cost.

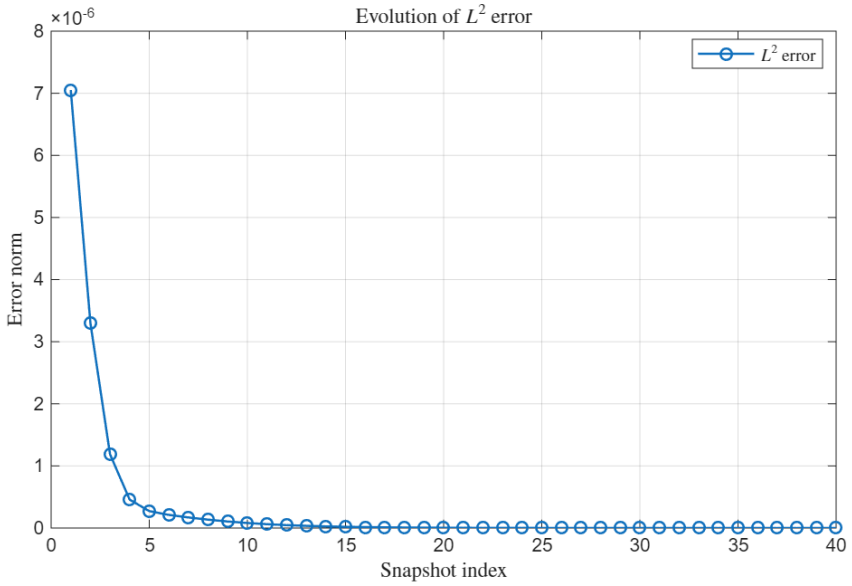


Figure 1.4: Evolution of the L^2 error during the simulation. The Crank-Nicolson method shows excellent agreement with the analytical solution and stable second-order accuracy.

The general process is structured into two main phases:

- Offline phase: In this phase, which requires a high computational cost, but is performed only once, the fundamental informations of the system are extracted and analysed, collecting a series of snapshots representative of the problem through a high fidelity approach.

- Online phase: Once the offline phase is complete, this phase allows solutions for new parameters to be calculated with reduced computational cost. Although the operations are repeated for each time step and parameter configuration, the calculation time is significantly lower than with the high fidelity approach.

1.2.1 Collocation-based Reduced Order Model

Reduced order models aim to reduce the computational complexity of high-fidelity models by projecting the full-order solution onto a low-dimensional subspace spanned by a few representative basis functions. In the collocation-based approach, the reduced model is formulated and evaluated only at a subset of selected cells in the computational domain, significantly decreasing the computational cost while preserving the essential physical accuracy.

Proper Orthogonal Decomposition of High-Fidelity Snapshots

Let $V_h(\Omega) \subset L^2(\Omega)$ denote the finite-dimensional discrete space associated with the high-fidelity discretization of the problem over the spatial domain Ω. For each parameter instance $\boldsymbol{\mu}_j \in \mathcal{P}$ and discrete time instant $t_k \in [0, t_{\text{fin}}]$, the corresponding high-fidelity numerical solution is denoted by:

$$u_h(x; t_k, \boldsymbol{\mu}_j) \in V_h(\Omega). \tag{1.11}$$

We define the snapshot set as the collection of K solutions obtained for different times and parameter instances:

$$\mathcal{S} = \{s_l(x) = u_h(x; t_k(l), \boldsymbol{\mu}_j(l))\}_{l=1}^{K}. \tag{1.12}$$

Each snapshot is then centered with respect to a reference state $u_0(x)$, for example, the mean of all snapshots or the steady solution:

$$\bar{s}_l(x) = s_l(x) - u_0(x). \tag{1.13}$$

The goal of the POD is to identify a reduced subspace of dimension $N \ll \dim(V_h)$, denoted by

$$V_N(\Omega) = \operatorname{span}\{\varphi_1(x), \ldots, \varphi_N(x)\} \subset V_h(\Omega), \tag{1.14}$$

that best approximates the snapshot set in the least squares sense. The continuous optimization problem reads:

$$\min_{\{\varphi_i\}_{i=1}^{N}} \sum_{l=1}^{K} \|s_l - \mathcal{P}_{V_N} s_l\|^2_{L^2(\Omega)}, \quad \text{subject to} \quad (\varphi_i, \varphi_j)_{L^2(\Omega)} = \delta_{ij}, \tag{1.15}$$

where $\mathcal{P}_{V_N}$ denotes the L^2-orthogonal projection on V_N.

Discrete POD Problem

Let $\{x_i\}_{i=1}^{M}$ denote the centers of the computational cells composing the static subdomain $S \subset \Omega$ and let $\mathbf{W} = \operatorname{diag}(|\Omega_i|) \in \mathbb{R}^{M \times M}$ be the diagonal matrix containing the corresponding quadrature weights. The discrete high-fidelity solutions, referred to as snapshots, are arranged in the matrix:

$$\mathbf{U} = \begin{bmatrix} u_1(x_1) & \cdots & u_K(x_1) \\ \vdots & \ddots & \vdots \\ u_1(x_M) & \cdots & u_K(x_M) \end{bmatrix} \in \mathbb{R}^{M \times K}. \tag{1.16}$$

Similarly, the reduced bases matrix is defined as

$$\mathbf{\Phi} = \begin{bmatrix} \varphi_1(x_1) & \cdots & \varphi_N(x_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(x_M) & \cdots & \varphi_N(x_M) \end{bmatrix} \in \mathbb{R}^{M \times N}. \tag{1.17}$$

The discrete POD problem becomes

$$\min_{\mathbf{\Phi}\in\mathbb{R}^{M\times N}} \|\mathbf{U}-\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{W}\mathbf{U}\|^2_{F_\mathbf{W}}, \quad \text{subject to } \mathbf{\Phi}^T\mathbf{W}\mathbf{\Phi}=\mathbf{I}_N, \tag{1.18}$$

where the weighted Frobenius norm is defined as:

$$\|\mathbf{A}\|^2_{F_\mathbf{W}} = \mathrm{Tr}(\mathbf{A}^T\mathbf{W}\mathbf{A}). \tag{1.19}$$

By performing the Cholesky factorization $\mathbf{W}=\mathbf{L}\mathbf{L}^T$, we define the weighted snapshots:

$$\tilde{\mathbf{U}} = \mathbf{L}^T\mathbf{U}, \tag{1.20}$$

and compute the truncated singular value decomposition:

$$\tilde{\mathbf{U}} \approx \tilde{\mathbf{U}}^N\mathbf{\Sigma}^N(\mathbf{V}^N)^T. \tag{1.21}$$

According to the Eckart-Young theorem, the optimal reduced bases is given by:

$$\mathbf{\Phi} = \mathbf{L}^{-T}\tilde{\mathbf{U}}^N. \tag{1.22}$$

Projection and Collocation-Based Hyperreduction

Given the reduced bases matrix $\mathbf{\Phi}\in\mathbb{R}^{M\times N}$, the orthogonal projection of any high-fidelity vector $\mathbf{u}\in\mathbb{R}^M$ onto the reduced space is:

$$\mathbf{u}^r = \mathbf{\Phi}\mathbf{\Phi}^T\mathbf{W}\mathbf{u}. \tag{1.23}$$

However, evaluating this projection on all M cells may still be computationally expensive. To reduce cost, the collocated approach introduces a hyper-reduced projection, in which only a subset of $L \ll M$ collocation points is used:

$$\mathbf{u}^{r,\varepsilon} = \mathbf{\Phi}\tilde{\mathbf{\Phi}}^T\mathbf{W}_\varepsilon\mathbf{u}_\varepsilon = \mathcal{P}_S(\mathbf{u}_\varepsilon), \tag{1.24}$$

where:

- $\tilde{\mathbf{\Phi}}\in\mathbb{R}^{L\times N}$ is the restriction of the bases matrix $\mathbf{\Phi}$ to the selected collocation points;
- $\mathbf{W}_\varepsilon\in\mathbb{R}^{L\times L}$ is the diagonal matrix of empirical quadrature weights;
- $\mathbf{u}_\varepsilon\in\mathbb{R}^L$ is the high-fidelity solution evaluated only at the selected points;
- the tolerance parameter ε controls the trade-off between accuracy and sparsity.

Empirical weights $\mathbf{W}_\varepsilon$ and associated collocation points are determined by solving a Nonnegative Least Squares problem:

$$\min_{\mathbf{w}\geq 0} \|\mathbf{G}\mathbf{w}-\mathbf{d}\|_2^2, \tag{1.25}$$

where $\mathbf{G}$ collects the evaluations of the products $u_s(x_i)\varphi_n(x_i)$ over all snapshots and basis functions and $\mathbf{d}$ represents their corresponding projection integrals. This ensures that the reduced integration based on the selected points approximates the full projection within the prescribed tolerance ε.

As $\varepsilon \to 0$, the hyper-reduced projection $\mathbf{u}^{r,\varepsilon}$ converges to the full projection $\mathbf{u}^r$, with $L \to M$. Thus, the collocation-based reduced model provides a computationally efficient approximation of the full projection, while maintaining a controllable level of accuracy.

1.2.2 Nonnegative Least Squares based hyper-reduction method

In model reduction for nonlinear systems, the POD based methodology is considered one of the most effective tools for generating efficient reduced bases. However, the POD approach requires the complete reconstruction of the original function, which is then projected onto the reduced bases, with a high computational cost.
When the nonlinear function $f_h[\tilde{u}_h](x,t;\mu)$ exhibits non-polynomial nonlinearities or when the precomputation-based approach is computationally prohibitive, hyper-reduction techniques are used. In this context, we propose an approach that exploits the NNLS technique. In this scheme, the vector of coefficients of the already projected nonlinear function is estimated directly, thus eliminating the need for a complete reconstruction of the original function. The main advantage of this method is that it operates entirely in reduced space, allowing for a significant reduction in computational load compared to POD and with a marginal loss of accuracy that makes the cost-benefit trade-off advantageous for more complex non-linear problems.

The basic idea is to replace the integral scalar product, which characterises the projection in POD mode, with a weighted sum over a discrete set of quadrature points. In this way, we have:

$$\langle f_h[u_h](x;t;\mu), \Phi_n(x)\rangle_\Theta \approx \sum_{i=1}^{L} w_i f_h[u_h](x_i;t;\mu)\Phi_n(x_i),$$

where x_i represent the quadrature points and $w_i > 0$ are the associated weights, with the matrix Θ assumed to be diagonal to simplify the calculations. The quadrature points and weights are optimised during the training phase in order to best approximate the exact scalar product.

To make the formulation more explicit, we define the matrix:

$$\underbrace{\begin{bmatrix} F_{1,1}[\tilde{u}_h](t;\mu) & \dots & F_{N,1}[\tilde{u}_h](t;\mu) \\ \vdots & \ddots & \vdots \\ F_{1,M}[\tilde{u}_h](t;\mu) & \dots & F_{N,M}[\tilde{u}_h](t;\mu) \end{bmatrix}}_{=\mathbf{F}[u_h](t;\boldsymbol{\mu})} \underbrace{\begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_N \end{pmatrix}}_{=\mathbf{w}} \approx \underbrace{\begin{pmatrix} \langle f_h[\tilde{u}_h](x;t;\mu), \Phi_1(x)\rangle_\Theta \\ \vdots \\ \langle f_h[\tilde{u}_h](x;t;\mu), \Phi_M(x)\rangle_\Theta \end{pmatrix}}_{=\mathbf{c}[\tilde{u}_h](t;\boldsymbol{\mu})}$$

with $F_{i,n}[u_h](t;\mu) = f_h[u_h](x_i;t;\mu)\Phi_n(x_i)$.

In a further step, we define:

$$\underbrace{\begin{bmatrix} F[s_1](t_k(1);\mu_j(1)) \\ F[s_2](t_k(2);\mu_j(2)) \\ \vdots \\ F[s_K](t_k(K);\mu_j(K)) \end{bmatrix}}_{=\mathbf{G}} \underbrace{\begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_N \end{pmatrix}}_{=\mathbf{w}} \approx \underbrace{\begin{bmatrix} c[s_1](t_k(1);\mu_j(1)) \\ c[s_2](t_k(2);\mu_j(2)) \\ \vdots \\ c[s_K](t_k(K);\mu_j(K)) \end{bmatrix}}_{=\mathbf{d}},$$

which expresses the problem of determining the optimal weights $\mathbf{w}_i$ for quadrature. This relationship can be written in compact form as:

$$\mathbf{G}\mathbf{w} \approx \mathbf{d}.$$

The determination of the weights is formulated as an optimisation problem:

$$\begin{cases} \underset{\mathbf{w}\in\mathbb{R}_+^N}{\text{minimize}} & \|\mathbf{w}\|_0 \\ \text{subject to} & \|\mathbf{G}\mathbf{w}-\mathbf{d}\|_2 \leq \epsilon\|\mathbf{d}\|_2, \end{cases}$$

where the norm $\|\cdot\|_0$ aims to reduce the number of non-zero weights and the constraint on the error ensures a good approximation of the exact scalar product. This problem is computationally very demanding, so it is solved using the NNLS algorithm, which imposes non-negativity on the weights and minimises the quadratic error between the product $\mathbf{G}\mathbf{w}$ and the data vector $\mathbf{d}$.

The weights $\tilde{\mathbf{w}}$ are finally obtained by keeping only the non-zero components of the solution to the minimisation problem.

Finally, in the predictive phase, $\Phi^T\Theta f_h[u_h](t;\mu)$ is approximated as:

$$\Phi^T\Theta f_h[u_h](t;\mu) \approx (P\Phi)^T \begin{bmatrix} \tilde{\mathbf{w}}_1 & 0 & \dots & 0 \\ 0 & \tilde{\mathbf{w}}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \tilde{\mathbf{w}}_L \end{bmatrix} \begin{bmatrix} f_h[u_h](\tilde{x}_1,t;\mu) \\ \vdots \\ f_h[u_h](\tilde{x}_L,t;\mu) \end{bmatrix},$$

where P represents the matrix that identifies the indices of the quadrature points [7].

1.3 Comparative analysis: transport and diffusion equations

In this section we analyse and compare the results obtained for transport and diffusion equations, using model reduction techniques based on POD and hyper-reduction methods. The aim is to evaluate the effectiveness of the reduced strategies adopted.

1.3.1 Selection of the optimal number of modes via RIC

A fundamental aspect of the POD method is choosing the optimal number of basis functions M, which is determined using Relative Information Content, abbreviated as RIC, defined as:

$$RIC(M) = \frac{\sum_{n=1}^{M} \sigma_n^2}{\sum_{n=1}^{\min(N,K)} \sigma_n^2},$$

where σ_n^2 are the squared singular values obtained from the SVD decomposition of the weighted snapshot matrix. The value of RIC represents the fraction of energy of the solution captured by the first M bases.

A common criterion for choosing M is to find the minimum value such that $RIC(M)$ exceeds a predefined threshold. In our case, the numerical results show that for $RIC(M) \geq 0.999$, the optimal value of the reduced bases is $M = 10$ in the case of the transport equation and $M = 2$ for the diffusion equation. This implies that the first 10 bases in the first case and the first 2 in the second capture 99.9% of the total energy of the system, allowing for a significant reduction in dimensionality without compromising the accuracy of the solution. The use of this reduction allows the computational cost of the online phase to be reduced, while maintaining an excellent approximation of the original snapshots. Furthermore, the evaluation of the projection error confirmed that the loss of information for $M = 10$ and $M = 2$ respectively is negligible compared to configurations with a larger number of bases.

To visualise the reconstruction trend for different bases sizes in Figure 1.5 are shown the reconstructions obtained using the optimal bases: on the left for the transport equation and on the right for the diffusion equation.

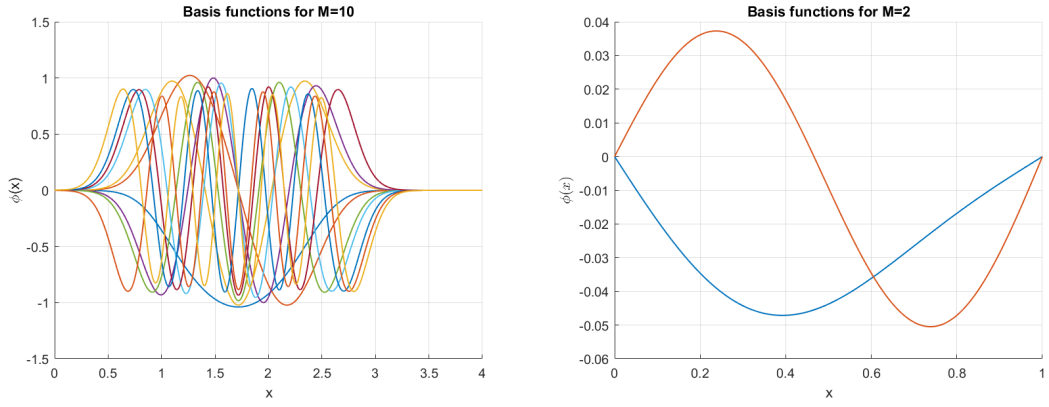


Figure 1.5: Reconstruction with optimal bases: $M = 10$ for transport equation (left) and $M = 2$ for diffusion equation (right).

1.3.2 SVD analysis

SVD is used to construct the reduced bases via POD. Figure 1.6 shows the singular values obtained for both problems.

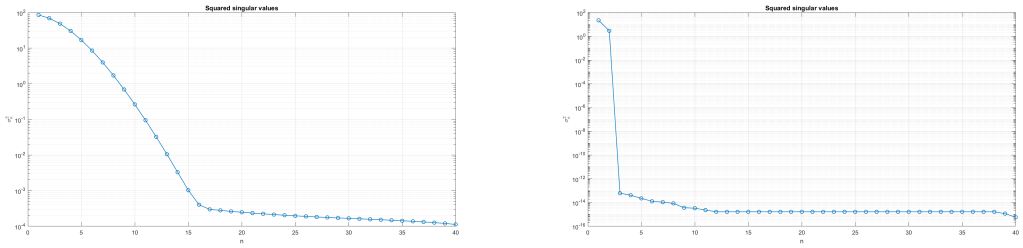


Figure 1.6: Singular value decomposition for transport (left) and diffusion (right).

In the case of diffusion, the squared singular values σ_n^2 decrease very rapidly, reaching a value of the order of 10^{-14} after only three modes. This value indicates that almost all of the system's energy, or information content, is captured by the first two modes.

This difference is related to the nature of the two equations: the diffusion equation, due to diffusion itself, favours rapid attenuation of high-frequency components, while the transport equation allows waves to be conserved without dissipation, leading to a slower decrease in singular values.
This result confirms that compression using POD is more effective for problems dominated by diffusion than for transport problems dominated by front propagation.

1.3.3 Relative projection error with standard POD

To further assess the effectiveness of the reduced compared to the high-fidelity solution, we perform an analysis of the relative projection error, which quantifies how accurately the reduced POD bases can reconstruct the original snapshots. The results obtained for different sizes of the bases M are shown in Table 1.1 for transport and in Table 1.2 for diffusion.

M	**Relative projection error**
1	5.3497×10^{-1}
5	5.8270×10^{-3}
10	6.4007×10^{-7}
15	6.5685×10^{-11}
20	3.6761×10^{-11}

Table 1.1: Relative projection error for the transport equation with respect to high fidelity solution.

M	**Relative projection error**
1	1.7192×10^{-2}
2	1.3630×10^{-29}
3	6.2696×10^{-30}
4	2.8988×10^{-30}
5	1.9069×10^{-30}

Table 1.2: Relative projection error for the diffusion equation with respect to high fidelity solution.

We observe that for the transport equation, as M increases, the error decreases, reaching zero in machine precision for $M = 40$. This result is consistent since the total number of snapshots used to construct the POD bases is 40: when $M = 40$, the reduced space is able to accurately represent all the original snapshots, completely eliminating the projection error. For lower values of M, the reduced bases is unable to capture all the variability of the system, leading to a higher error. However, even with $M = 10$, the error is extremely low, suggesting that most of the dynamic information is concentrated in the first few principal modes.
In the case of the diffusion equation we see that the reduced bases method works much better as the error is already of the order of 10^{-30} at the second bases.

1.3.4 Relative error after hyper-reduction with NNLS

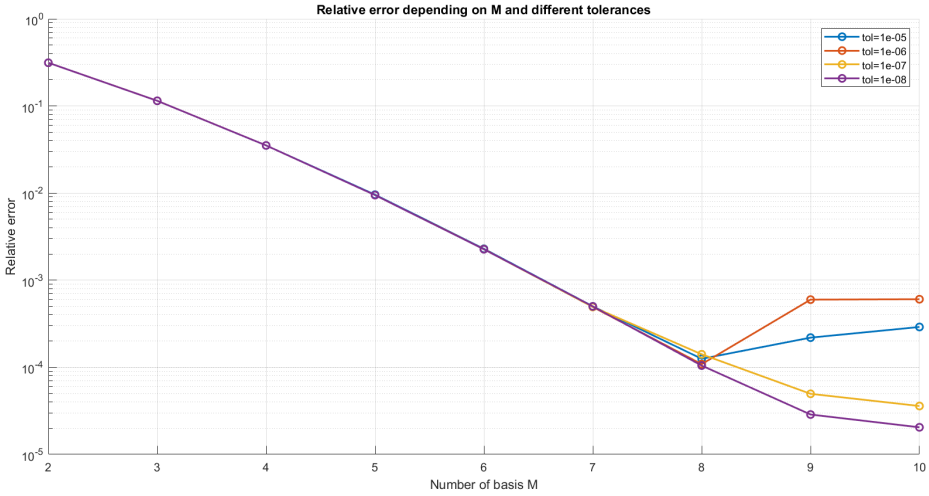


Figure 1.7: Relative error in the transport equation between the reduced solution and the high-fidelity solution for different tolerances in the NNLS solution as a function of the number of bases M.

The graph represented in Figure 1.7 is shown to evaluate the effectiveness of order reduction via POD in the transport equation. It shows the relative error between the approximate solution projected in the reduced space and the exact solution, as a function of the number of modes M and for different tolerances imposed during the solution of the NNLS system for the calculation of the empirical weights.

The weights $\mathbf{w}$ used in the reduced order model are computed as follows:

1. Given a set of snapshots $\{s_l(x)\}_{l=1}^{K}$, the reduced bases $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M]$ of size $N \times M$ is obtained via singular value decomposition of the snapshot matrix:

$$\text{snapshots}^T = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \quad \mathbf{\Phi} = \mathbf{U}(:, 1:M),$$

   where the first M left singular vectors are selected.

2. For each snapshot $\mathbf{u}^{(l)}$, a linear system is constructed to enforce the collocation condition:

$$\mathbf{G}\mathbf{w} \approx \mathbf{d},$$

   where $\mathbf{G} \in \mathbb{R}^{MK \times N}$ and $\mathbf{d} \in \mathbb{R}^{MK}$ are defined as

$$\mathbf{G} = \begin{bmatrix} (\mathbf{u}^{(1)} \odot \boldsymbol{\phi}_1)^T & \cdots & (\mathbf{u}^{(1)} \odot \boldsymbol{\phi}_M)^T \\ \vdots & & \vdots \\ (\mathbf{u}^{(K)} \odot \boldsymbol{\phi}_1)^T & \cdots & (\mathbf{u}^{(K)} \odot \boldsymbol{\phi}_M)^T \end{bmatrix}, \qquad \mathbf{d} = \begin{bmatrix} \sum \mathbf{u}^{(1)} \boldsymbol{\phi}_1 \\ \vdots \\ \sum \mathbf{u}^{(K)} \boldsymbol{\phi}_M \end{bmatrix}.$$

   Here, $\odot$ denotes the element-wise multiplication.

3. The weights $\mathbf{w}$ are then obtained by solving the NNLS problem

$$\mathbf{w} = \arg\min_{\mathbf{w} \geq 0} \|\mathbf{G}\mathbf{w} - \mathbf{d}\|_2^2,$$

   ensuring non-negativity.

A relevant portion of the Matlab implementation is shown below:

```
[U, ~, ~] = svd(snapshots', 'econ');
phi = U(:, 1:M);

% Construct G and d
G = zeros(M*Ns, N);
d = zeros(M*Ns, 1);
for sn = 1:Ns
    u_sn = snapshots(sn, :)';
    for i = 1:M
        G_sn(i, :) = u_sn' .* phi(:, i)';
        d_sn(i) = sum(u_sn .* phi(:, i));
    end
    G((sn-1)*M+1:sn*M, :) = G_sn;
    d((sn-1)*M+1:sn*M) = d_sn;
```

```
end

% Solve the Nonnegative Least Squares problem
options = optimset('TolX', 1e-8);
w_tilde = lsqnonneg(G, d, options);
```

From Figure 1.7 can be observed that:

- as the number of bases M increases, the error decreases, indicating that a larger number of modes allows for a more accurate capture of the dynamics of the solution;
- for a fixed M, a tighter tolerance (e.g., 10^{-8} compared to 10^{-5}) in the NNLS algorithm leads to a smaller error, since the weighting system is solved with greater precision, maintaining consistency with the data structure.

These results confirm that the POD technique combined with positive weights allows for a good approximation even with a small number of degrees of freedom, provided that the numerical resolution of the optimization is accurate.

We now apply the hyper-reduction procedure by enforcing a non-negativity constraint on the quadrature weights. In particular, we solve a NNLS problem to determine the reduced integration weights $\mathbf{w}_{\tilde{j}} \geq 0$. This approach ensures that all weights are physically meaningful, since they play the role of integration coefficients in the reduced model and prevents the occurrence of negative weights that could lead to non-physical contributions in the projection or instability in the reduced dynamics.

The NNLS problem is formulated as:

$$\min_{\mathbf{w}_{\tilde{j}} \geq 0} \|\mathbf{G}\mathbf{w}_{\tilde{j}} - \mathbf{d}\|_2^2,$$

where $\mathbf{G}$ and $\mathbf{d}$ are constructed from the snapshot matrix and the selected POD modes. The imposition of the nonnegativity constraint modifies the least squares solution slightly, producing weights that are close to, but not exactly identical to, the unconstrained optimal ones. As a result, a small residual projection error remains even when the number of modes M is large.

1.3.5 Comparison of projection approaches

We now compare the relative errors obtained, in comparison with the high-fidelity solution using four different projection approaches, with and without hyper-reduction for both equations.

- P_1: Reduction based on standard POD with unit weight;
- P_2: Hyper-reduction method;

- P_3: POD method with optimized weight matrix;
- P_4: Hyper-reduction method with optimized weight matrix.

All these approaches are based on a projection of the solution in the reduced space, defined by a projection operator $\mathbf{P}$. The four different projections used in the various cases are as follows:

$$P_1 = \mathbf{\Phi}\mathbf{\Phi}^T I, \tag{1.26}$$

$$P_2 = \mathbf{\Phi}\mathbf{\Phi}^T \mathbf{W}_1^\epsilon, \tag{1.27}$$

$$P_3 = \mathbf{\Phi}\mathbf{\Phi}^T \mathbf{W}_3, \tag{1.28}$$

$$P_4 = \mathbf{\Phi}\mathbf{\Phi}^T \mathbf{W}_3^\epsilon. \tag{1.29}$$

where $\mathbf{W}_3 = \Delta x \cdot I$ denotes the diagonal matrix of Gauss quadrature weights.

To evaluate the quality of the model reduction, we calculate the relative error between the exact solution u_{exact} and the approximate solution u_{approx} as follows:

$$\text{Relative error} = \frac{\int_0^{t_{\max}} \int_0^L (u_{\text{exact}}(x,t) - u_{\text{approx}}(x,t))^2 \, dx \, dt}{\int_0^{t_{\max}} \int_0^L u_{\text{exact}}(x,t)^2 \, dx \, dt}. \tag{1.30}$$

The numerical discretization of this formula was implemented in the code through the summation over spatial and temporal points weighted by the step Δx and Δt.

The relative error values for each case are shown in Tables 1.3 for transport and 1.4 for diffusion.

Case	**Relative error**
1	$1.22026638576021 \times 10^{-5}$
2	$6.04311987385492 \times 10^{-4}$
3	$1.22026638569027 \times 10^{-5}$
4	$1.22870839982512 \times 10^{-5}$

Table 1.3: Comparison of relative errors for the transport equation for $M = 10$.

Case	**Relative error**
1	$9.69537441208694 \times 10^{-6}$
2	$9.69537441249129 \times 10^{-6}$
3	$9.69537441208335 \times 10^{-6}$
4	$9.69537441247759 \times 10^{-6}$

Table 1.4: Comparison of relative errors for the diffusion equation for $M = 2$.

Both tables show that (1.26) and (1.28) have very low and almost identical errors. In both cases we see that the use of the optimized weight matrix leads to a slightly lower

relative error than the standard POD approach with weight matrix I. As for the two hyper-reduced cases, (1.27) has a significantly higher error, suggesting that the use of weights estimated with lsqnonneg does not necessarily guarantee optimal error reduction; while (1.29) shows an error slightly higher than those of (1.26) and (1.28) but still much lower than (1.27), indicating that hyper-reduction does not significantly compromise the accuracy of the method if an optimized weight matrix is used.

Chapter 2

The high-fidelity approach

In this chapter, we present the high-fidelity discretisation employed to generate the snapshots used in the construction of the reduced order model for the inviscid Burgers equation. Our attention is devoted to the Lax-Friedrichs method, for which we derive entropy stability properties and examine its numerical behaviour.

The inviscid Burgers equation is a nonlinear conservation law frequently adopted as a benchmark problem for analysing shock formation and nonlinear wave propagation. In one dimension it can be written in conservative form as:

$$\begin{aligned} u_t + f(u)_x &= 0, \\ f(u) &= \frac{1}{2}u^2. \end{aligned} \tag{2.1}$$

This equation describes the nonlinear transport of a scalar quantity and exhibits the development of shocks even when starting from smooth initial conditions. Due to its simplicity and rich dynamics, Burgers equation is particularly suitable for analysing the behaviour of numerical schemes and assessing the effectiveness of reduced order models.

2.1 The Lax-Friedrichs method

Among the various discretization techniques for conservation laws, the Lax-Friedrichs method provides a simple yet effective first order approach. It belongs to the class of explicit, dissipative schemes, where numerical diffusion is introduced to stabilize the solution in the presence of discontinuities.

The scheme is defined as:

$$u_i^{n+1} = \frac{1}{2}\left(u_{i+1}^n + u_{i-1}^n\right) - \frac{\Delta t}{2\Delta x}\left(f(u_{i+1}^n) - f(u_{i-1}^n)\right), \tag{2.2}$$

where u_i^n approximates the solution $u(x_i, t^n)$ at cell center x_i at discrete time t^n.

Equation (2.2) can be equivalently rewritten in conservative form by introducing a numerical flux function:

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}\right), \tag{2.3}$$

where the intercell numerical fluxes are defined as:

$$\hat{f}_{i+\frac{1}{2}} = \frac{1}{2}\left(f(u_i^n) + f(u_{i+1}^n)\right) - \frac{\alpha}{2}(u_{i+1}^n - u_i^n), \tag{2.4}$$

with $\alpha \geq \max_u |f'(u)| = \max_u |u|$.

The first term in (2.4) is the central flux, consistent with the physical flux $f(u)$, while the second term is a dissipative contribution proportional to the jump in u across the interface. This artificial viscosity stabilizes the scheme, particularly near steep gradients or discontinuities.

The Lax-Friedrichs method does not require solving a Riemann problem at each interface and the dissipative term ensures that the scheme converges to an entropy admissible weak solution, at the cost of excessive numerical diffusion, which can smear out shocks and contact discontinuities [11].

In numerical simulations, transmissive boundary conditions are typically used:

$$u_0^n = u_1^n, \quad u_{M+1}^n = u_M^n,$$

allowing waves to exit the computational domain without artificial reflection.

The method is stable under the CFL condition:

$$\Delta t = \frac{C_{\text{CFL}}\,\Delta x}{\alpha}, \quad \alpha = \max_i |u_i^n|,$$

where $0 < C_{\text{CFL}} \leq 1$. In practice, $C_{\text{CFL}} \approx 0.9$ is a common choice for efficiency and stability.

2.2 Entropy stability of the Lax-Friedrichs scheme

Before employing Lax-Friedrichs as a reference solver for snapshot generation, it is essential to verify that this discretization satisfies the property of entropy stability.

We start by considering the scalar conservation law in one spatial dimension:

$$u_t + f(u)_x = 0, \qquad u(x,0) = u_0(x), \tag{2.5}$$

where $f : \mathbb{R} \to \mathbb{R}$ is sufficiently smooth [11].

To single out the physically relevant weak solution of (2.5), it is customary to introduce an entropy pair. Let $\eta : \mathbb{R} \to \mathbb{R}$ be a convex entropy function and let $q : \mathbb{R} \to \mathbb{R}$ denote the associated entropy flux. The pair (η, q) is required to satisfy the compatibility condition

$$q'(u) = \eta'(u) f'(u), \tag{2.6}$$

which ensures that (η, q) is consistent with the underlying conservation law [3].

To derive this condition, multiply (2.5) by $\eta'(u)$:

$$\eta'(u)u_t + \eta'(u)f'(u)u_x = 0.$$

Using (2.6), we rewrite the second term as $q'(u)u_x$, so that

$$\eta'(u)u_t + q'(u)u_x = 0.$$

Recognizing the chain rule identities $\eta_t = \eta'(u)u_t$ and $q_x = q'(u)u_x$, we obtain the entropy conservation law

$$\eta_t + q_x = 0. \tag{2.7}$$

For regular solutions, (2.7) is exactly equivalent to the original conservation law. However, once shocks form and weak solutions must be considered, (2.7) is replaced by the entropy inequality:

$$\eta(u)_t + q(u)_x \leq 0, \tag{2.8}$$

which expresses the decay of entropy. The inequality (2.8) plays a key role in selecting the physically admissible weak solution among the infinitely many weak solutions of (2.5) [11].

2.2.1 Discrete entropy inequality

We want to prove that the scheme (2.3) is entropy stable, i.e. that there exists a numerical entropy flux $Q_{i+1/2}$ such that:

$$\eta(u_i^{n+1}) - \eta(u_i^n) + \lambda(Q_{i+1/2} - Q_{i-1/2}) \leq 0, \tag{2.9}$$

for every i and n.

Starting from (2.3), we multiply by $\eta'(u_i^{n+1})$:

$$\eta'(u_i^{n+1})(u_i^{n+1} - u_i^n) = -\lambda\,\eta'(u_i^{n+1})(F_{i+1/2}^{LF} - F_{i-1/2}^{LF}). \tag{2.10}$$

Due to the convexity of η, the following inequality holds [9]:

$$\eta(u_i^{n+1}) - \eta(u_i^n) \leq \eta'(u_i^{n+1})(u_i^{n+1} - u_i^n). \tag{2.11}$$

Substituting (2.10) into (2.11), we obtain:

$$\eta(u_i^{n+1}) - \eta(u_i^n) \leq -\lambda\,\eta'(u_i^{n+1})(F_{i+1/2}^{LF} - F_{i-1/2}^{LF}). \tag{2.12}$$

At this point, one introduces a consistent numerical entropy flux $Q_{i+1/2}$ associated with the Lax-Friedrichs flux $F_{i+1/2}^{LF}$. If the dissipative term in F^{LF} ensures that the discrete inequality (2.9) is satisfied, the relation (2.11) establishes the entropy stability of the scheme.

2.2.2 Flow decomposition

We write the Lax-Friedrichs flow as the sum of a centered part plus a dissipative part [2]:

$$F^{LF}_{i+1/2} = F^{c}_{i+1/2} - \alpha(u^n_{i+1} - u^n_i), \qquad F^{c}_{i+1/2} = \frac{1}{2}\Big(f(u^n_i) + f(u^n_{i+1})\Big). \tag{2.13}$$

Then, the term on the right in (2.12) becomes:

$$\eta'(u^n_i)(F^{LF}_{i+1/2} - F^{LF}_{i-1/2}) = \eta'(u^n_i)(F^{c}_{i+1/2} - F^{c}_{i-1/2}) - \alpha\,\eta'(u^n_i)\Big[(u^n_{i+1} - u^n_i) - (u^n_i - u^n_{i-1})\Big]. \tag{2.14}$$

2.2.3 Numerical entropy flux

First we define a numerical entropy flux as [10]:

$$Q_{i+1/2} = \frac{1}{2}\Big(q(u^n_i) + q(u^n_{i+1})\Big) - \alpha\,\Psi(u^n_i, u^n_{i+1}), \tag{2.15}$$

where the corrective function Ψ is given by:

$$\Psi(u^n_i, u^n_{i+1}) = \frac{1}{2}(\eta'(u^n_{i+1}) - \eta'(u^n_i))(u^n_{i+1} - u^n_i). \tag{2.16}$$

With this definition, we intend to show how we arrive at the decomposition:

$$\eta'(u^n_i)(F^{LF}_{i+1/2} - F^{LF}_{i-1/2}) = (Q_{i+1/2} - Q_{i-1/2}) + D_i, \tag{2.17}$$

where the additional term D_i is dissipative in nature and is equal to

$$D_i = \frac{\alpha}{2}\Big[(\eta'(u^n_{i+1}) - \eta'(u^n_i))(u^n_{i+1} - u^n_i) + (\eta'(u^n_i) - \eta'(u^n_{i-1}))(u^n_i - u^n_{i-1})\Big]. \tag{2.18}$$

To prove this identity, we start from the expression:

$$\eta'(u^n_i)(F^{LF}_{i+1/2} - F^{LF}_{i-1/2}), \tag{2.19}$$

and substitute the definition of the Lax-Friedrichs flow. In fact, we have:

$$F^{LF}_{i+1/2} = \tfrac{1}{2}(f(u^n_i) + f(u^n_{i+1})) - \alpha(u^n_{i+1} - u^n_i),$$

$$F^{LF}_{i-1/2} = \tfrac{1}{2}(f(u^n_{i-1}) + f(u^n_i)) - \alpha(u^n_i - u^n_{i-1}),$$

so their difference is:

$$F^{LF}_{i+1/2} - F^{LF}_{i-1/2} = \frac{1}{2}\Big(f(u^n_{i+1}) - f(u^n_{i-1})\Big) - \alpha\Big[(u^n_{i+1} - u^n_i) - (u^n_i - u^n_{i-1})\Big]. \tag{2.20}$$

Multiplying this term by $\eta'(u_i^n)$ we obtain:

$$\eta'(u_i^n)(F_{i+1/2}^{LF} - F_{i-1/2}^{LF}) = \tfrac{1}{2}\eta'(u_i^n)\Big(f(u_{i+1}^n) - f(u_{i-1}^n)\Big) \\ - \alpha\,\eta'(u_i^n)\Big[(u_{i+1}^n - u_i^n) - (u_i^n - u_{i-1}^n)\Big]. \tag{2.21}$$

At this point, we introduce the discrete entropy flux. For definition (2.15), we have:

$$Q_{i+1/2} = \tfrac{1}{2}(q(u_i^n) + q(u_{i+1}^n)) - \alpha\,\Psi(u_i^n, u_{i+1}^n),$$

$$Q_{i-1/2} = \tfrac{1}{2}(q(u_{i-1}^n) + q(u_i^n)) - \alpha\,\Psi(u_{i-1}^n, u_i^n),$$

with $\Psi(u,v) = \frac{1}{2}(\eta'(v) - \eta'(u))(v-u)$. The difference between the two entropy flows is therefore:

$$Q_{i+1/2} - Q_{i-1/2} = \tfrac{1}{2}\Big(q(u_{i+1}^n) - q(u_{i-1}^n)\Big) \\ - \alpha\Big(\Psi(u_i^n, u_{i+1}^n) - \Psi(u_{i-1}^n, u_i^n)\Big). \tag{2.22}$$

To compare (2.21) and (2.22), we observe that, for the compatibility relation $q'(u) = \eta'(u)f'(u)$ and applying a trapezoidal symmetric quadrature, the following identity holds:

$$\tfrac{1}{2}\eta'(u_i^n)\Big(f(u_{i+1}^n) - f(u_{i-1}^n)\Big) = \tfrac{1}{2}\Big(q(u_{i+1}^n) - q(u_{i-1}^n)\Big).$$

Substituting in (2.21) we obtain:

$$\eta'(u_i^n)(F_{i+1/2}^{LF} - F_{i-1/2}^{LF}) = \tfrac{1}{2}\Big(q(u_{i+1}^n) - q(u_{i-1}^n)\Big) \\ - \alpha\,\eta'(u_i^n)\Big[(u_{i+1}^n - u_i^n) - (u_i^n - u_{i-1}^n)\Big]. \tag{2.23}$$

A comparison between (2.23) and (2.22) shows that their difference is exactly the dissipative term D_i. In fact, subtracting (2.22) from (2.23) we obtain:

$$D_i = -\alpha\,\eta'(u_i^n)\Big[(u_{i+1}^n - u_i^n) - (u_i^n - u_{i-1}^n)\Big] \\ + \alpha\Big(\Psi(u_i^n, u_{i+1}^n) - \Psi(u_{i-1}^n, u_i^n)\Big). \tag{2.24}$$

Now explicitly expanding the terms Ψ, i.e.

$$\Psi(u_i^n, u_{i+1}^n) = \tfrac{1}{2}(\eta'(u_{i+1}^n) - \eta'(u_i^n))(u_{i+1}^n - u_i^n),$$

$$\Psi(u_{i-1}^n, u_i^n) = \tfrac{1}{2}(\eta'(u_i^n) - \eta'(u_{i-1}^n))(u_i^n - u_{i-1}^n),$$

and substituting in (2.24), we obtain:

$$D_i = -\alpha\,\eta'(u_i^n)(u_{i+1}^n - u_i^n) + \alpha\,\eta'(u_i^n)(u_i^n - u_{i-1}^n) \\ + \tfrac{\alpha}{2}(\eta'(u_{i+1}^n) - \eta'(u_i^n))(u_{i+1}^n - u_i^n) \\ - \frac{\alpha}{2}(\eta'(u_i^n) - \eta'(u_{i-1}^n))(u_i^n - u_{i-1}^n). \tag{2.25}$$

By grouping and simplifying the common terms, we obtain the final formula:

$$D_i = \frac{\alpha}{2}\Big[(\eta'(u_{i+1}^n) - \eta'(u_i^n))(u_{i+1}^n - u_i^n) + (\eta'(u_i^n) - \eta'(u_{i-1}^n))(u_i^n - u_{i-1}^n)\Big], \tag{2.26}$$

which coincides with what we wanted.

As for the sign of D_i, we note that, since η is a convex function, the following always holds:

$$(\eta'(v) - \eta'(u))(v - u) \geq 0 \qquad \forall u, v \in \mathbb{R}.$$

It follows that every term that appears in the sum is non-negative and therefore:

$$D_i \geq 0. \tag{2.27}$$

This shows that D_i is dissipative.

Finally, inserting the decomposition (2.17) into the entropy evolution relation

$$\eta(u_i^{n+1}) - \eta(u_i^n) \leq -\lambda\, \eta'(u_i^n)(F_{i+1/2}^{LF} - F_{i-1/2}^{LF}), \tag{2.28}$$

we obtain

$$\eta(u_i^{n+1}) - \eta(u_i^n) \leq -\lambda\Big[(Q_{i+1/2} - Q_{i-1/2}) + D_i\Big]. \tag{2.29}$$

At this point, thanks to non-negativity (2.27), it immediately follows that:

$$\eta(u_i^{n+1}) - \eta(u_i^n) \leq -\lambda(Q_{i+1/2} - Q_{i-1/2}), \tag{2.30}$$

which coincides with the desired discrete entropy inequality.

2.3 Numerical results

After verifying that the Lax-Friedrichs scheme satisfies the entropy stability condition, we now apply it to the inviscid Burgers equation.
In the first example, we analyse a classical test case to observe and discuss the numerical behaviour of the method.
In the second example, we employ the scheme as a high-fidelity reference solver for the construction of reduced order models.

2.3.1 Riemann test case

In the context of nonlinear hyperbolic conservation laws, the inviscid Burgers equation (2.1) represents a fundamental testbed for assessing the ability of numerical schemes to capture discontinuities, rarefaction waves and entropy satisfying shock structures. To

this end, we consider the classical Riemann-type test case described in [11], consisting of a step initial condition over the domain $\Omega = [0,1.5]$:

$$u(x,0) = \begin{cases} -\frac{1}{2}, & \text{if } x \leq \frac{1}{2}, \\ 1, & \text{if } \frac{1}{2} < x \leq 1, \\ 0, & \text{if } x > 1. \end{cases}$$

This configuration generates a composite wave pattern: a rarefaction wave forms between $x = 0.5$ and $x = 1$ and a shock wave develops on the right, beyond $x = 1$. The rarefaction region evolves according to the method of characteristics, where the trajectories follow:

$$\frac{dx}{dt} = u(x,t),$$

resulting in the continuous solution:

$$u(x,t) = \frac{x - 0.5}{t}, \quad x \in [0.5 - 0.5t,\, 0.5 + 0.5t]\,.$$

On the right, a shock forms whose propagation speed is given by the Rankine-Hugoniot condition:

$$s = \frac{f(u_R) - f(u_L)}{u_R - u_L}.$$

From a numerical perspective, we consider the Lax-Friedrichs method. The simulation is conducted on $M = 75$ equispaced cells with $\Delta x = 0.02$, CFL number 0.8 and final time $T = 0.5$.

A particularly interesting phenomenon emerges near the point $x = 0.5$, known as the sonic point, where the fluid velocity coincides with the characteristic speed:

$$\frac{dx}{dt} = u(x,t).$$

Here, characteristics from different initial states intersect, causing non-uniqueness in the weak solution and requiring entropy based selection. However, both methods exhibit a localised anomaly at this point, known in the literature as the sonic point glitch or entropy glitch.

In this region, the exact solution involves a delicate balance between the head of the rarefaction and the constant state to its left, making the solution particularly sensitive to the numerical viscosity [3]. Although Lax-Friedrichs satisfies the entropy condition through its built-in dissipative term, the scheme may locally fail to reproduce the smooth rarefaction profile, generating a small but sharp discontinuity or plateau. This phenomenon is interpreted as a numerical artifact arising from the inability of the method to correctly resolve the characteristic structure near the sonic point [8].

For the Lax-Friedrichs method, the entropy condition is enforced via artificial viscosity. The amount of numerical viscosity introduced by the scheme is proportional to $\frac{\Delta x}{\Delta t}$ and

therefore tends to smear shocks while also degrading the accuracy in smooth regions. Near the sonic point, this artificial viscosity acts unevenly, leading to an over-diffused representation of the rarefaction and producing the visible entropy glitch [8]. A visible glitch still appears in the solution, which is attributed to the excessive diffusivity of the method.

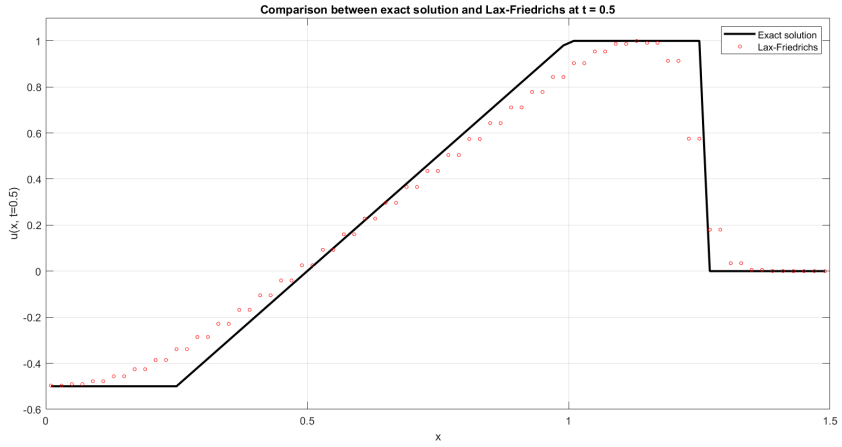


Figure 2.1: Lax-Friedrichs method at $t = 0.5$ compared with the exact solution.

Overall, while Lax-Friedrichs is more diffusive than other schemes, it remains robust and stable, which is why it is adopted for the simulations presented in the following sections [11].

2.3.2 High-fidelity simulations for snapshots generation

We now turn to the use of the Lax-Friedrichs method to generate high-fidelity solutions of the scalar Burgers equation (2.1), which serve as the bases for constructing a reduced order model. The main goal is to evaluate how effectively the reduced model reproduces the dynamics of the full-order system while significantly reducing the computational cost.

To construct the reference solution the Lax-Friedrichs scheme is used and the compact support function (1.9) is considered as the initial condition.

Entropy evolution

Figure 2.2 shows the temporal evolution of the entropy integral

$$S(t) = \int_{\Omega} \eta(u(x,t))\, dx, \qquad \eta(u) = \tfrac{1}{2}u^2,$$

computed numerically through the discrete entropy functional:

$$S_h(t^n) = \sum_k \tfrac{1}{2}(u_k^n)^2 \, \Delta x.$$

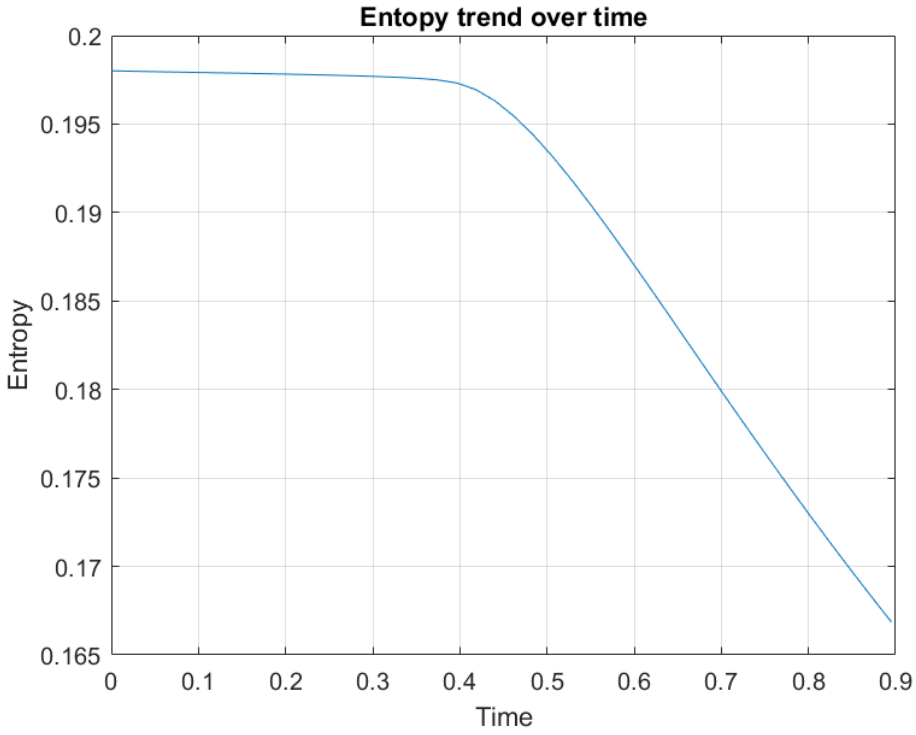


Figure 2.2: Temporal evolution of the entropy integral for the Lax-Friedrichs scheme applied to the inviscid Burgers equation.

This quantity provides a global measure of the entropy of the system and allows us to assess numerically whether the discrete solution satisfies the entropy inequality. For the inviscid Burgers equation, the entropy inequality reads:

$$\frac{d}{dt} \int_\Omega \eta(u) \, dx \leq 0,$$

meaning that the entropy must be non-increasing in time, with a strictly decreasing trend after shock formation.

The plot in Figure 2.2 confirms the expected behaviour of the Lax-Friedrichs scheme. Before the development of the shock, the entropy already shows a slight decay. This is fully consistent with the nature of Lax-Friedrichs, because due to the presence of the artificial viscosity term the scheme is already dissipative from an entropy point of view, even when the exact solution is regular.

This implies that the scheme produces a numerical dissipation of entropy greater than the physical one, which is a direct manifestation of its high numerical diffusivity.

Once the shock forms, the entropy starts to decrease more rapidly. This transition marks the moment in which the dissipation of the physical entropy, associated with the entropy satisfying shock, dominates the temporal behaviour of the discrete functional $S_h(t)$. The monotonic decrease observed after this point is therefore fully consistent with the entropy inequality of the continuous problem.

Another very important observation to mention is that the curve in the figure never increases at any point. This is clear numerical confirmation of the theoretical result established earlier, which is that the Lax-Friedrichs scheme for Burgers is entropy stable, meaning that its discrete entropy production satisfies $S_h(t^{n+1}) \leq S_h(t^n)$ for all n.

Therefore, the evolution of entropy in Figure 2.2 not only illustrates the dissipative behaviour of Lax-Friedrichs, but also serves as numerical validation of its entropy stability property. This justifies the use of the scheme as a reliable reference solver in the context of reduced order models, since despite being diffusive, it guarantees physically admissible solutions at all times.

SVD decomposition and POD bases construction

Figure 2.3 shows the trend of the squared singular values σ^2 obtained from the SVD decomposition of the snapshot matrix built from $N_s = 40$ temporal snapshots of the Lax-Friedrichs solution.

The singular values decrease rapidly, reaching the order of about 10^{-6}. This indicates that the Lax-Friedrichs solutions are relatively smooth, with fewer sharp transitions, making it possible to capture most of the dynamics with fewer dominant modes. This property enhances the efficiency of the POD technique for compression. In this work, 40

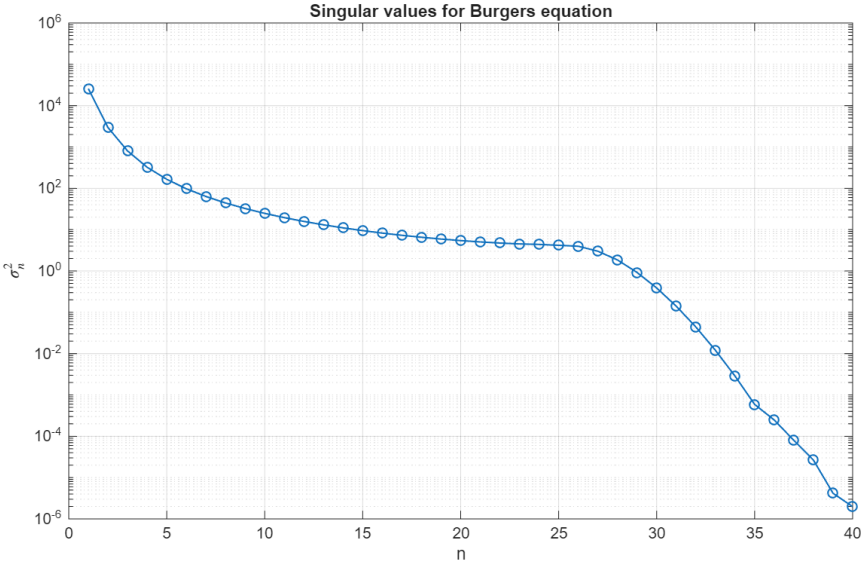
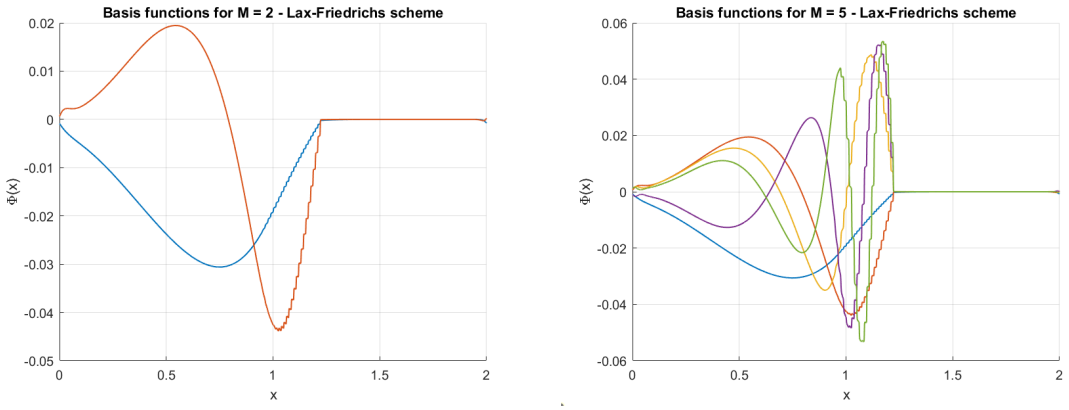


Figure 2.3: Squared singular values of the snapshot matrix for the Lax-Friedrichs scheme.

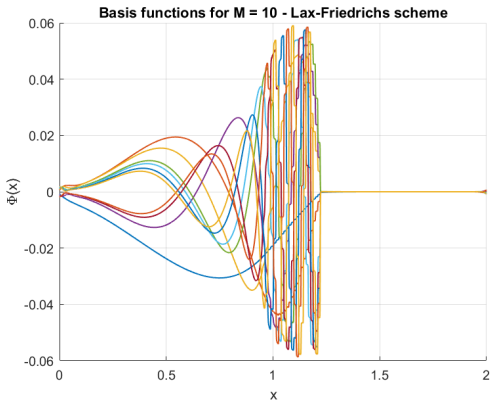
POD basis functions, equal to the number of available snapshots, are considered for the construction of the reduced model.

Figure 2.4 shows examples of POD basis functions obtained from the Lax-Friedrichs snapshots for different values of M ($M = 2$, $M = 5$, $M = 10$ and $M = 21$). Increasing M allows progressively finer details of the solution to be represented. The value $M = 21$ was chosen as optimal according to the RIC criterion, which ensures the capture of 99.9% of the total energy.

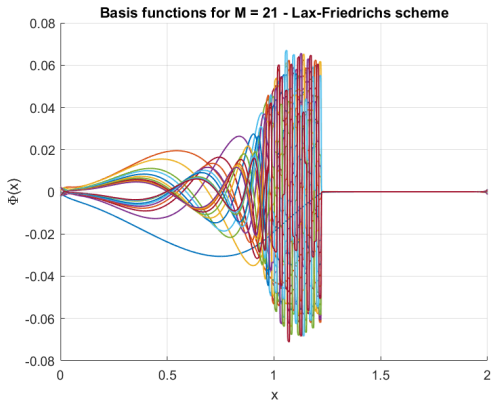


(a) $M = 2$

(b) $M = 5$



(c) $M = 10$



(d) $M = 21$

Figure 2.4: POD basis functions computed for different values of M for Lax-Friedrichs scheme.

Chapter 3

The reduced order strategy

In this chapter, after verifying the entropy stability of the high-fidelity discretisation, we now turn our attention to the reduced order models. The central question is whether the reduced approximation is still able to preserve the entropy as the high fidelity scheme.

3.1 Entropy stability of the high-fidelity scheme

The goal of this section is to prove that if the numerical flux used in the semi-discrete finite volume scheme is entropy stable, then the fully discrete method obtained after time discretisation also satisfies a discrete entropy inequality [9, 10].
We work with a first-order finite volume scheme, specifically the Lax-Friedrichs method given in (2.2) and we assume that the entropy pair (η, g) is convex.

In the finite volume setting, the approximate solution is a piecewise constant function:

$$u_h(x,t) = \sum_{j=1}^{N} u_j(t)\, \chi_{I_j}(x),$$

where $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, and $u_j(t)$ denotes the cell average of the solution over I_j. This piecewise constant representation is consistent with the formulation of first order finite volume schemes such as the Lax-Friedrichs method in (2.2), with:

$$\chi_{I_j}(x) = \begin{cases} 1 & \text{if } x \in I_j, \\ 0 & \text{otherwise.} \end{cases}$$

Theorem: Let $u_h(x,t)$ be an entropy-stable solution of the governing equations. Then, for every $\tau \in [t, t+\Delta t]$, the solution remains entropy stable:

$$u_h(x,\tau) \text{ is entropy stable for every } \tau \in [t, t+\Delta t].$$

In the continuous setting, a scheme is said to be entropy stable if it satisfies the discrete entropy inequality:

$$\frac{\partial}{\partial t}\eta(u_h(x,t)) + \frac{\partial}{\partial x}g(u_h(x,t)) \leq 0. \tag{3.1}$$

This inequality must hold pointwise for all x and t, ensuring that the high-fidelity solution is entropy stable in the continuous sense.

We consider the rectangular space-time cell $e_j = I_j \times [t^n, t^{n+1}]$ and compute the cell average of the time derivative of the entropy:

$$⨍_{e_j} \partial_t \eta(u_h)\, dt\, dx = ⨍_{I_j} \left(⨍_{t^n}^{t^{n+1}} \partial_t \eta(u_h(x,t))\, dt \right) dx.$$

Applying the fundamental theorem of calculus,

$$⨍_{e_j} \partial_t \eta(u_h)\, dt\, dx = ⨍_{I_j} \Big[\eta(u_h(x,t^{n+1})) - \eta(u_h(x,t^n))\Big]\, dx = \overline{\eta(u_h)}_j^{n+1} - \overline{\eta(u_h)}_j^{n},$$

where the cell average of the entropy is defined as:

$$\overline{\eta(u_h)}_j^{n} := \frac{1}{\Delta x} \int_{I_j} \eta(u_h(x,t^n))\, dx.$$

Since $u_h(x,t^n) = u_j^n$ is constant over the cell I_j, we have:

$$\overline{\eta(u_h)}_j^{n} = \eta(u_j^n).$$

At the next time level, we define the cell average of the conservative variable as

$$u_j^{n+1} := ⨍_{I_j} u_h(x,t^{n+1})\, dx.$$

Because η is convex, Jensen's inequality yields:

$$\eta(u_j^{n+1}) = \eta\left(⨍_{I_j} u_h(x,t^{n+1})\, dx \right) \;\leq\; ⨍_{I_j} \eta\Big(u_h(x,t^{n+1})\Big)\, dx = \overline{\eta(u_h)}_j^{n+1}.$$

Integrating the entropy flux contribution in (3.1) over e_j yields

$$\frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} \left[g(u_h(x_{j+\frac{1}{2}},t)) - g(u_h(x_{j-\frac{1}{2}},t)) \right] dt.$$

In the numerical scheme, the entropy flux g is replaced by a consistent numerical entropy flux $\tilde{g}$, which depends on the left and right traces at the interfaces [9].

Thus the space-time integral of (3.1) gives

$$\overline{\eta(u_h^{n+1})}_j \leq \eta(u_j^n) - \lambda \left(\tilde{g}^n_{j+\frac{1}{2}} - \tilde{g}^n_{j-\frac{1}{2}} \right), \qquad \lambda = \frac{\Delta t}{\Delta x}.$$

Combining this with Jensen's inequality, we obtain the fully discrete entropy inequality

$$\eta(u_j^{n+1}) \leq \eta(u_j^n) - \lambda \left(\tilde{g}^n_{j+\frac{1}{2}} - \tilde{g}^n_{j-\frac{1}{2}} \right). \tag{3.2}$$

Since $\tilde{g}$ is entropy stable and consistent with g, inequality (3.2) proves that the fully discrete finite volume scheme is entropy stable.

3.2 Reduced order model of the Burgers equation with proper orthogonal decomposition

Let us now consider the Burgers equation in the context of the reduced model with POD bases. The reduced solution at time t^n is denoted by $\mathbf{u}_c^n$ and the evolution is given by:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathcal{L}(\mathbf{u}_c^n),$$

where $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ is the reduced orthonormal bases matrix, $\mathcal{L}$ is the operator representing the numerical scheme and $M \ll N$. Consequently $\mathbf{u}_c^n, \mathbf{u}_c^{n+1} \in \mathbb{R}^N$ and $\mathbf{\Phi}\mathbf{\Phi}^\top \in \mathbb{R}^{N \times N}$.

In the case of Burgers, we have:

$$v = u, \qquad \eta(u) = \frac{1}{2}u^2.$$

Let us start from the assumption that $\mathbf{u}_c^n$ is entropy stable.

The numerical operator $\mathcal{L}$ can be expressed as:

$$\mathcal{L}(\mathbf{u}_c^n) = \mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n),$$

where $\mathbf{\Delta f} = (f_{i+1/2} - f_{i-1/2})_i$.

Therefore:

$$\begin{aligned} \mathbf{u}_c^{n+1} &= \mathbf{\Phi}\mathbf{\Phi}^\top \left(\mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n) \right) \\ &= \left(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I}\right) \mathbf{u}_c^n + \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathcal{F}(\mathbf{u}_c^n), \end{aligned}$$

where we have defined:

$$\mathcal{F}(\mathbf{u}_c^n) := -\frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n).$$

Let us call:

$$\hat{\mathbf{u}}_c^{n+1} := \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathcal{F}(\mathbf{u}_c^n);$$

so we can write the evolution as:

$$\mathbf{u}_c^{n+1} = \left(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I}\right) \mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}.$$

3.2.1 Step 1: Entropy stability of $\hat{\mathbf{u}}_c^{n+1}$

We first check whether $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable. To do that we follow the approach introduced by Chan in [1], where an entropy conservative reduced model is constructed starting from an entropy stable discretisation of the full order model. The strategy consists in projecting the entropy variables onto the reduced bases, so that the entropy conservation law is also preserved at the reduced level.

For completeness, the detailed proof of Chan's result is written in Appendix A.

In our setting, we define the reduced entropy variables as:

$$\mathbf{v}_c(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{v}(t).$$

Using this definition and the reduced operator, the time derivative of the conservative variables can be written as:

$$\frac{d}{dt}\mathbf{u}(\mathbf{v}_c(t)) = \mathbf{\Phi}\mathbf{\Phi}^\top \frac{d}{dt}\mathbf{u}(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathcal{F}(\mathbf{u}(\mathbf{v}(t))) \approx \mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

Writing the derivative of the reduced entropy as:

$$\frac{d}{dt}\eta_c(t) = \frac{d}{dt}\eta_c(\mathbf{u}(\mathbf{v}_c(t))) = \mathbf{v}_c^\top(t)\frac{d}{dt}\,\mathbf{u}(\mathbf{v}_c(t)),$$

we can rewrite:

$$\mathbf{v}_c^\top(t)\,\frac{d}{dt}\,\mathbf{u}(\mathbf{v}_c(t)) \simeq \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{\Phi}\mathbf{\Phi}^\top\mathcal{F}(\mathbf{u}(\mathbf{v}(t)))).$$

Now considering that $\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi} = I$, then:

$$\mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{\Phi}\mathbf{\Phi}^\top\mathcal{F}(\mathbf{u}(\mathbf{v}(t)))) = \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top\mathcal{F}(\mathbf{u}(\mathbf{v}(t)))) = \mathbf{v}_c^\top(t)\mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

We have:

$$\frac{d}{dt}\eta_c(t) = \mathbf{v}_c^\top(t)\mathcal{F}(\mathbf{u}(\mathbf{v}_c(t)));$$

so we have verified that $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable.

3.2.2 Step 2: Entropy stability of $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{u}_c^n$

Returning to the equation obtained previously:

$$\mathbf{u}_c^{n+1} = (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1},$$

where the term $\hat{\mathbf{u}}_c^{n+1}$ has been shown to be entropy stable, we now demonstrate that the term $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{u}_c^n$ is also entropy stable.

The term $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{u}_c^n$ represents an orthogonal projection onto the subspace generated by the columns of $\mathbf{\Phi}$. Since $\mathbf{\Phi}\mathbf{\Phi}^\top$ is an orthogonal projection matrix, it has the following properties:

- it is symmetric, i.e. $\mathbf{\Phi}\mathbf{\Phi}^\top = (\mathbf{\Phi}\mathbf{\Phi}^\top)^\top$;
- it is idempotent, i.e. $(\mathbf{\Phi}\mathbf{\Phi}^\top)^2 = \mathbf{\Phi}\mathbf{\Phi}^\top$;
- it is positive semidefinite, i.e. for every $\mathbf{x} \in \mathbb{R}^N$, we have $\mathbf{x}^\top \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{x} \geq 0$.

We now want to prove that the matrix $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$ is negative semidefinite, i.e. that:

$$\mathbf{x}^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{x} \leq 0 \quad \forall \mathbf{x} \in \mathbb{R}^N.$$

Suppose that $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ has orthonormal columns. This implies that:

$$\mathbf{\Phi}^\top \mathbf{\Phi} = \mathbf{I}_M,$$

where $\mathbf{I}_M$ is the identity matrix of dimension M.

Let us consider the following scalar product:

$$\mathbf{x}^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{x} = \mathbf{x}^\top \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{x} - \mathbf{x}^\top \mathbf{x}.$$

Let us define:

$$\mathbf{z} := \mathbf{\Phi}^\top \mathbf{x} \in \mathbb{R}^M.$$

Then:

$$\mathbf{x}^\top \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{x} = (\mathbf{\Phi}^\top \mathbf{x})^\top (\mathbf{\Phi}^\top \mathbf{x}) = \mathbf{z}^\top \mathbf{z} = \|\mathbf{z}\|_2^2,$$

and also:

$$\mathbf{x}^\top \mathbf{x} = \|\mathbf{x}\|_2^2.$$

Therefore:

$$\mathbf{x}^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{x} = \|\mathbf{z}\|_2^2 - \|\mathbf{x}\|_2^2.$$

Since $\mathbf{\Phi}^\top$ is a projection operator, it is a contraction with respect to the Euclidean norm. Therefore:

$$\|\mathbf{z}\|_2 = \|\mathbf{\Phi}^\top \mathbf{x}\|_2 \leq \|\mathbf{x}\|_2.$$

It immediately follows that:

$$\|\mathbf{z}\|_2^2 \leq \|\mathbf{x}\|_2^2 \quad \Rightarrow \quad \mathbf{x}^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{x} \leq 0.$$

The matrix $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$ is symmetric and negative semidefinite.

Consequently, the term $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\mathbf{u}_c^n$ does not contribute positively to entropy and the entire scheme remains entropy stable.

3.2.3 Step 3: Final estimate of the entropy

Let us consider of the solution reduced to time step $n+1$, i.e. $\eta(\mathbf{u}_c^{n+1})$.

Starting from the decomposition obtained above:

$$\mathbf{u}_c^{n+1} = (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1},$$

we study:

$$\eta(\mathbf{u}_c^{n+1}) = \eta\left((\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right).$$

In the case of the Burgers equation, the entropy function is:

$$\eta(\mathbf{u}) = \frac{1}{2}\mathbf{u}^\top\mathbf{u}.$$

Applying this definition, we obtain:

$$\begin{aligned}\eta(\mathbf{u}_c^{n+1}) &= \frac{1}{2}\left((\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right)^\top\left((\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right)\\ &= \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^T(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\mathbf{u}_c^n)^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n\\ &+ \frac{1}{2}(\mathbf{u}_c^n)^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^T(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n;\end{aligned}$$

where we used the fact that $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$ is idempotent, so $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})^2 = (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$.

Since $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$ is also symmetric, we can use the property:

$$A \in \mathrm{Sym} \Rightarrow \mathbf{v}^\top A\mathbf{w} = \mathbf{w}^\top A\mathbf{v},$$

and therefore we obtain:

$$\eta(\mathbf{u}_c^{n+1}) = \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\mathbf{u}_c^n)^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n.$$

Since $(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})$ is negative semidefinite, we have:

$$\frac{1}{2}(\mathbf{u}_c^n)^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n \leq 0,$$

so all that remains is to evaluate :

$$\eta(\mathbf{u}_c^{n+1}) = \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n. \tag{3.3}$$

From the definition:

$$\hat{\mathbf{u}}_c^{n+1} = \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top\mathcal{F}(\mathbf{u}_c^n),$$

substituting in (3.3) follows:

$$\begin{aligned}
&\leq \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}\left(\mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathcal{F}(\mathbf{u}_c^n)\right)^\top (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n \\
&= \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\mathbf{u}_c^n)^\top (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n + \frac{1}{2}\mathcal{F}(\mathbf{u}_c^n)^\top \mathbf{\Phi}\mathbf{\Phi}^\top (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{I})\,\mathbf{u}_c^n \\
&\leq \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}\mathcal{F}(\mathbf{u}_c^n)^\top (\mathbf{\Phi}\mathbf{\Phi}^\top - \mathbf{\Phi}\mathbf{\Phi}^\top)\mathbf{u}_c^n \\
&= \eta(\hat{\mathbf{u}}_c^{n+1}).
\end{aligned}$$

Hence, we obtain the final estimate showing that the entropy of the reduced solution at time step $n+1$ is:

$$\eta(\mathbf{u}_c^{n+1}) \leq \eta(\hat{\mathbf{u}}_c^{n+1}),$$

which ensures that the reduced model does not increase the entropy.

3.3 Reduced order model of the Burgers equation with weighted scalar product

Let us consider the Burgers equation in the context of a reduced model based on orthonormal POD bases with respect to a scalar product weighted by $\mathbf{W}$. The reduced solution at time t^n is denoted by $\mathbf{u}_c^n$ and the evolution is given by:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{L}(\mathbf{u}_c^n), \tag{3.4}$$

where $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ is the reduced bases matrix, $M \ll N$.

From now on we assume: $\mathbf{W} = \Delta x\mathbf{I}$, i.e. $\mathbf{W}$ is diagonal with constant entries on the diagonal. In this case the weighted scalar product coincides, up to the factor Δx, with the euclidean scalar product. For a general diagonal $\mathbf{W}$ with non uniform entries the first step of the following demonstration does not hold, see equation (3.8).

In the case of Burgers:

$$\mathbf{v} = \mathbf{u}, \qquad \eta(\mathbf{u}) = \frac{1}{2}\mathbf{u}^2.$$

Let us assume that $\mathbf{u}_c^n$ is entropy stable. The numerical operator $\mathcal{L}$ is defined as:

$$\mathcal{L}(\mathbf{u}_c^n) = \mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n),$$

where $\mathbf{\Delta f}(\mathbf{u}_c^n) = (f_{i+1/2} - f_{i-1/2})_i$.

Then:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\left(\mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n)\right) \tag{3.5}$$

$$= \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top W\mathcal{F}(\mathbf{u}_c^n), \quad \text{with } \mathcal{F}(\mathbf{u}_c^n) := -\frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n). \tag{3.6}$$

We define:

$$\hat{\mathbf{u}}_c^{n+1} := \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}_c^n).$$

from which:

$$\mathbf{u}_c^{n+1} := (\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}. \tag{3.7}$$

3.3.1 Step 1: Entropy stability of $\hat{\mathbf{u}}_c^{n+1}$

First, we check whether $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable:

$$\mathbf{v}_c(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{v}(t)$$

$$\frac{d}{dt}\mathbf{u}(\mathbf{v}_c(t)) = \frac{d}{dt}(\mathbf{u}(\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{u}(t))) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\frac{d}{dt}\mathbf{u}(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))) \approx \mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

Writing the derivative of the reduced entropy as:

$$\frac{d}{dt}\eta_c(t) = \frac{d}{dt}\eta_c(\mathbf{u}(\mathbf{v}_c(t))) = \mathbf{v}_c^T(t)\frac{d}{dt}\ \mathbf{u}(\mathbf{v}_c(t)),$$

We can rewrite:

$$\mathbf{v}_c^\top(t)\ \frac{d}{dt}\ \mathbf{u}(\mathbf{v}_c(t)) \simeq \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))). \tag{3.8}$$

Note that the transpose of $\mathbf{v}_c(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{v}(t)$ is $\mathbf{v}_c^\top(t) = \mathbf{v}^\top(t)W^\top \mathbf{\Phi}\mathbf{\Phi}^\top$. We can write $\mathbf{v}_c^\top(t) = \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}$ only when $\mathbf{W}$ is constant. Therefore equation (3.8) holds only if $\mathbf{W}$ is diagonal with constant entries.

Now, considering that $\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi} = \mathbf{I}$, then:

$$\mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))) = \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))) = \mathbf{v}_c^T(t)\mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

So we have:

$$\frac{d}{dt}\eta_c(t) = \mathbf{v}_c^T(t)\mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

In this way we verified that, considering W diagonal with constant entries, $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable.

3.3.2 Step 2: Entropy stability of $(\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I})\mathbf{u}_c^n$

We now demonstrate that the other term of equation (3.7), so $(\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I})\mathbf{u}_c^n$ is entropy stable.

To prove that the matrix $\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I}$ is negative semidefinite we first introduce a useful theorem.

Orthogonal subspace decomposition theorem

Let $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ with $\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi} = \mathbf{I}_M$ and let $\mathbf{E} = \mathrm{Im}(\mathbf{\Phi}) \subseteq \mathbb{R}^N$. Then:

$$\mathbb{R}^N = \mathbf{E} \oplus \mathbf{E}^{\perp \mathbf{w}}$$

where $\mathbf{E}^{\perp \mathbf{w}}$ is the orthogonal of $\mathbf{E}$ with respect to the weighted scalar product $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{W}} = \mathbf{x}^\top \mathbf{W}\mathbf{y}$ and:

$$\dim(\mathbf{E}) = M, \quad \dim(\mathbf{E}^{\perp \mathbf{w}}) = N - M$$

Proof

Since $\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi} = \mathbf{I}$, the columns of $\mathbf{\Phi}$ are orthogonal to each other with respect to $\mathbf{W}$ and form a $\mathbf{W}$-weighted orthonormal bases of $\mathbf{E}$.

We complete this bases to a complete orthonormal bases of $\mathbb{R}^N$ with respect to $\mathbf{W}$, introducing vectors $\mathbf{z}_1, \ldots, \mathbf{z}_{N-M}$ such that:

- $\langle \boldsymbol{\phi}_i, \mathbf{z}_j \rangle_W = 0$ for all i, j
- $\langle \mathbf{z}_i, \mathbf{z}_j \rangle_W = \delta_{ij}$

Given any $\mathbf{v} \in \mathbb{R}^N$, we can write it as:

$$\mathbf{v} = \mathbf{w} + \mathbf{w}^\perp, \quad \text{with } \mathbf{w} \in \mathbf{E},\ \mathbf{w}^\perp \in \mathbf{E}^{\perp \mathbf{w}}$$

This decomposition is unique, therefore:

$$\mathbb{R}^N = \mathbf{E} \oplus \mathbf{E}^{\perp \mathbf{w}}$$

■

Given $\mathbf{v} \in \mathbb{R}^N$, its weighted orthogonal projection onto $E = \mathrm{Im}(\mathbf{\Phi})$ is:

$$\mathbb{P}_E(\mathbf{v}) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{v}$$

and $\mathbb{P}_E = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} \in \mathbb{R}^{N\times N}$ is its linear application.

Since the columns of $\mathbf{\Phi}$ form an orthonormal bases with respect to $\mathbf{W}$, then:

$$\mathbb{P}_E(\mathbf{v}) = \sum_{i=1}^{M} \langle \mathbf{v}, \boldsymbol{\phi}_i \rangle_{\mathbf{W}} \cdot \boldsymbol{\phi}_i = \sum_{i=1}^{M} (\boldsymbol{\phi}_i^\top \mathbf{W}\mathbf{v}) \boldsymbol{\phi}_i = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{v}$$

The kernel and image are defined as:

$$\ker(\mathbb{P}_{\mathbf{E}}) = \mathbf{E}^{\perp \mathbf{w}}, \quad \mathrm{Im}(\mathbb{P}_{\mathbf{E}}) = \mathbf{E}.$$

We have $\mathbb{P}_{\mathbf{E}}(\mathbf{v}) = 0$ if and only if $\langle \mathbf{v}, \phi_i \rangle_{\mathbf{W}} = 0\ \forall i$, which implies that $\mathbf{v} \in \mathbf{E}^{\perp \mathbf{w}}$. Similarly, every vector $\mathbf{w} \in \mathbf{E}$ is the image of itself: $\mathbb{P}_{\mathbf{E}}(\mathbf{w}) = \mathbf{w}$.

The matrix associated with the weighted orthogonal projection is:

$$\mathbb{P}_{\mathbf{E}} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}.$$

This is symmetric with respect to $\mathbf{W}$, i.e.:

$$(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W})^\top = \mathbf{W}\mathbf{\Phi}\mathbf{\Phi}^\top.$$

Since $\mathbb{P}_\mathbf{E} = \mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W}$ is an idempotent projection, the only possible eigenvalues are 0 and 1. In fact:

- $\mathbf{v} \in \mathbf{E} \Leftrightarrow \mathbb{P}_\mathbf{E}(\mathbf{v}) = \mathbf{v} \Rightarrow \mathbf{v}$ eigenvector with eigenvalue 1
- $\mathbf{v} \in \mathbf{E}^{\perp_\mathbf{W}} \Leftrightarrow \mathbb{P}_\mathbf{E}(\mathbf{v}) = \mathbf{0}$ eigenvalue 0

where the eigenvalue $\lambda = 1$ has geometric multiplicity M, while the eigenvalue $\lambda = 0$ has geometric multiplicity $N - M$.
There are no other eigenvalues, otherwise the sum of the geometric multiplicities would be greater than N, which is impossible. Since P_E is diagonalizable because it is symmetric, the algebraic multiplicity of each eigenvalue must equal the geometric multiplicity.

Knowing the eigenvalues of $\mathbf{\Phi}\mathbf{\Phi}^\top$, we can deduce the eigenvalues of the matrix $\mathbf{A} = \mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I}_N$, which are:

$$\lambda_i(\mathbf{A}) = \lambda_i(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W}) - 1 \in \{-1, 0\}$$

- -1 with multiplicity $N - M$ defined on $\mathrm{Im}(\mathbf{\Phi})^{\perp_\mathbf{W}}$
- 0 with multiplicity M defined on $\mathrm{Im}(\mathbf{\Phi})$

In fact for the spectral theorem if A is a symmetric matrix, then the quadratic form $\mathbf{x}^\top\mathbf{A}\mathbf{x}$ is negative semidefinite if and only if every eigenvalue of A is non positive and at least one is zero.

Therefore, we conclude that the matrix: $\mathbf{A} = \mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I}_N$ is negative semidefinite.

Consequently, the term $(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n$ does not contribute positively to entropy and the scheme remains entropy stable.

3.3.3 Step 3: Final estimate of the entropy

Let us consider the entropy of the solution at time step $n + 1$, i.e., $\eta(\mathbf{u}_c^{n+1})$.

Starting from the decomposition:

$$\mathbf{u}_c^{n+1} = (\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1},$$

we study:

$$\eta(\mathbf{u}_c^{n+1}) = \eta\left((\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right).$$

In the case of the Burgers equation, the entropy function is:

$$\eta(\mathbf{u}) = \frac{1}{2}\mathbf{u}^\top \mathbf{u}.$$

Applying this definition, we obtain:

$$\begin{aligned}\eta(\mathbf{u}_c^{n+1}) &= \frac{1}{2}\left((\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right)^\top\left((\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}\right)\\ &= \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\mathbf{u}_c^n)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n\\ &\quad + \frac{1}{2}(\mathbf{u}_c^n)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\hat{\mathbf{u}}_c^{n+1} + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n.\end{aligned}$$

where we have used the fact that $(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})$ is idempotent, so

$$(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})^2 = (\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I}).$$

Since $(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})$ is also symmetric, we can use the property:

$$\mathbf{A}\in \text{Sym} \Rightarrow \mathbf{v}^\top\mathbf{A}\mathbf{w} = \mathbf{w}^\top\mathbf{A}\mathbf{v},$$

and therefore we obtain:

$$\eta(\mathbf{u}_c^{n+1}) = \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\mathbf{u}_c^n)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n.$$

Since $(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})$ is negative semidefinite, we have:

$$\frac{1}{2}(\mathbf{u}_c^n)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n \leq 0,$$

and finally:

$$\eta(\mathbf{u}_c^{n+1}) = \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}(\hat{\mathbf{u}}_c^{n+1})^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n.$$

From the definition:

$$\hat{\mathbf{u}}_c^{n+1} = \mathbf{u}_c^n + \boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}\mathcal{F}(\mathbf{u}_c^n),$$

substituting in (3.3) follows:

$$\begin{aligned}\eta(\mathbf{u}_c^{n+1}) &\leq \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}\left(\mathbf{u}_c^n + \boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}\mathcal{F}(\mathbf{u}_c^n)\right)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\mathbf{I})\mathbf{u}_c^n\\ &= \eta(\hat{\mathbf{u}}_c^{n+1}) + \frac{1}{2}\mathcal{F}(\mathbf{u}_c^n)^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W}-\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{W})\mathbf{u}_c^n\\ &= \eta(\hat{\mathbf{u}}_c^{n+1}).\end{aligned}$$

So we can conclude saying that the whole scheme is entropy stable under the assumption that $\mathbf{W}$ is diagonal with constant entries.

3.4 Numerical results

In this section, we assess the performance of the reduced order models by comparing them against the high-fidelity Lax-Friedrichs discretisation introduced in the previous chapter. The full-order solution of the Burgers equation (2.1), obtained with the same numerical setup and the compact exponential initial condition (1.9), is used as a reference to evaluate the ability of the reduced models to reproduce the dynamics of the system at a significantly lower computational cost.

3.4.1 Relative error in reduced order models

This subsection analyzes the accuracy of the reduced model obtained for $M = 21$, which represents the optimal number of modes capturing 99.9% of the energy of the system. Table 3.1 presents the relative errors computed during the online simulations, comparing each reduced model with respect to the high-fidelity solution.

Four projection strategies are compared:

- **Case 1**: ROM using POD bases with the identity matrix as weighting;
- **Case 2**: c-ROM with empirical weights from case 1 selected using NNLS algorithm;
- **Case 3**: ROM using POD bases with weight matrix $\mathbf{W} = \Delta x \mathbf{I}$;
- **Case 4**: c-ROM with empirical weights from case 3 selected using NNLS algorithm.

Table 3.1: Relative errors and number of collocation points for Lax-Friedrichs reduced models with $M = 21$ and NNLS tolerance 10^{-8}.

Case	Error	# Coll.	NNLS tol.
1	$3.9192301855532751 \times 10^{-3}$	4000	–
2	$6.1857182141209915 \times 10^{-3}$	163	10^{-8}
3	$3.9192301854904027 \times 10^{-3}$	4000	–
4	$6.1857182141114054 \times 10^{-3}$	163	10^{-8}

The results show that the relative error remains of the order of 10^{-3} in all cases. Even though the hyper-reduction approach in cases 2 and 4 uses significantly fewer collocation points, the resulting loss of accuracy is negligible. This confirms that the cMOR achieves substantial computational savings without compromising the solution quality.

3.4.2 Entropy stability in reduced order models

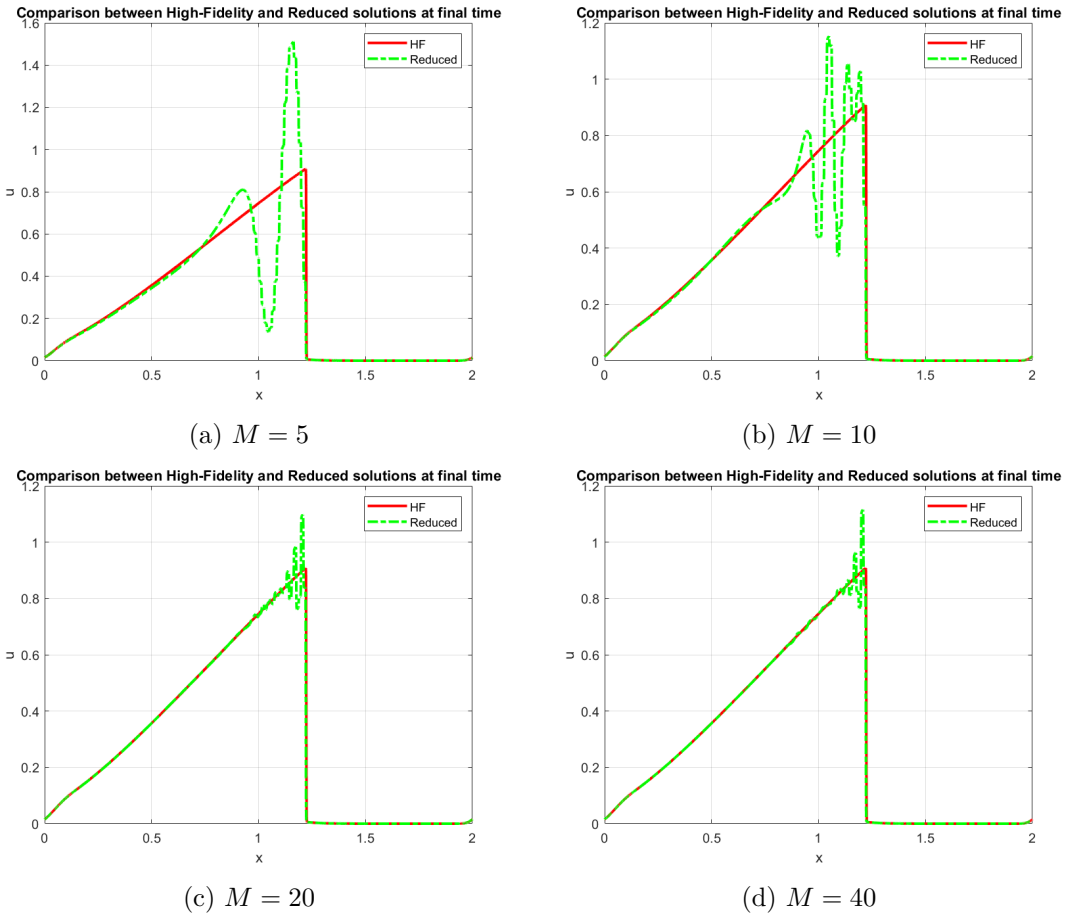
In this section we present a numerical validation of the theoretical properties of entropy conservation demonstrated in the previous sections.
The two cases studied are:

- **Case 1:** identity matrix $\mathbf{W} = \mathbf{I}$;
- **Case 2:** diagonal matrix proportional to the spatial step, $\mathbf{W} = \Delta x \cdot \mathbf{I}$.

In the first and second cases, as shown analytically in the previous sections, the reduced system maintains the entropy conservation properties of the complete model.

To validate these theoretical observations, we report the numerical results relating to the evolution of total entropy over time, calculated for both the high fidelity solution and the reduced solution. In each of the two cases, the analysis is conducted for different sizes of the reduced model, corresponding to $M = 5, 10, 20, 40$.

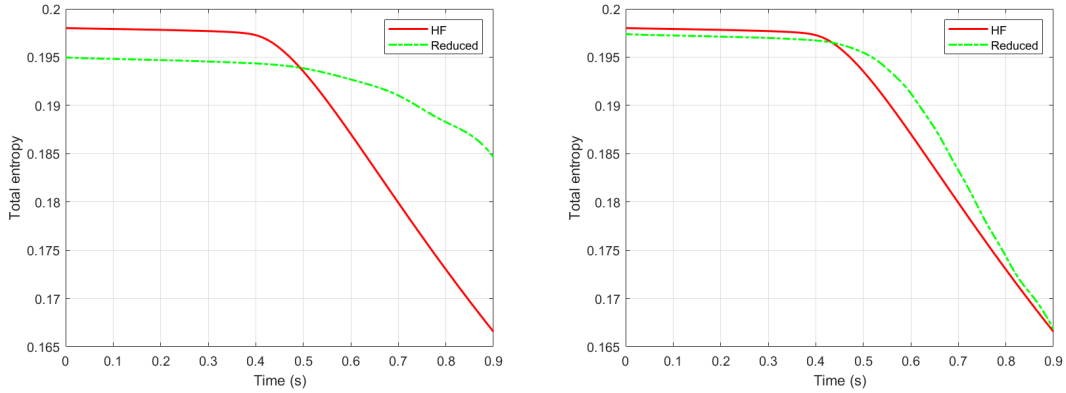
Figures 3.1 and 3.3 show the comparison between the entropy of the high fidelity solution and that of the reduced solution in the two cases described above. For $\mathbf{W} = \mathbf{I}$ in Figure 3.1 and for $\mathbf{W} = \Delta x \cdot \mathbf{I}$ with $\Delta x = 0.5 \cdot 10^{-3}$ in Figure 3.3, there is an almost perfect overlap of the entropy curves, indicating that the reduced model effectively preserves the total entropy, in accordance with the theoretical demonstrations.



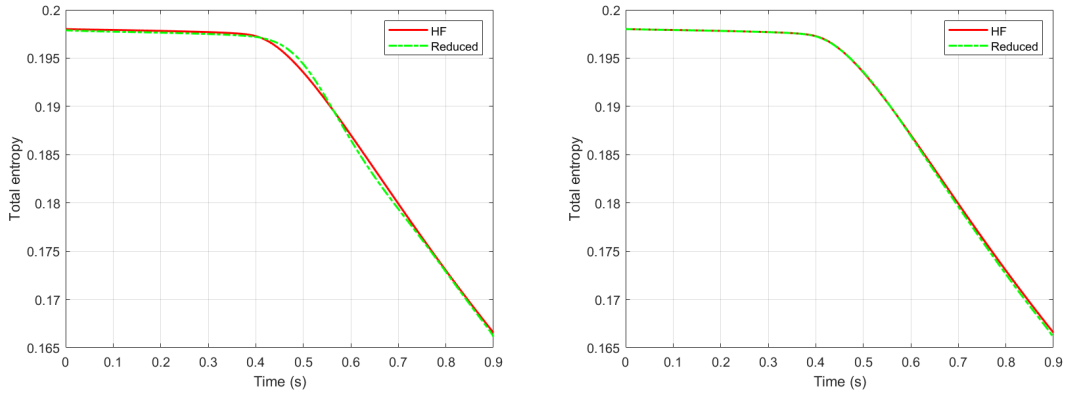
(a) $M = 5$ (b) $M = 10$

(c) $M = 20$ (d) $M = 40$

Figure 3.4: Comparison between high fidelity and reduced solutions with $\mathbf{W} = \Delta x \cdot \mathbf{I}$.



(a) $M = 5$ (b) $M = 10$



(c) $M = 20$ (d) $M = 40$

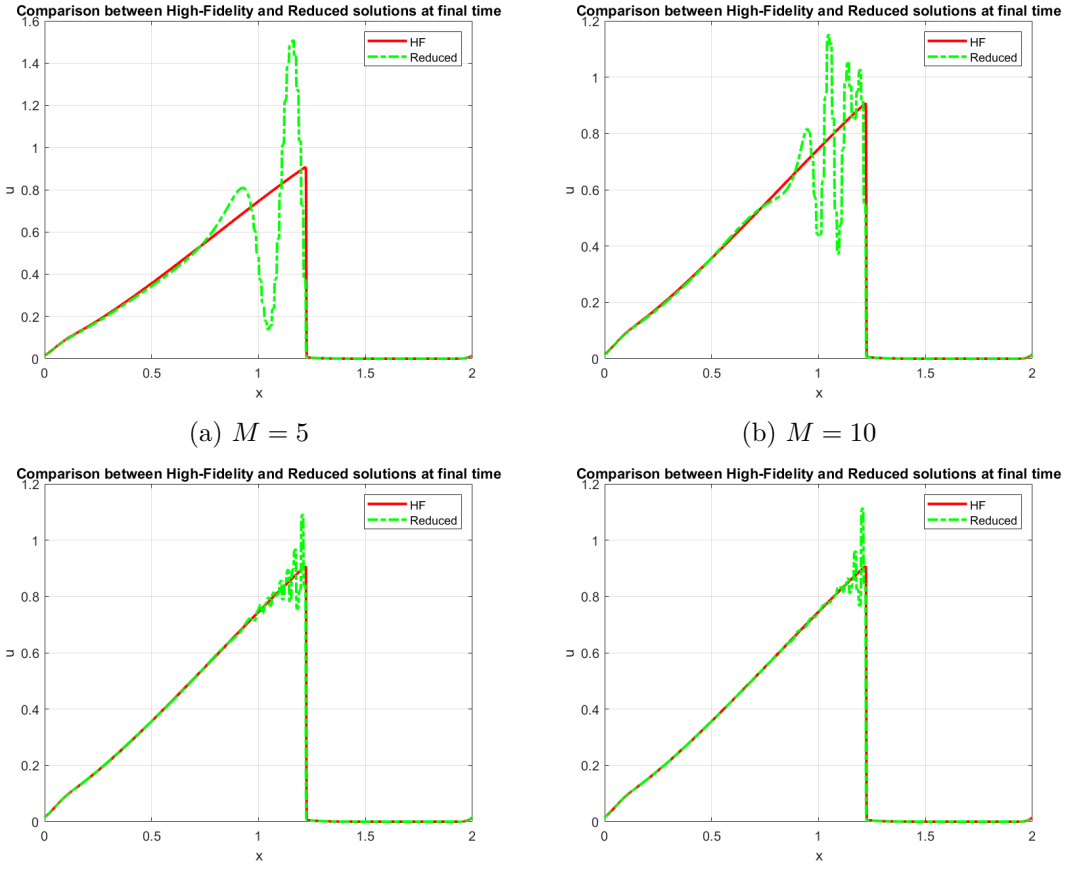
Figure 3.1: Comparison between the total entropy of the high fidelity solution and the reduced solution with $\mathbf{W} = \mathbf{I}$ for different values of M.

In order to evaluate the quality of the approximation provided by the reduced methods, we computed the relative error, defined in 1.30, between the high fidelity solution and the approximated solution obtained for different dimensions of the reduced bases M.

The results obtained for various choices of M are reported in Table 3.2.

Table 3.2: Relative error between the high-fidelity solution and the reduced one, for different numbers of modes M, using in the first column $\mathbf{W} = \mathbf{I}$, in the second column $\mathbf{W} = \Delta x \cdot \mathbf{I}$ with $\Delta x = 0.5 \times 10^{-3}$ and 10^{-12} as tolerance for the NNLS algorithm.

M	**Relative error** $(W = I)$	**# Coll.**	**Relative error** $(W = \Delta x I)$	**# Coll.**
5	$5.5580475064447238 \times 10^{-1}$	175	$5.5580475064411551 \times 10^{-1}$	175
10	$4.2418338161957631 \times 10^{-1}$	195	$4.2418338161911698 \times 10^{-1}$	195
20	$6.1495876546867982 \times 10^{-2}$	216	$6.1495876544178914 \times 10^{-3}$	216
40	$5.4072289986985722 \times 10^{-2}$	254	$5.4072289986306556 \times 10^{-2}$	254

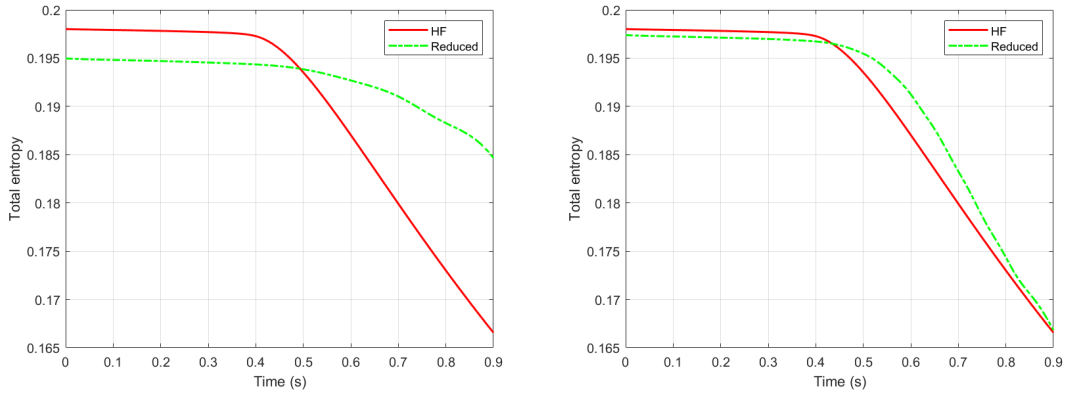


(a) $M = 5$ (b) $M = 10$

(c) $M = 20$ (d) $M = 40$

Figure 3.2: Comparison between high fidelity and reduced solutions with $\mathbf{W} = \Delta x\mathbf{I}$.

The results shown in Table 3.2 show that the relative error computed with $\mathbf{W} = \mathbf{I}$ and with $\mathbf{W} = \Delta x\mathbf{I}$ assumes almost identical values, but the error of $\mathbf{W} = \Delta x\mathbf{I}$ is slightly lower. This behaviour was expected, as the use of $\mathbf{W} = \Delta x\mathbf{I}$ is more accurate, since it takes into account the spatial step Δx in the discretisation of the integral. Consequently, the error estimate with $\mathbf{W} = \Delta x\mathbf{I}$ is slightly lower than that given by $\mathbf{W} = \mathbf{I}$.



(a) $M = 5$ (b) $M = 10$

(c) $M = 20$ (d) $M = 40$

Figure 3.3: Comparison between the total entropy of the high fidelity solution and the reduced solution with $\mathbf{W} = \Delta x \cdot \mathbf{I}$ for different values of M.

Chapter 4

Reduced order model through a collocation strategy based on Nonnegative Least Squares

In the previous chapters, we introduced the inviscid Burgers equation and discussed the role of entropy stability in reduced order models.
We now turn to the construction of a hyper-reduced model, in which the equations are imposed only at a restricted set of collocates points, i.e. selected cells of the domain that effectively represent the global behaviour of the system. Among the possible approaches for choosing collocation points, we adopt a strategy based on Nonnegative Least Squares, which provides positive weights and a consistent selection of nodes representative of the domain.
Below we present the mathematical formulation of the reduced model based on collocation and subsequently we analyse its entropy stability.

4.1 Hyper-reduced model with NNLS weights

The reduced model is defined by the relation:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}\, \bar{\mathbf{\Phi}}^\top \mathbf{W}_\epsilon\, \bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n),$$

where the matrices and vectors involved have the following dimensions: $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$, $\bar{\mathbf{\Phi}} \in \mathbb{R}^{N_c\times M}$, $\mathbf{W}_\epsilon \in \mathbb{R}^{N_c\times N_c}$, $\bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n) \in \mathbb{R}^{N_c\times 1}$, $\tilde{\mathbf{u}}_c^n \in \mathbb{R}^{N_{cn}\times 1}$. Here, N_c are the placed cells, while N_{cn} are the stencil cells, composed of collocation points and their neighboring cells to the left and right.

This model can be rewritten in a less compact form using the sparse weight matrix $\mathbf{W}$ of size $N \times N$. This reformulation is particularly useful in our context because it allows us

to express the model in the same form as (3.4), namely:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{L}(\mathbf{u}_c^n),$$

with the difference that, in this version, the matrix $\mathbf{W}$ is obtained through an NNLS decomposition.

Thanks to this formulation, we can define as in the case of (3.6):

$$\mathbf{u}_c^{n+1} := (\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I})\mathbf{u}_c^n + \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}_c^n). \tag{4.1}$$

with $\hat{\mathbf{u}}_c^{n+1} = \mathbf{u}_c^n + \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}_c^n)$.

4.1.1 Step 1: Entropy stability of $\hat{\mathbf{u}}_c^{n+1}$

First, we check whether $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable:

$$\mathbf{v}_c(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{v}(t).$$

Then:

$$\frac{d}{dt}\mathbf{u}(\mathbf{v}_c(t)) = \frac{d}{dt}(\mathbf{u}(\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathbf{u}(t))) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\frac{d}{dt}\mathbf{u}(t) = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))) \approx \mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))).$$

Writing the derivative of the reduced entropy as:

$$\frac{d}{dt}\eta_c(t) = \mathbf{v}_c^\top(t)\frac{d}{dt}\,\mathbf{u}(\mathbf{v}_c(t)),$$

we would like to project it in the form used for the standard weighted scalar product with constant $\mathbf{W}$. However, since here $\mathbf{W}$ is obtained through NNLS, it is sparse and its diagonal entries are not constant. Therefore, the identity:

$$\mathbf{v}_c^\top(t) = \mathbf{v}^\top(t)\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}$$

does not hold and the reduction step used in equation (3.8) is no longer valid. As a consequence we cannot conclude that $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable.

The derivation in (3.8) is valid if $\mathbf{W}$ is diagonal with constant entries. In this case the operator $\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}$ acts as an orthogonal projector with respect to it, thus ensuring entropy stability of $\hat{\mathbf{u}}_c^{n+1}$.

However, in the case of NNLS the matrix $\mathbf{W}$ obtained is sparse and its diagonal entries are not constant. Therefore the identity used above does not hold: $\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W}$ is not an orthogonal projector and entropy conservation is not guaranteed.

4.1.2 Step 2: The term $(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\,\mathbf{u}_c^n$

Returning to equation (4.1), where we showed that the term $\hat{\mathbf{u}}_c^{n+1}$ is not entropy stable in the general case, we now show that the term $(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n$ also does not preserve entropy.

Let us begin by observing that the projector $\mathbf{P} = \mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W}$ is not orthogonal with respect to the inner product weighted by $\mathbf{W}$. Therefore $\mathbf{P}$ does not enjoy the classical properties of orthogonal projectors:

- $\mathbf{P}$ is not symmetric with respect to $\mathbf{W}$;
- $\mathbf{P}^2 \neq \mathbf{P}$ in general;
- The eigenvalues of $\mathbf{P}$ may be outside the interval [0,1];
- The operator $\mathbf{P}$ is not a contraction in the norm induced by $\mathbf{W}$; in particular, there exist vectors whose norm is increased:

$$\|\mathbf{P}\|_{\mathbf{W}} := \sup_{\mathbf{x}\neq 0} \frac{\|\mathbf{P}\mathbf{x}\|_{\mathbf{W}}}{\|\mathbf{x}\|_{\mathbf{W}}} > 1.$$

The operator $\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I}$ is not semidefinite negative with respect to the inner product weighted by $\mathbf{W}$:

$$\exists \mathbf{x} \in \mathbb{R}^N \quad \text{such that} \quad \mathbf{x}^\top\mathbf{W}(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{x} > 0,$$

which implies that the term $(\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n$ can produce an increase in entropy in the reduced dynamics, violating the entropy stability of the system.

4.2 Hyper-reduced model with weighted projection

In the previous section we introduced a collocation-based hyper-reduced model with NNLS weights and showed that the resulting formulation is not entropy stable. To overcome this limitation, we consider here a modified reduced formulation based on POD bases together with the weighted inner product induced by $\mathbf{W}$. The idea is to project the nonlinear operator $\mathcal{L}$ onto this weighted space, leading to the following reduced dynamics:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W}\mathcal{L}(\mathbf{u}_c^n), \tag{4.2}$$

where $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ is the reduced bases matrix, $M \ll N$.

In the case of Burgers:

$$v = u, \qquad \eta(u) = \frac{1}{2}u^2.$$

Let us assume that $\mathbf{u}_c^n$ is entropy stable. The numerical operator $\mathcal{L}$ is defined as:

$$\mathcal{L}(\mathbf{u}_c^n) = \mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n),$$

where $\mathbf{\Delta f}(\mathbf{u}_c^n) = (f_{i+1/2} - f_{i-1/2})_i$.

Then:

$$\begin{aligned}\mathbf{u}_c^{n+1} &= \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W}\left(\mathbf{u}_c^n - \frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n)\right)\\ &= \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W}\mathbf{u}_c^n + \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W}\mathcal{F}(\mathbf{u}_c^n),\end{aligned}$$

with $\mathcal{F}(\mathbf{u}_c^n) := -\frac{1}{\Delta x}\mathbf{\Delta f}(\mathbf{u}_c^n)$.

We define:

$$\hat{\mathbf{u}}_c^{n+1} := \mathbf{u}_c^n + \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W}\mathcal{F}(\mathbf{u}_c^n).$$

from which:

$$\mathbf{u}_c^{n+1} := (\mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}^\top\mathbf{W} - \mathbf{I})\mathbf{u}_c^n + \hat{\mathbf{u}}_c^{n+1}.$$

4.2.1 Step 1: Entropy stability of $\hat{\mathbf{u}}_c^{n+1}$

First, we check whether $\hat{\mathbf{u}}_c^{n+1}$ is entropy stable:

$$\mathbf{v}_c(t) = \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\,\mathbf{v}(t)$$

$$\begin{aligned}\frac{d}{dt}\mathbf{u}(\mathbf{v}_c(t)) &= \frac{d}{dt}\left(\mathbf{u}\left(\mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\,\mathbf{u}(t)\right)\right)\\ &= \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\,\frac{d}{dt}\mathbf{u}(t)\\ &= \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\,\mathcal{F}(\mathbf{u}(\mathbf{v}(t)))\\ &\approx \mathcal{F}(\mathbf{u}(\mathbf{v}_c(t)))\end{aligned}$$

Writing the derivative of the reduced entropy as:

$$\frac{d}{dt}\eta_c(t) = \frac{d}{dt}\eta_c(\mathbf{u}(\mathbf{v}_c(t))) = \mathbf{v}_c^T(t)\frac{d}{dt}\,\mathbf{u}(\mathbf{v}_c(t)),$$

We can rewrite:

$$\mathbf{v}_c^T(t)\,\frac{d}{dt}\,\mathbf{u}(\mathbf{v}_c(t)) \simeq \mathbf{v}^\top(t)\mathbf{W}\mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\mathcal{F}(\mathbf{u}(\mathbf{v}(t))). \quad (4.3)$$

We observe that the obtained term contains the combination:

$$\mathbf{v}^\top(t)\mathbf{W}\mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\underbrace{\mathbf{\Phi}^\top\mathbf{\Phi}}_{\neq\mathbf{I}\text{ in general}}(\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{W}\,\mathcal{F}(\mathbf{u}(\mathbf{v}(t))).$$

If the reduced bases $\mathbf{\Phi}$ were $\mathbf{W}$-orthonormal, we would have $\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi} = \mathbf{I}$ and $\mathbf{\Phi}^\top \mathbf{\Phi} = \mathbf{I}$. Therefore the term would reduce to:

$$\mathbf{v}_c^\top(t)\, \mathcal{F}(\mathbf{u}(\mathbf{v}_c(t))),$$

which coincides with the entropy production of the full model and inherits its dissipativity.

In the general case, however, the factor $\mathbf{\Phi}^\top \mathbf{\Phi}$ is not the identity and this introduces a residual term that does not have a sign determined a priori.

We also observe the projection operator:

$$\mathbf{P} \;:=\; \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W}.$$

Under the assumption that the columns of $\mathbf{\Phi}$ are linearly independent and $\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi}$ is invertible, $\mathbf{P}$ is a square $N \times N$ matrix, idempotent and self-adjoint with respect to the $\mathbf{W}$-inner product, that is $\mathbf{P}^T \mathbf{W} = \mathbf{W}\mathbf{P}$. The image of $\mathbf{P}$ is exactly the subspace range($\mathbf{P}$) and rank($\mathbf{P}$) = rank($\mathbf{\Phi}$) = M.

Consequently, $\mathbf{P}$ acts as the identity only on the elements of range(Φ): $\mathbf{P}\,\mathbf{u} = \mathbf{u} \iff \mathbf{u} \in \text{range}(\mathbf{\Phi})$. Normally one has $M \ll N$, thus rank($\mathbf{P}$) = $M < N$ and therefore $\mathbf{P} \neq \mathbf{I}$.

This observation explains why it is not legitimate to simplify expression (4.3). The fact that $\mathbf{P} \neq \mathbf{I}$ leaves a residual term that does not have a sign determined a priori and prevents deducing the entropy dissipativity of the term $\hat{u}_c^{n+1}$.

Consequently it is not possible to guarantee a priori that $\mathbf{v}_c^\top(t)\, \frac{d}{dt}\mathbf{u}(\mathbf{v}_c(t)) \leq 0$ is entropy stable.

4.2.2 Step 2: The term $(\mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I})\, \mathbf{u}_c^n$

Let:

- $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ with $M \ll N$;
- $\mathbf{W} \in \mathbb{R}^{N \times N}$ be a positive definite symmetric matrix;
- $\mathbf{P} := \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W} \in \mathbb{R}^{N \times N}$ the oblique projector onto Im($\mathbf{\Phi}$) with respect to the weighted scalar product $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{W}} := \mathbf{u}^\top \mathbf{W} \mathbf{v}$;
- $\mathbf{A} := \mathbf{P} - \mathbf{I} \in \mathbb{R}^{N \times N}$.

We want to prove that the matrix A is negative semidefinite, that is:

$$\forall \mathbf{x} \in \mathbb{R}^N,\; \mathbf{x}^\top A \mathbf{x} \leq 0.$$

Let $\mathbf{\Phi} \in \mathbb{R}^{n \times M}$ be a matrix with linearly independent columns and let $\mathbf{W} \in \mathbb{R}^{n \times n}$ be a positive definite symmetric matrix, which induces the weighted scalar product:

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{W}} = \mathbf{u}^\top \mathbf{W} \mathbf{v}.$$

In this context, the weighted orthogonal projector onto the column subspace of $\mathbf{\Phi}$ is defined by the matrix

$$\mathbf{P} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W}. \tag{4.4}$$

The matrix $\mathbf{P}$ has important properties. First of all it is idempotent. In fact, setting $\mathbf{A} := \mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi}$, we have:

$$\mathbf{P}^2 = \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^\top \mathbf{W} = \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{A}\mathbf{A}^{-1}\mathbf{\Phi}^\top \mathbf{W} = \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^\top \mathbf{W} = \mathbf{P},$$

and therefore $\mathbf{P}^2 = \mathbf{P}$. Furthermore, $\mathbf{P}$ is symmetric with respect to the weighted scalar product, in the sense that $\mathbf{P}^\top \mathbf{W} = \mathbf{W}\mathbf{P}$, a property that implies that $\mathbf{P}$ is self-adjoint in the weighted Hilbert space $(\mathbb{R}^n, \langle\cdot,\cdot\rangle_{\mathbf{W}})$. It follows that the eigenvalues of $\mathbf{P}$ are real. Idempotency, combined with weighted symmetry, guarantees that these eigenvalues are either 0 or 1.

First, we show that the matrix $\mathbf{P}$ is idempotent, i.e.:

$$\mathbf{P}^2 = \mathbf{P}.$$

In fact:

$$\begin{aligned}\mathbf{P}^2 &= \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W} \cdot \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W} \\ &= \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}(\mathbf{\Phi}^\top \mathbf{W}\Phi)(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W} \\ &= \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W} = \mathbf{P}.\end{aligned}$$

Now let's calculate the spectrum of $\mathbf{P}$. Since $\mathbf{P}$ is a projector, its eigenvalues belong to the set $\{0, 1\}$. Therefore:

$$\text{Spectrum}(\mathbf{P}) \subset \{0, 1\}.$$

Finally, we verify that $\mathbf{A}$ is negative semidefinite. Since $\mathbf{A} = \mathbf{P} - \mathbf{I}$, we have:

$$\text{Spectrum}(\mathbf{A}) = \text{Spectrum}(\mathbf{P} - \mathbf{I}) \subset \{-1, 0\}.$$

Therefore, for every $\mathbf{x} \in \mathbb{R}^N$, we have:

$$\mathbf{x}^\top \mathbf{A}\mathbf{x} \leq 0,$$

therefore:

$$\mathbf{A} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top \mathbf{W} - \mathbf{I} \quad \text{is negative semidefinite.}$$

4.3 Hyper-reduced model with NNLS weights and extended bases

As observed in the previous section, increasing the number of collocation points improves the accuracy of the reduced solution and ensures ensures better capacity to preserve

entropy. So, to better capture the dynamics and preserve entropy, we perform bases enrichment adding extra bases vectors that correspond to additional collocation points.

Let $\mathbf{u}(x,t;\mu) \in \mathbb{R}^N$ be the solution to the high-fidelity problem.

Starting from the POD bases $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ obtained from the snapshots, we extend the reduced space to incorporate conservative and derivative information.

We define the matrix $\mathbf{\Psi} \in \mathbb{R}^{N\times(2M+1)}$ as

$$\mathbf{\Psi} := [\mathbf{1},\, \mathbf{\Phi},\, \mathbf{Q\Phi}]\,,$$

where

- $\mathbf{1} \in \mathbb{R}^N$ is the constant column vector consisting of all 1s;
- $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ is the POD bases;
- $\mathbf{Q} \in \mathbb{R}^{N\times N}$ is the skew-symmetric matrix defined as

$$\mathbf{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & -1 \\ -1 & 0 & 1 & \cdots & 0 & 0 \\ 0 & -1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & -1 & 0 \end{bmatrix},$$

  which satisfies $\mathbf{Q} = -\mathbf{Q}^T$ [1].

The extended space is defined as:

$$\text{span}(\mathbf{\Phi}_{\text{ext}}) := \text{span}\,(\mathbf{\Phi},\, \mathbf{1},\, \mathbf{\Psi})\,.$$

The extended bases matrix $\mathbf{\Phi}_{\text{ext}} \in \mathbb{R}^{N\times(3M+2)}$ is therefore

$$\mathbf{\Phi}_{\text{ext}} := [\mathbf{\Phi},\, \mathbf{1},\, \mathbf{\Psi}]\,.$$

To define a projection consistent with this reduced space, a diagonal matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$ is constructed that approximates a quadrature formula on the collocation points.

The non-zero weights of $\mathbf{W}$ and their position are determined using NNLS algorithm where $\mathbf{W}_\epsilon$ has only $N_c \ll N$ non-zero elements corresponding to the collocation points.

Using the extended bases $\mathbf{\Phi}_{\text{ext}}$ and the weight matrix $\mathbf{W}$, the projection of the high-fidelity solution $\mathbf{u}_{\text{HF}}^{n+1}$ in the reduced space is obtained by

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}}\mathbf{\Phi}_{\text{ext}}^T\mathbf{W}\mathcal{L}(\mathbf{u}_c^n),$$

or in compact form:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}}\bar{\mathbf{\Phi}}_{\text{ext}}^\top\ \mathbf{W}_\epsilon\overline{\mathcal{L}(\tilde{\mathbf{u}}_c^n)}.$$

where $\mathbf{\Phi}_{\text{ext}} \in \mathbb{R}^{N\times(3M+2)}$, $\bar{\mathbf{\Phi}}_{\text{ext}} \in \mathbb{R}^{N_c\times(3M+2)}$, $\mathbf{W}_\epsilon \in \mathbb{R}^{N_c\times N_c}$, $\bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n) \in \mathbb{R}^{N_c\times 1}$, $\tilde{\mathbf{u}}_c^n \in \mathbb{R}^{N_{cn}\times 1}$. Here N_c are the collocated cells and N_{cn} are the stencil cells.

4.3.1 Projection and reduced bases with weighted inner product

In this section we propose a new version of the previous model, built using a matrix $\mathbf{W}$ such that $\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi} \neq \mathbf{I}$ since the matrix $\mathbf{W}$ is obtained through an NNLS procedure.

Let $\mathbf{u}(t) \in \mathbb{R}^N$ be the high fidelity numerical solution of the Burgers equation at a given time t and let $\mathbf{W} \in \mathbb{R}^{N \times N}$ be a symmetric positive definite diagonal matrix. The $\mathbf{W}$-orthogonal projection of a vector $\mathbf{u}(t)$ onto the reduced space $\mathrm{Im}(\mathbf{\Phi})$ is given by:

$$\mathbf{u}_{\mathrm{proj}}(t) = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W} \mathbf{u}(t),$$

which represents the unique vector in $\mathrm{Im}(\mathbf{\Phi})$ that minimizes the error in the weighted norm:

$$\|\mathbf{u}(t) - \mathbf{u}_{\mathrm{proj}}(t)\|_{\mathbf{W}} = \min_{v \in \mathrm{Im}(\Phi)} \|\mathbf{u}(t) - \mathbf{v}\|_{\mathbf{W}}.$$

To ensure entropy stability of the reduced dynamics, an extended bases is introduced:

$$\mathbf{\Phi}_{\mathrm{ext}} := [\mathbf{1}, \mathbf{\Phi}, \mathbf{Q}\mathbf{\Phi}],$$

where $\mathbf{Q}$ is a discrete operator that approximates the spatial derivative. This bases is used exclusively in the evolution phase of the reduced system:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\mathrm{ext}}(\mathbf{\Phi}_{\mathrm{ext}}^\top \mathbf{W} \mathbf{\Phi}_{\mathrm{ext}})^{-1} \mathbf{\Phi}_{\mathrm{ext}}^\top \mathbf{W} \mathcal{L}(\mathbf{u}_c^n),$$

where $\mathcal{L}(\mathbf{u}_c^n)$ represents the high-fidelity solution.

However, the use of $\mathbf{\Phi}_{\mathrm{ext}}$ in the projection phase would compromise the optimality of the POD method. In fact, the bases $\mathbf{\Phi}$ is constructed as the solution of the following minimization problem:

$$\mathbf{\Phi} = \arg \min_{\substack{\mathbf{\Phi} \in \mathbb{R}^{N \times M} \\ \mathrm{rank}(\mathbf{\Phi}) = M}} \sum_{n=1}^{N_s} \|\mathbf{u}^n - \mathbf{P}_{\mathbf{\Phi}}^{\mathbf{W}} \mathbf{u}^n\|_{\mathbf{W}}^2, \quad \text{with} \quad \mathbf{P}_{\mathbf{\Phi}}^{\mathbf{W}} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W}.$$

Using the extended bases $\mathbf{\Phi}_{\mathrm{ext}}$ in such projection would mean projecting $\mathbf{u}$ onto a larger space:

$$\tilde{\mathbf{u}}_{\mathrm{approx}} = \mathbf{\Phi}_{\mathrm{ext}}(\mathbf{\Phi}_{\mathrm{ext}}^\top \mathbf{W} \mathbf{\Phi}_{\mathrm{ext}})^{-1} \mathbf{\Phi}_{\mathrm{ext}}^\top \mathbf{W} \mathbf{u},$$

which includes vectors not present in the span of the snapshots, such as the constant vector $\mathbf{1}$ and the derivatives $\mathbf{Q}\mathbf{\Phi}$. These components are not contained in the observed data and break the link between the reduced bases and the distribution of real solutions.

Consequently, the correct formula for the reduced approximation of a high-fidelity solution remains:

$$\mathbf{u}_{\mathrm{approx}} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W} \mathbf{u},$$

which guarantees the best fit with respect to the norm $\|\cdot\|_{\mathbf{W}}$.

It follows that the optimal projection of a vector $\mathbf{u} \in \mathbb{R}^N$ onto the space generated by $\mathbf{\Phi}$ is:

$$\mathbf{u}_{\mathrm{approx}} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W} \mathbf{u},$$

which is the unique vector in $\mathrm{Im}(\mathbf{\Phi})$ that minimizes the error $\|\mathbf{u}-\mathbf{v}\|_{\mathbf{W}}$.

The space $\mathrm{Im}(\mathbf{\Phi})$ is built solely from the data, thus it is the best M-dimensional subspace in which to project the snapshots.

The extended bases is defined as:

$$\mathbf{\Phi}_{\text{ext}} := [\mathbf{1}, \mathbf{\Phi}, \mathbf{Q\Phi}] \in \mathbb{R}^{N\times(2M+1)}.$$

Now suppose we use this bases to project a snapshot $\mathbf{u}$:

$$\mathbf{u}_{\text{ext}} = \mathbf{\Phi}_{\text{ext}}(\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathbf{\Phi}_{\text{ext}})^{-1}\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathbf{u}.$$

We note that in general:

$$\mathbf{u}_{\text{ext}} \notin \mathrm{Im}(\mathbf{\Phi}),$$

since $\mathbf{\Phi}_{\text{ext}}$ contains columns that are not in the span of the snapshots, in particular:

- $\mathbf{Q\Phi}$ contains nonlinear spatial derivatives, thus directions not observed in the data,
- $\mathbf{1}$ is a constant vector, also generally outside the span of the snapshots.

This means that the projection $\mathbf{u}_{\text{ext}}$ is no longer optimal with respect to the provided data and may even have a significant component outside the space generated by the snapshots, hence:

$$\|\mathbf{u}-\mathbf{u}_{\text{ext}}\|_{\mathbf{W}} > \|\mathbf{u}-\mathbf{u}_{\text{approx}}\|_{\mathbf{W}}.$$

The extended bases $\mathbf{\Phi}_{\text{ext}}$ is used to compute the weight matrix $\mathbf{W}$ through NNLS, so that:

$$\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathbf{\Phi}_{\text{ext}} \approx \mathbf{I}.$$

In practice, the following minimization problem is solved:

$$\min_{w_i\geq 0}\left\|\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathbf{\Phi}_{\text{ext}}-\mathbf{I}\right\|,$$

that is, one seeks a diagonal matrix $\mathbf{W}=\mathrm{diag}(w_1,\ldots,w_N)$, with non-negative weights, which makes the extended bases orthonormal with respect to the weighted inner product.

The operator $\mathcal{L}(\mathbf{u})$, which describes the evolution of the solution in the reduced system and represents only the coordinates in the bases $\mathbf{\Phi}$, thus the dynamical evolution must remain confined in that space:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}(\mathbf{\Phi}^{\top}\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\top}\mathbf{W}\mathcal{L}(\mathbf{u}_c^n).$$

If the extended bases is used in the advancement:

$$\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}}(\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathbf{\Phi}_{\text{ext}})^{-1}\mathbf{\Phi}_{\text{ext}}^{\top}\mathbf{W}\mathcal{L}(\mathbf{u}_c^n),$$

then the result has components outside the space of the reduced coordinates, leading to numerical instability. This happens because $\mathbf{u}_c^n \in \mathrm{Im}(\mathbf{\Phi})$, hence the solution evolution must take place in that subspace and not in $\mathrm{Im}(\mathbf{\Phi}_{\text{ext}})$, which is much larger.

Therefore, the extended bases is more rich and allows finding $\mathbf{W}$ that makes the bases almost orthonormal in the $\mathbf{W}$-metric, thus ensuring stability and numerical conservation.

4.4 Numerical results

In this section, we present a set of numerical experiments designed to evaluate the practical behavior of the hyper-reduced models introduced in the previous sections, with particular attention to entropy conservation and approximation accuracy.
The tests focus on two complementary strategies for constructing hyper-reduced operators: the NNLS-based approach and the extended-base variant.

The numerical results have two objectives: to verify how much the qualitative properties predicted by theory emerge in practical examples and to understand how the lack of formal guarantees influences the behavior of reduced models. They also provide quantitative data on the accuracy of reduced solutions and their sensitivity to the number of collocation points chosen.

4.4.1 NNLS-based model

As discussed in the first section of this chapter, the theoretical proof of entropy stability cannot be directly extended to the case where the weight matrix $\mathbf{W}$ is obtained through NNLS. The key difficulty is that, unlike the standard case $\mathbf{W} = \Delta x \mathbf{I}$, the NNLS procedure produces a sparse diagonal matrix with non-constant entries. As a consequence, the operator $\mathbf{\Phi}\mathbf{\Phi}^\top\mathbf{W}$ is not an orthogonal projector with respect to the $\mathbf{W}$-weighted inner product and the entropy is not preserved.

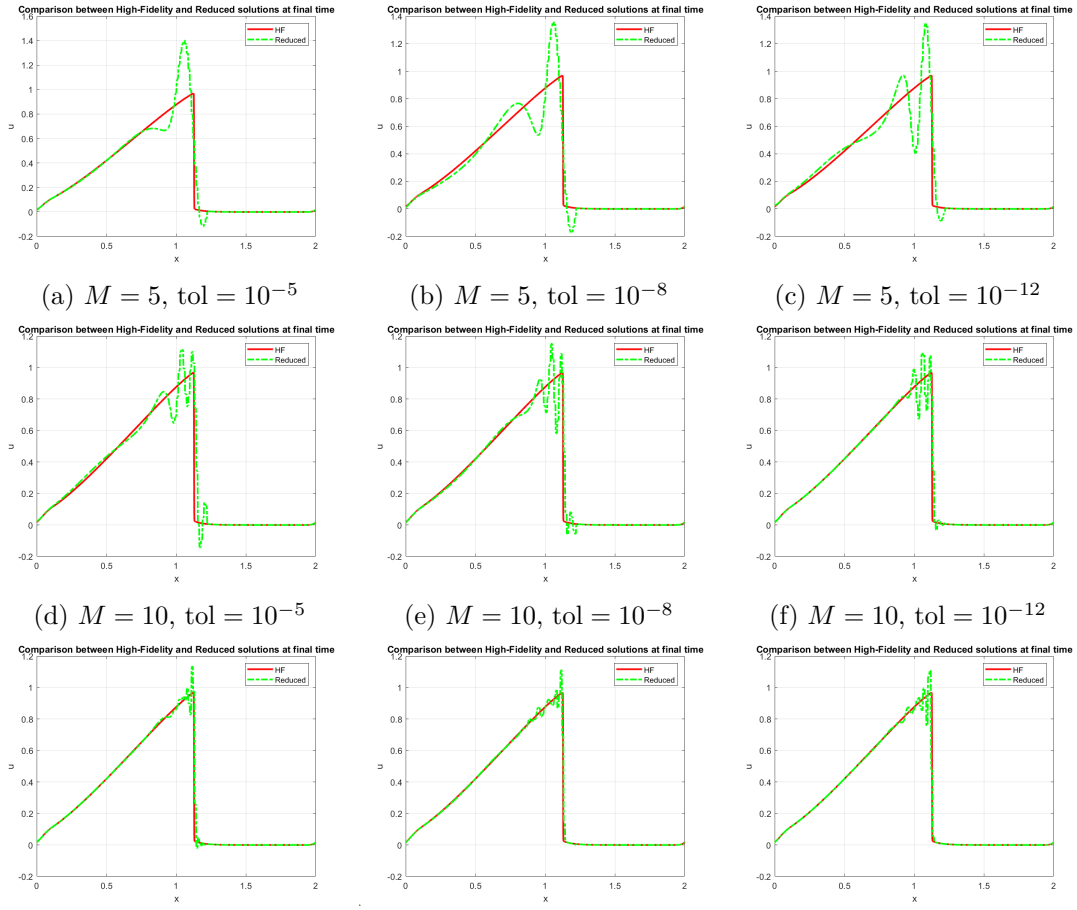
Despite this theoretical limitation, the Matlab simulations reveal that the preservation of entropy depends much more on the number of collocation points effectively selected by NNLS than on the physical properties. In particular, our results show that as the number of collocation points increase, the reduced dynamics displays entropy conservation. So, for practical computations, increasing the richness of the collocation set is more important than relying on a strict theoretical guarantee of entropy preservation.

We now illustrate this behaviour with two examples: a compact exponential initial condition and a sinusoidal one.

Compact exponential as initial condition

In the first numerical experiment we consider as initial condition the compact exponential profile (1.9).

Figure 4.1 compares the high fidelity and reduced order solutions for different values of bases M and different tolerances for the NNLS method at the final simulation time. We observe that larger values of M allow the reduced model to better capture the dynamics of the reference solution. However, the reduced solutions do not always preserve key physical properties, in particular positivity, in fact we note that for small bases sizes, strong oscillations in the negative region are clearly visible.



(a) $M = 5$, tol $= 10^{-5}$ (b) $M = 5$, tol $= 10^{-8}$ (c) $M = 5$, tol $= 10^{-12}$

(d) $M = 10$, tol $= 10^{-5}$ (e) $M = 10$, tol $= 10^{-8}$ (f) $M = 10$, tol $= 10^{-12}$

(g) $M = 40$, tol $= 10^{-5}$ (h) $M = 40$, tol $= 10^{-8}$ (i) $M = 40$, tol $= 10^{-12}$

Figure 4.1: Comparison between the high-fidelity and hyper-reduced solutions of the model $\mathbf{u}_c^{n+1} = \boldsymbol{\Phi}\bar{\boldsymbol{\Phi}}^\top \mathbf{W}_\epsilon \bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n)$ with compact exponential initial condition, for different values of M and NNLS tolerances.

We summarize the performance of the hyper-reduced method in Table 4.1, which reports the relative error computed with respect to the high fidelity solution, the number of collocation points selected by NNLS, the entropy preservation property and the conditioning of $\boldsymbol{\Phi}^\top \mathbf{W} \boldsymbol{\Phi}$ for different values of M and NNLS tolerance.

From the data we observe that for small M and coarse tolerances the number of collocation points is limited, entropy is not preserved and the relative error remains high. However as the tolerance becomes stricter, the NNLS algorithm selects more collocation points and entropy preservation is satisfied.
The conditioning of $\boldsymbol{\Phi}^\top \mathbf{W} \boldsymbol{\Phi}$ remains always close to 1, showing that numerical instabilities are not due to ill conditioning but rather to the insufficient richness of the collocation set.

Tol.	**M**	**Err. Rel.**	**# Coll.**	**Ent. Pres.**	**Cond. $\mathbf{\Phi}^T\mathbf{W}\Phi$**
10^{-5}	2	$6.87543321645768 \times 10^{-1}$	38	No	1.00267990674545
	3	$3.29876543321546 \times 10^{-1}$	54	No	1.00086543454362
	5	$6.40253727691713 \times 10^{-1}$	69	No	1.00069020020147
	10	$4.60953117304925 \times 10^{-1}$	93	No	1.00073478123503
	15	$3.58297726903105 \times 10^{-1}$	110	No	1.00069167766477
	20	$6.20029866321497 \times 10^{-2}$	122	No	1.00062483226846
	40	$5.79048339130534 \times 10^{-2}$	143	No	1.00106248535922
10^{-8}	2	$5.23489632790634 \times 10^{-1}$	45	No	1.00190267395062
	3	$2.75323780753352 \times 10^{-1}$	68	No	1.00083798565345
	5	$5.49288105026196 \times 10^{-1}$	131	No	1.00000584444660
	10	$4.24966676836111 \times 10^{-1}$	152	No	1.00001475945922
	15	$3.13045395847122 \times 10^{-1}$	157	No	1.00002183043410
	20	$6.18567329105403 \times 10^{-2}$	163	Yes	1.00002846340971
	40	$5.47965156789253 \times 10^{-2}$	189	Yes	1.00005856059764
10^{-12}	2	$4.69854321235466 \times 10^{-1}$	76	No	1.00137869090308
	3	$2.38766328998785 \times 10^{-1}$	98	No	1.00075290158762
	5	$5.55798684839837 \times 10^{-1}$	175	No	1.00000002027889
	10	$4.24178584699436 \times 10^{-1}$	195	Yes	1.00000007359620
	15	$2.93118975625424 \times 10^{-1}$	204	Yes	1.00000024102159
	20	$6.14953145678923 \times 10^{-2}$	216	Yes	1.00000014868800
	40	$5.40723765321896 \times 10^{-2}$	254	Yes	1.00000011066825

Table 4.1: Relative error, number of collocation points, entropy preservation and conditioning of $\mathbf{\Phi}^T\mathbf{W}\mathbf{\Phi}$ for $\mathbf{u}_c^{n+1} = \mathbf{\Phi}\bar{\mathbf{\Phi}}^\top \mathbf{W}_\epsilon \bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n)$, varying M and NNLS tolerance.

Sine as initial condition

In the second experiment we consider the inviscid Burgers equation with a sinusoidal initial condition:

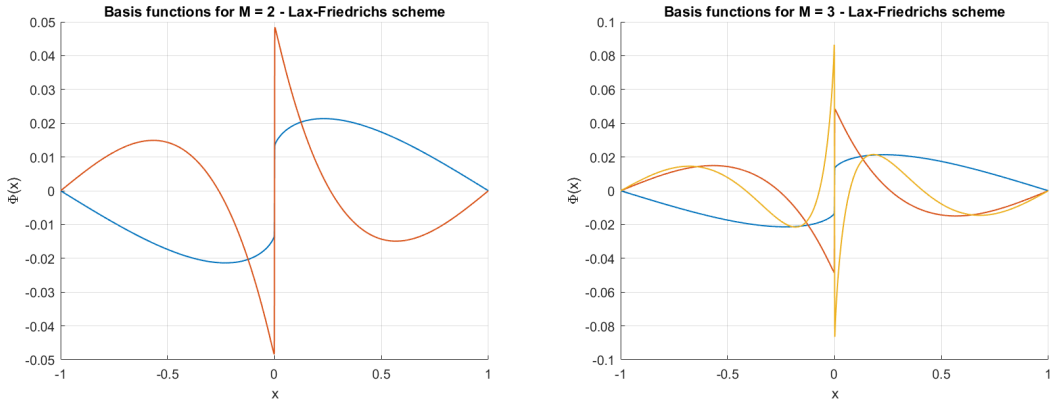
$$u(x,0) = -sin(\pi x). \tag{4.5}$$

As pointed out in [1], this setting is rather delicate since the exact solution $u(x,t)$ evolves into a decaying stationary shock that preserves antisymmetry across the origin for all times t. Consequently, the POD modes remain antisymmetric. However, their spatial derivatives turn out to be nearly symmetric across the origin. This structural mismatch creates a fundamental issue, because the inner products between antisymmetric basis functions and nearly symmetric derivatives are close to zero. As a result, the reduced operators built from such products become nearly singular and the hyper-reduced model obtained in this scenario suffers from a significant loss of accuracy.

This particular behavior also explains the observed difference in the number of collocation points. When comparing the number of points required in this example, reported in Table 4.3, with those of the previous case in Table 4.1, a drastic reduction is observed. The antisymmetric structure of the POD bases, combined with the near-symmetry of their

derivatives, induces the hyper-reduction procedure to select significantly fewer collocation points.

Figure 4.2 shows the POD basis functions computed for different values of the reduced dimension M. As expected the modes appear antisymmetric with respect to the spatial domain, reflecting the structure of the initial condition. Increasing the number of basis functions enriches the approximation space, allowing more accurate representations of the nonlinear dynamics.



(a) $M = 2$ (b) $M = 3$

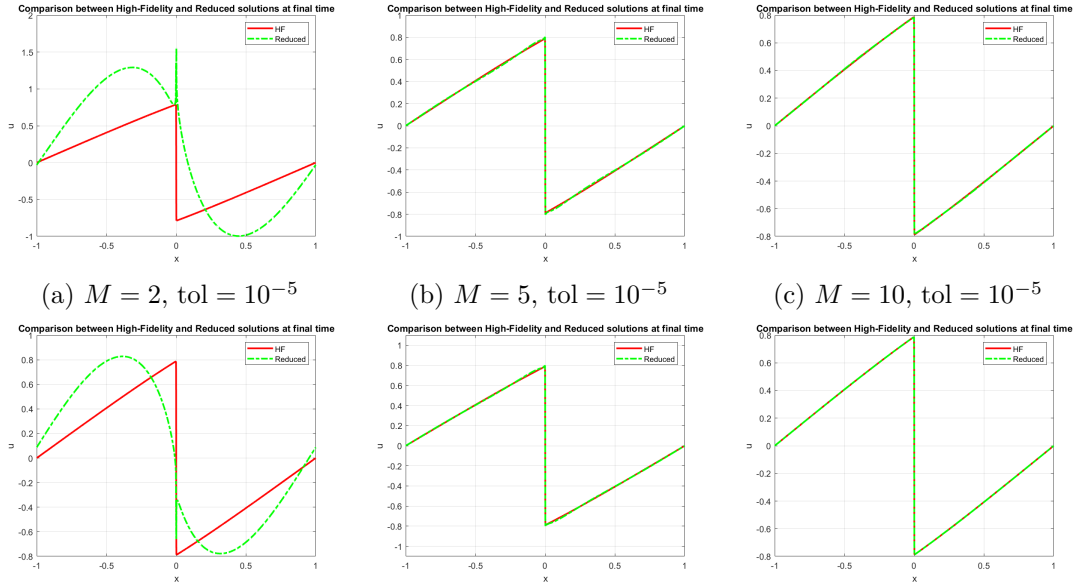
(c) $M = 5$ (d) $M = 10$

Figure 4.2: POD basis functions associated with the sinusoidal initial condition $u(x,0) = -sin(\pi x)$. The antisymmetry of the initial data is clearly reflected in the structure of the modes.

Figure 4.3 show the comparison between the high fidelity and reduced solutions for different values of M.

For small values of M, $M = 2$, the comparison between the high fidelity and reduced solutions is poor: the reduced model relies on too few collocation points to correctly capture the dynamics. Instead, for moderate dimensions, such as $M = 5$, the reduced

solution becomes much closer to the high fidelity one. Therefore, thanks to the symmetries of the POD bases and a sufficient number of placement points, an accurate reduced solution is obtained, even with a moderate number of basis functions.



(a) $M = 2$, tol $= 10^{-5}$ (b) $M = 5$, tol $= 10^{-5}$ (c) $M = 10$, tol $= 10^{-5}$

(d) $M = 2$, tol $= 10^{-12}$ (e) $M = 5$, tol $= 10^{-12}$ (f) $M = 10$, tol $= 10^{-5}$

Figure 4.3: Comparison between the high fidelity and reduced solutions given by $\mathbf{u}_c^{n+1} = \mathbf{\Phi}\,\bar{\mathbf{\Phi}}^\top \mathbf{W}_\epsilon\, \bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n)$, with initial condition $u(x,0) = -\sin(\pi x)$, for different values of M and NNLS tolerances.

We summarize the performance of the method in Table 4.2, which reports the relative error, the number of collocation points selected by NNLS, the entropy preservation property and the conditioning of $\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi}$ for different values of M and tolerance.

The results highlight that for small values of M entropy is generally preserved, even when the number of collocation points is limited.
Nevertheless, the symmetry of the basis functions allows the reduced dynamics to be accurately captured, except when only a very small number of collocation points is available, that is, when the number of considered modes is too low.

This second example demonstrates that entropy preservation depends not only on the density of the collocation set but also on the structural properties of the model. When the POD bases exhibits symmetries compatible with the conservation law, entropy stability is naturally favoured and can be observed even with a relatively small number of collocation points. On the other hand, when such compatibility is absent, as in the compact exponential case, a significant enrichment of both the reduced bases and the collocation set is required to have entropy preservation.

So, although the theoretical proof of entropy stability cannot be extended to the NNLS

Tol.	M	Rel. Err.	# Coll.	Ent.	Cond. $\mathbf{\Phi}^\top \mathbf{W} \Phi$
10^{-3}	2	$2.96472895074931 \times 10^{-1}$	5	No	1.00006895022798
	3	$7.68590231578596 \times 10^{-2}$	7	No	1.00002984301057
	5	$3.20547867820062 \times 10^{-2}$	9	No	1.00687192587156
	10	$3.90264225551920 \times 10^{-3}$	14	Yes	1.71238955960302
	15	$4.20429766200872 \times 10^{-3}$	19	Yes	1.97574940178454
	20	$1.29023636636840 \times 10^{-5}$	24	Yes	2.65894030865132
	40	$9.91086389788753 \times 10^{-7}$	74	Yes	9.84692205683462
10^{-5}	2	$9.36562812878561 \times 10^{-2}$	8	Yes	1.00000538255430
	3	$3.77473939821832 \times 10^{-2}$	11	Yes	1.00000947130653
	5	$2.86213071132473 \times 10^{-2}$	12	Yes	1.00279247854848
	10	$2.87125821381268 \times 10^{-3}$	18	Yes	1.63021285052877
	15	$7.87485447951928 \times 10^{-4}$	22	Yes	1.95541478593272
	20	$1.15092218675906 \times 10^{-5}$	28	Yes	2.38687749361435
	40	$8.26372923643026 \times 10^{-7}$	97	Yes	9.73550570703136
10^{-8}	2	$9.29641344343029 \times 10^{-2}$	12	Yes	1.00000017531362
	3	$3.67310232101809 \times 10^{-2}$	13	Yes	1.00000248758703
	5	$2.59181935973105 \times 10^{-2}$	16	Yes	1.00005490843260
	10	$2.26536320982737 \times 10^{-3}$	23	Yes	1.00074895039863
	15	$5.06617028961148 \times 10^{-4}$	27	Yes	1.07898323659464
	20	$1.01449310730834 \times 10^{-5}$	32	Yes	2.58917479617829
	40	$7.73448629668196 \times 10^{-7}$	140	Yes	6.75598320063845
10^{-12}	2	$9.19189288885916 \times 10^{-2}$	21	Yes	1.00000000117580
	3	$5.07971173872655 \times 10^{-2}$	23	Yes	1.00000001137751
	5	$3.62597498787724 \times 10^{-2}$	25	Yes	1.00000004205208
	10	$2.38905553495455 \times 10^{-3}$	30	Yes	1.00000683527933
	15	$4.67066032518975 \times 10^{-5}$	33	Yes	1.00060149895004
	20	$1.50750677818243 \times 10^{-6}$	39	Yes	1.05352946114740
	40	$6.20488998627510 \times 10^{-7}$	217	Yes	4.41359016421868

Table 4.2: Relative error, collocation points, entropy preservation and conditioning of $\mathbf{P}$ for the model $\mathbf{u}_c^{n+1} = \mathbf{\Phi}\bar{\mathbf{\Phi}}^\top \mathbf{W}_\epsilon \bar{\mathcal{L}}(\tilde{\mathbf{u}}_c^n)$ with initial condition $u(x,0) = -sin(\pi x)$, using different values of M and NNLS tolerances.

case, the numerical results clearly show that the richness of the collocation set is the essential ingredient for entropy preservation. A sufficiently dense set of collocation points effectively compensates for the lack of orthogonality of the projector, ensuring that the reduced dynamics behaves in an entropy consistent way.

4.4.2 Extended-bases model

In the previous section of this chapter we introduced the concept of extended bases. From the theoretical point of view this modification of the reduced order model does not solve

the entropy issue because entropy stability is not guaranteed. Nevertheless, the logic behind their introduction lies in the possibility of enriching the reduced approximation space, thus increasing the number of collocation points and potentially improving both accuracy and robustness of the method.

The numerical experiments reported in this section confirm this intuition. Although the entropy stability of the method is not theoretically restored, in practice we observe that the reduced models with extended bases tend to preserve entropy more frequently, especially for smaller tolerances in the NNLS procedure and yield significantly smaller relative errors with respect to the high fidelity solution. This suggests that, while entropy stability remains an open theoretical limitation, the introduction of extended bases provides a clear numerical advantage.

We now present two test cases: the compact exponential initial condition and the sinusoidal initial condition. Each case is followed by figures and tables that highlight the improvements obtained with extended bases and compared with the results of the previous section without extended bases, see Tables 4.3 and 4.4.

Compact exponential as initial condition

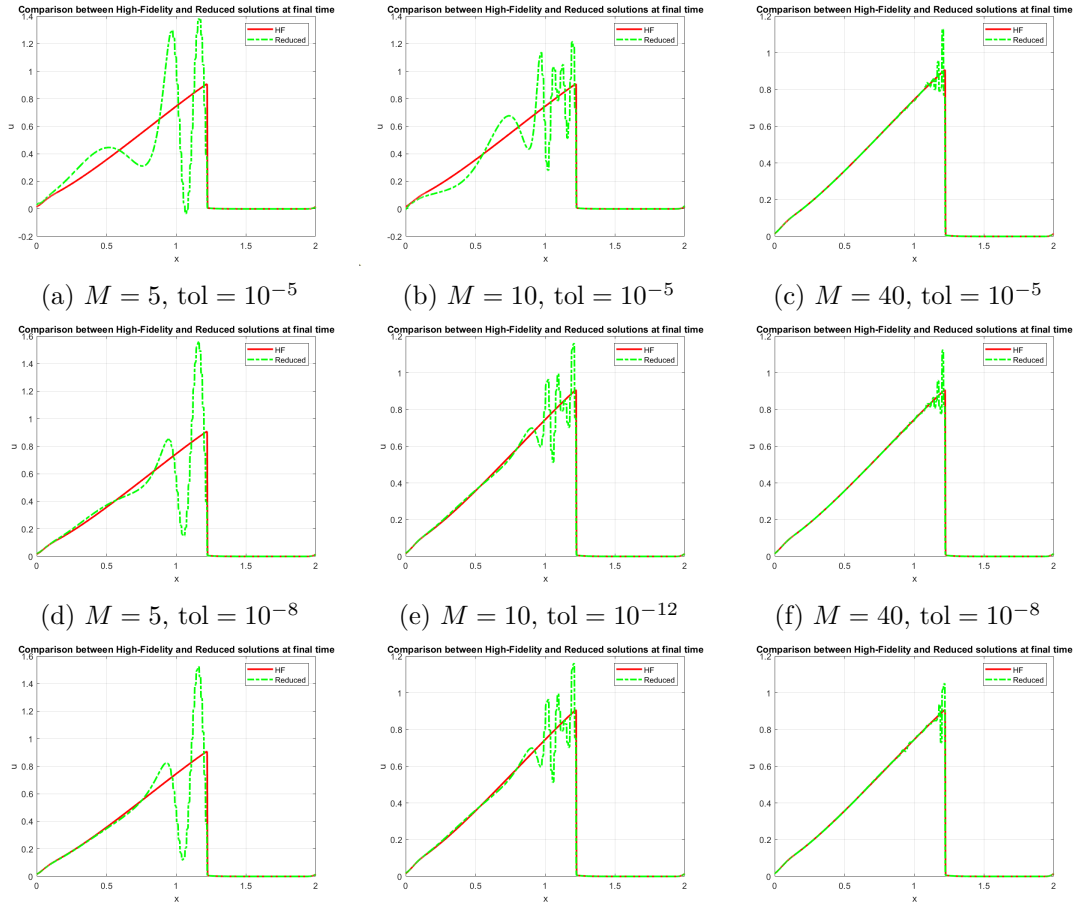
The first test case corresponds to the compact exponential initial condition described in equation (1.9). This example is particularly challenging since the formation of a shock induces oscillations in the reduced approximation, making it a stringent benchmark for stability and accuracy.

From Figure 4.4, we observe that oscillations around the shock are still present but they are significantly reduced compared to the case without extended bases. The overall profile of the reduced solution is closer to the high fidelity one and the positivity property is preserved almost everywhere unlike in the case without extended bases shown in Figure 4.1.

Table 4.3 shows the relative error, the number of collocation points, the entropy preservation property and the conditioning of the projector $\mathbf{P}$, for different values of the reduced dimension M and of the NNLS tolerance. A comparison with the case without extended bases, see Table 4.1, reveals a clear improvement: for the same tolerance, the relative error is consistently reduced and the number of collocation points increases significantly, allowing a richer sampling of the solution space. Moreover, for small NNLS tolerances, numerical entropy preservation is recovered for all values of M, while this was not the case without extended bases.

Finally, let us comment on the conditioning of the projector $\mathbf{P}$.

From the table we observe that in all cases the reported condition number of the projector $\mathbf{P}$ is infinite.
The matrix $\mathbf{P} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{W} \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{W}$, being idempotent, has all eigenvalues equal to 0 or 1.



(a) $M = 5$, tol $= 10^{-5}$ (b) $M = 10$, tol $= 10^{-5}$ (c) $M = 40$, tol $= 10^{-5}$

(d) $M = 5$, tol $= 10^{-8}$ (e) $M = 10$, tol $= 10^{-12}$ (f) $M = 40$, tol $= 10^{-8}$

(g) $M = 5$, tol $= 10^{-12}$ (h) $M = 10$, tol $= 10^{-12}$ (i) $M = 40$, tol $= 10^{-12}$

Figure 4.4: Comparison between high fidelity and hype-reduced solution for the model $\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}} \bar{\mathbf{\Phi}}_{\text{ext}}^{\top} \mathbf{W}_{\epsilon} \overline{\mathcal{L}(\tilde{\mathbf{u}}_c^n)}$ with initial condition the compact exponential, using different values of M and NNLS tolerances.

We now compute the numerical conditioning of the matrix $\mathbf{P}$ in order to understand how sensitive it is to perturbations in the data. For an invertible matrix A, the condition number in the 2-norm is defined as:

$$\kappa(\mathbf{A}) = \|\mathbf{A}\|_2 \cdot \|\mathbf{A}^{-1}\|_2 = \frac{\sigma_{\max}(\mathbf{A})}{\sigma_{\min}(\mathbf{A})}, \tag{4.6}$$

where $\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ are respectively the maximum and minimum singular values of $\mathbf{A}$ [12].

Applying the definition to $\mathbf{P}$ we note that formally $\mathbf{P}$ is not invertible; in fact, having eigenvalues 0 and 1, $\kappa(\mathbf{P}) = \infty$.

The numerical computation of the projector $\mathbf{P}$ requires the inversion of the matrix $\mathbf{A} = \mathbf{\Phi}^{\top}\mathbf{W}\Phi$. If $\mathbf{A}$ is well conditioned, the computation of $\mathbf{P}$ is numerically stable. On the

Tol.	M	Rel. Err.	# Coll.	Ent.	Cond. P
10^{-5}	2	$3.7689024963849537 \times 10^{-1}$	46	No	∞
	3	$2.0555128992227287 \times 10^{-1}$	56	No	∞
	5	$1.7959106759242077 \times 10^{-1}$	75	No	∞
	10	$1.3475387605926206 \times 10^{-1}$	118	No	∞
	15	$7.8301877677028110 \times 10^{-2}$	132	No	∞
	21	$6.1000920979298693 \times 10^{-2}$	135	Yes	∞
	40	$4.1965451346567843 \times 10^{-2}$	154	Yes	∞
10^{-8}	2	$3.7689742915646732 \times 10^{-1}$	80	No	∞
	3	$2.0555128992227287 \times 10^{-1}$	115	Yes	∞
	5	$1.7959106759242077 \times 10^{-1}$	140	Yes	∞
	10	$1.3475387605926206 \times 10^{-1}$	156	Yes	∞
	15	$7.8301877677028110 \times 10^{-2}$	171	Yes	∞
	21	$6.0073595932931932 \times 10^{-2}$	174	Yes	∞
	40	$4.1574146843873654 \times 10^{-2}$	197	Yes	∞
10^{-12}	2	$3.7693589210556984 \times 10^{-1}$	105	No	∞
	3	$2.0555128992227287 \times 10^{-1}$	154	Yes	∞
	5	$1.7959106759242077 \times 10^{-1}$	248	Yes	∞
	10	$1.3475387605926206 \times 10^{-1}$	264	Yes	∞
	15	$7.8301877677028110 \times 10^{-2}$	299	Yes	∞
	21	$5.9284022366302775 \times 10^{-2}$	310	Yes	∞
	40	$4.1480227699486943 \times 10^{-2}$	326	Yes	∞

Table 4.3: Relative error, collocation points, entropy preservation and conditioning of $\mathbf{P}$ for the model with extended bases $\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}}\bar{\mathbf{\Phi}}_{\text{ext}}^{\top}\ \mathbf{W}_{\epsilon}\overline{\mathcal{L}(\tilde{\mathbf{u}}_c^n)}$ and compact exponential initial condition, for different values of M and NNLS tolerances.

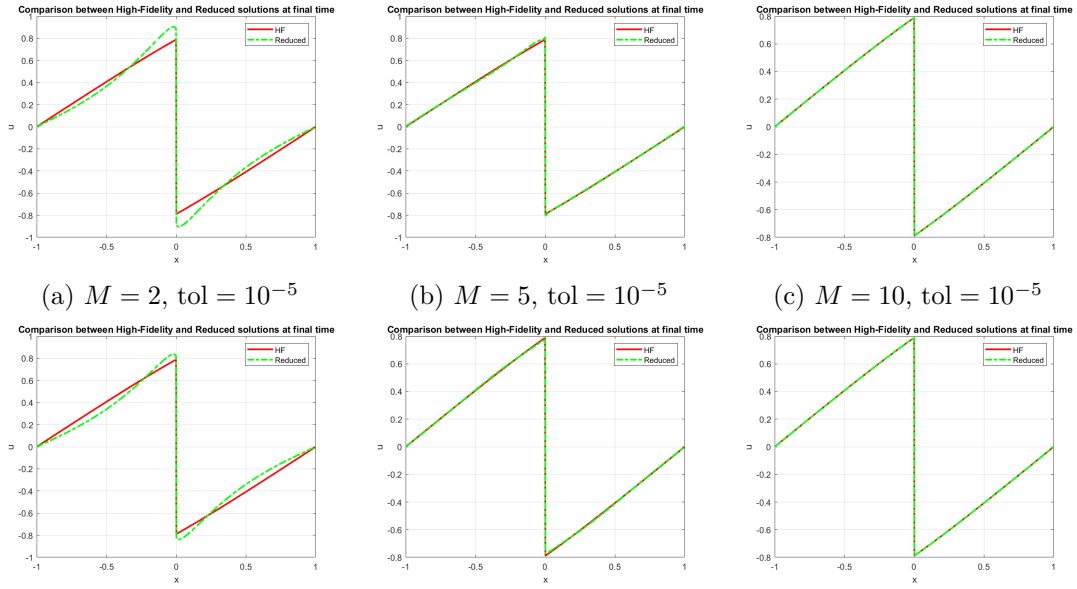
contrary, if $\mathbf{A}$ is nearly singular, that is, it has very small eigenvalues, then the inversion of $\mathbf{A}$ inevitably amplifies numerical errors, making the computation of $\mathbf{P}$ unstable. In such case, while remaining theoretically idempotent and symmetric, the projector $\mathbf{P}$ becomes numerically ill-conditioned and its condition number can be very large.

Sine as initial condition

We now consider the second analysis, done with the sinusoidal initial condition described in equation 4.5.

Figure 4.5 compares the high fidelity and reduced solutions of this case for increasing values of M and different tolerances. The results show that as M increases the reduced solutions reproduce the high fidelity reference with remarkable accuracy. In particular, for $M = 40$ the entropy curves are nearly indistinguishable and the reduced dynamics are almost perfectly aligned with the reference solution. Even for small values of M, such as $M \leq 5$, the accuracy remains satisfactory, with only minor discrepancies appearing after the shock time. It is also interesting to compare the case $M = 2$ in this figure with

the corresponding one in Figure 4.3. In that case, the dynamics were poorly captured because the reduced model relied on too few collocation points. The introduction of extended bases reduces this problem by effectively increasing the number of collocation points, which explains the improved accuracy observed here.



(a) $M = 2$, tol $= 10^{-5}$ (b) $M = 5$, tol $= 10^{-5}$ (c) $M = 10$, tol $= 10^{-5}$

(d) $M = 2$, tol $= 10^{-12}$ (e) $M = 5$, tol $= 10^{-12}$ (f) $M = 10$, tol $= 10^{-5}$

Figure 4.5: Comparison between the high fidelity and reduced solutions given by $\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}}\bar{\mathbf{\Phi}}_{\text{ext}}^{\top}\ \mathbf{W}_\epsilon\overline{\mathcal{L}(\tilde{\mathbf{u}}_c^n)}$, with initial condition $u(x,0) = -\sin(\pi x)$, for different values of M and NNLS tolerances.

Table 4.4 further confirms this trend: the relative errors decrease by several orders of magnitude as M increases and are substantially smaller than those obtained without extended bases, see Table 4.2. Entropy preservation is consistently achieved in all tested configurations, even at moderate values of M, which marks a clear improvement over the non-extended case.

Both test cases confirm that the use of extended bases does not guarantee entropy stability at the theoretical level, but it does improve the numerical behavior of the reduced models: the relative error decreases, entropy preservation is observed more frequently and for smaller tolerances and the number of collocation points increases.
Overall, these results show that extended bases can practically improve NNLS weighted reduced models, even though the theoretical entropy limitation remains.

Tol.	**M**	**Rel. Err.**	**# Coll.**	**Ent.**	**Cond. P**
10^{-3}	2	$7.71059054582031 \times 10^{-2}$	15	No	∞
	3	$3.21180963994359 \times 10^{-2}$	16	No	∞
	5	$1.20547867820062 \times 10^{-2}$	19	Yes	∞
	10	$3.60264225551920 \times 10^{-4}$	25	Yes	∞
	15	$4.20429766200872 \times 10^{-5}$	36	Yes	∞
	20	$1.09023636636840 \times 10^{-5}$	46	Yes	∞
	40	$9.91086389788753 \times 10^{-7}$	122	Yes	∞
10^{-5}	2	$8.20721168515041 \times 10^{-2}$	22	Yes	∞
	3	$3.62896056517128 \times 10^{-2}$	23	Yes	∞
	5	$1.03564261570120 \times 10^{-2}$	26	Yes	∞
	10	$3.67576645835057 \times 10^{-4}$	32	Yes	∞
	15	$7.87485447951928 \times 10^{-5}$	44	Yes	∞
	20	$1.15092218675906 \times 10^{-5}$	51	Yes	∞
	40	$8.26372923643026 \times 10^{-7}$	178	Yes	∞
10^{-8}	2	$8.10921153944064 \times 10^{-2}$	31	Yes	∞
	3	$3.57218327404979 \times 10^{-2}$	32	Yes	∞
	5	$1.03278724040830 \times 10^{-2}$	34	Yes	∞
	10	$3.17103942289129 \times 10^{-4}$	42	Yes	∞
	15	$4.82151892605793 \times 10^{-5}$	49	Yes	∞
	20	$1.01449310730834 \times 10^{-5}$	62	Yes	∞
	40	$7.73448629668196 \times 10^{-7}$	263	Yes	∞
10^{-12}	2	$8.20375911841656 \times 10^{-2}$	37	Yes	∞
	3	$3.57215564782045 \times 10^{-2}$	41	Yes	∞
	5	$1.03125465010772 \times 10^{-2}$	45	Yes	∞
	10	$3.56223343221524 \times 10^{-4}$	51	Yes	∞
	15	$3.79187003392885 \times 10^{-5}$	58	Yes	∞
	20	$1.10210206341099 \times 10^{-6}$	72	Yes	∞
	40	$5.38635429813342 \times 10^{-7}$	353	Yes	∞

Table 4.4: Relative error, collocation points, entropy preservation and conditioning of $\mathbf{P}$ for the model $\mathbf{u}_c^{n+1} = \mathbf{\Phi}_{\text{ext}} \bar{\mathbf{\Phi}}_{\text{ext}}^{\top} \mathbf{W}_{\epsilon} \overline{\mathcal{L}(\tilde{\mathbf{u}}_c^n)}$ with initial condition $u(x,0) = -sin(\pi x)$, using different values of M and NNLS tolerances.

Conclusion

This thesis analyses the problem of numerical model reduction, with reference to the Burgers equation as a case study and the fundamental issue of entropy stability.

The work is divided into four chapters. Following the introduction of the main reduction techniques, such as Proper Orthogonal Decomposition and Nonnegative Least Squares, the analysis focused on the Burgers equation and the phenomenon of discontinuity formation, using the Lax-Friedrichs scheme as a high fidelity reference and applying reduction via POD.

A significant result has been obtained in the case of reduced models with orthonormal POD bases compared to a scalar product weighted by a constant diagonal matrix. In this context, it has been possible to theoretically demonstrate and numerically verify entropy conservation: the reduced model maintains one of the fundamental structural properties of the original problem, ensuring stability and physical consistency.

The situation is different in the case of hyper-reduction based on collocation points selected using NNLS. In this regime, the theoretical demonstration of entropy stability is not valid and numerical tests also showed difficulties in preserving entropy. This highlighted an intrinsic limitation of the approach, linked to the role of the weights calculated by NNLS and the loss of structural orthogonality induced by the collocation.

To overcome these problems, two improvement strategies are considered. The first involved projecting the non-linear operator into a space weighted by W, obtaining a formulation more consistent with the entropic structure but not entirely satisfactory from a theoretical point of view. The second introduced an extended bases, enriching the reduced space with additional vectors linked to conservative and derived information. Although not satisfying a rigorous theoretical demonstration, this choice made it possible to obtain numerically more accurate solutions and to better preserve entropy even with a reduced number of bases. It is also observed that an increase in the number of collocation points and a lower tolerance of the NNLS algorithm lead to better entropy preservation, highlighting a balance to be sought between computational efficiency and physical fidelity.

These results offer several perspectives for future research. In particular, it will be essential to develop a more rigorous theory for hyper-reduced models, capable of clarifying

and guaranteeing entropy stability.
For the construction of physically consistent models, it is also crucial to study and include additional structural properties, such as the positivity of variables and the conservation of physical invariants, which are essential for the reliability of simulations.
A further direction of investigation concerns the exploration of alternative techniques for the selection of collocation points and for the construction of the weight matrix, beyond the NNLS-based approach.

In conclusion, the work done confirms the potential of collocation-based Reduced Order Models as reliable and promising reduction tools, but also highlights the critical issues that remain unresolved, particularly concerning hyper-reduction. The hope is that the ideas and results presented here will provide a useful bases for further developments towards reduced models becoming increasingly accurate, stable and computationally efficient.

Appendix A

Entropy conservative reduced order formulation

In this appendix we summarise the main steps of the proof presented in the article *Entropy stable reduced order modeling of nonlinear conservation laws* [1]. The aim is to show how entropy stability at the full order level can be derived from the reduced order model.

Starting from an entropy stable discretisation of the high fidelity model, the author constructs a low dimensional model that conserves entropy in semi-discrete form and, using hyper-reduction techniques, allows efficient computation of nonlinear terms.

A.0.1 Entropy conservative formulation of FOM

A central feature of both the full and reduced models is the use of entropy conservative numerical flows in finite volume formulation. Let $\mathbf{u}_L$ and $\mathbf{u}_R$ be two (conservative) states on the left and right. A two-point numerical flow $\mathbf{f}_s(\mathbf{u}_L, \mathbf{u}_R)$ is said to be entropy conservative if it satisfies the following three properties:

1. Consistency: $\mathbf{f}_s(\mathbf{u}, \mathbf{u}) = \mathbf{f}(\mathbf{u})$

2. Symmetry: $\mathbf{f}_s(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{f}_s(\mathbf{u}_R, \mathbf{u}_L)$

3. Entropy conservation: $(\mathbf{v}_L - \mathbf{v}_R)^T \mathbf{f}_s(\mathbf{u}_L, \mathbf{u}_R) = \psi(\mathbf{u}_L) - \psi(\mathbf{u}_R)$

where $\mathbf{v} = \frac{\partial \psi}{\partial \mathbf{u}}$ is the entropy variable and $\psi(\mathbf{u})$ is the entropy potential.

Let the domain be divided into K cells of size Δx and let $(\mathbf{u}_h)_i$ be the mean of the conservative variable on the i-th cell. Then, an entropy conservative finite volume scheme

can be written as:

$$\begin{aligned}
\frac{d(\mathbf{u}_h)_1}{dt} + \frac{\mathbf{f}_s((\mathbf{u}_h)_2, (\mathbf{u}_h)_1) - \mathbf{f}_s((\mathbf{u}_h)_1, (\mathbf{u}_h)_K)}{\Delta x} &= \mathbf{0} \\
\frac{d(\mathbf{u}_h)_i}{dt} + \frac{\mathbf{f}_s((\mathbf{u}_h)_{i+1}, (\mathbf{u}_h)_i) - \mathbf{f}_s((\mathbf{u}_h)_i, (\mathbf{u}_h)_{i-1})}{\Delta x} &= \mathbf{0}, \quad 1 < i < K \\
\frac{d(\mathbf{u}_h)_K}{dt} + \frac{\mathbf{f}_s((\mathbf{u}_h)_1, (\mathbf{u}_h)_K) - \mathbf{f}_s((\mathbf{u}_h)_K, (\mathbf{u}_h)_{K-1})}{\Delta x} &= \mathbf{0}
\end{aligned}$$

where periodicity is imposed through the equations for $(\mathbf{u}_h)_1$ and $(\mathbf{u}_h)_K$.

To rewrite the system in a matrix form more suitable for model reduction, we define:

- a skew-symmetric differentiation matrix $\mathbf{Q} \in \mathbb{R}^{K \times K}$:

$$\mathbf{Q} = \frac{1}{2} \begin{bmatrix} 0 & 1 & & \cdots & -1 \\ -1 & 0 & 1 & & \\ & -1 & 0 & 1 & \\ & & & \ddots & \\ 1 & & & \cdots & 0 \end{bmatrix}$$

- a flux matrix $\mathbf{F}$ with $\mathbf{F}_{ij} = \mathbf{f}_s((\mathbf{u}_h)_i, (\mathbf{u}_h)_j)$.

Let $\mathbf{1}$ be the column vector of all 1s. Then, the semi-discrete formulation can be written as:

$$\Delta x \frac{d\mathbf{u}_h}{dt} + 2(\mathbf{Q} \circ \mathbf{F})\mathbf{1} = \mathbf{0} \tag{A.1}$$

where $\circ$ indicates the Hadamard product, that is an element-by-element product.

This rewriting highlights the nonlinear interactions between grid points through the combination of $\mathbf{Q}$ and $\mathbf{F}$.

Now that we have written the equation in matrix form (A.1), we want to prove that it conserves entropy in a semi-discrete manner. Suppose we have a scalar conservation law and that the numerical flux $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is entropy conservative. We will then prove that the numerical system conserves entropy over time, i.e. that:

$$\Delta x \cdot \mathbf{1}^T \frac{dS(\mathbf{u}_h)}{dt} = 0$$

where $S(\mathbf{u}_h)$ is the entropy function evaluated on the discrete values of the solution.

We multiply equation (A.1) scalarly by the vector of entropy variables $\mathbf{v}_h = \frac{dS}{d\mathbf{u}}(\mathbf{u}_h)$:

$$\Delta x \cdot \mathbf{v}_h^T \frac{d\mathbf{u}_h}{dt} + \mathbf{v}_h^T \cdot 2(\mathbf{Q} \circ \mathbf{F}) \cdot \mathbf{1} = 0$$

Since $\mathbf{v}_h = \frac{dS}{d\mathbf{u}}(\mathbf{u}_h)$, then:

$$\mathbf{v}_h^T \frac{d\mathbf{u}_h}{dt} = \sum_j \frac{dS((\mathbf{u}_h)_j)}{dt} = \mathbf{1}^T \frac{dS(\mathbf{u}_h)}{dt}$$

So the first term becomes:

$$\Delta x \cdot \mathbf{1}^T \frac{dS(\mathbf{u}_h)}{dt}$$

We expand the second term using Hadamard's product:

$$\mathbf{v}_h^T \cdot 2(\mathbf{Q} \circ \mathbf{F}) \cdot \mathbf{1} = 2 \sum_{i,j} \mathbf{Q}_{ij} \cdot \mathbf{v}_i^T \mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j)$$

Since $\mathbf{Q}_{ij} = -\mathbf{Q}_{ji}$ and $\mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j) = \mathbf{f}_S(u_j, u_i)$ due to antisymmetry and symmetry respectively we can rewrite the sum as:

$$\sum_{i,j} \mathbf{Q}_{ij} \cdot ((\mathbf{v}_h)_i - (\mathbf{v}_h)_j)^T \mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j)$$

This reformulation is possible thanks to the combination of the symmetries of $\mathbf{f}_S$ and the antisymmetry of $\mathbf{Q}$.

Now let's use the fundamental property $\mathbf{Q} \cdot \mathbf{1} = \mathbf{0}$. Then:

$$\sum_{i,j} \mathbf{Q}_{ij} \cdot ((\mathbf{v}_h)_i - (\mathbf{v}_h)_j)^T \mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j) = 0$$

because each difference $((\mathbf{v}_h)_i - (\mathbf{v}_h)_j)^T \mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j)$ is compensated for in the sum due to the antisymmetry of $\mathbf{Q}$.

Adding the two terms obtained:

$$\Delta x \cdot \mathbf{1}^T \frac{dS(\mathbf{u}_h)}{dt} + 0 = 0 \quad \Rightarrow \quad \Delta x \cdot \mathbf{1}^T \frac{dS(\mathbf{u}_h)}{dt} = 0$$

which is the entropy conservation in the semi-discrete form [1].

A.0.2 Entropy conservative reduced model

Let us now consider a reduced bases $\{\phi_j(x)\}_{j=1}^N$, obtained, for example, using POD. Let $\mathbf{V} \in \mathbb{R}^{K \times N}$ be the matrix of evaluations of the basis functions on the grid points:

$$\mathbf{V}_{ij} = \phi_j(x_i)$$

and assume that $\mathbf{u}_h \approx \mathbf{V}\mathbf{u}_N$, where $\mathbf{u}_N$ contains the coefficients of the reduced model.

Substituting this approximation into the matrix formulation (A.1) and requiring that the residual be orthogonal to all columns of $\mathbf{V}$ (Galerkin projection), we obtain the following reduced system:

$$\Delta x \mathbf{V}^T \mathbf{V} \frac{d\mathbf{u}_N}{dt} + 2\mathbf{V}^T(\mathbf{Q} \circ \mathbf{F})\mathbf{1} = \mathbf{0} \tag{A.2}$$

where $\mathbf{F}_{ij} = \mathbf{f}_s((\mathbf{u}_h)_i, (\mathbf{u}_h)_j)$ is still evaluated at the grid nodes using the approximate solution $\mathbf{u}_h = \mathbf{V}\mathbf{u}_N$.

However, this formulation (A.2) does not guarantee entropy conservation. This is because, unlike the complete model, the entropy variables $\mathbf{v} = \frac{\partial \psi}{\partial \mathbf{u}}$ may not belong to the space generated by the basis functions $\mathbf{V}$, thus preventing direct testing with $\mathbf{v}$.

To overcome this, we introduce the projection of the entropy variables $\mathbf{v}(\mathbf{V}\mathbf{u}_N)$ onto the reduced space. We therefore define:

$$\mathbf{v}_N := (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{v}(\mathbf{V}\mathbf{u}_N) = \mathbf{V}^\dagger\mathbf{v}(\mathbf{V}\mathbf{u}_N)$$

where $\mathbf{V}^\dagger$ is the pseudoinverse of $\mathbf{V}$. The entropy variables projected onto the grid nodes are then obtained as:

$$\mathbf{v}_e := \mathbf{V}\mathbf{v}_N = \mathbf{V}\mathbf{V}^\dagger\mathbf{v}(\mathbf{V}\mathbf{u}_N).$$

Now we can test the system with $\mathbf{v}_N$. The first term (the time derivative) becomes:

$$\mathbf{v}_N^T\mathbf{V}^T\mathbf{V}\frac{d\mathbf{u}_N}{dt} = \mathbf{v}(\mathbf{V}\mathbf{u}_N)^T\mathbf{V}\mathbf{V}^T\mathbf{V}^{-1}\mathbf{V}^T\mathbf{V}\frac{d\mathbf{u}_N}{dt} \tag{A.3}$$

$$= \mathbf{v}(\mathbf{V}\mathbf{u}_N)^T\frac{d(\mathbf{V}\mathbf{u}_N)}{dt} = \mathbf{1}^T\frac{dS(\mathbf{V}\mathbf{u}_N)}{dt} \tag{A.4}$$

where we have used the chain rule for $S = \psi(u)$ and the fact that $\mathbf{v} = \partial\psi/\partial u$. The nonlinear term remains to be evaluated:

$$\mathbf{v}_N^T\mathbf{V}^T(\mathbf{Q}\circ\mathbf{F})\mathbf{1} = \mathbf{v}_e^T(\mathbf{Q}\circ\mathbf{F})\mathbf{1}$$

where $\mathbf{v}_e$ are the nodal values of the projected entropy variables. Expanding this product as a double sum:

$$\mathbf{v}_e^T(\mathbf{Q}\circ\mathbf{F})\mathbf{1} = \sum_{i,j}\mathbf{Q}_{ij}(\mathbf{v}_e^{(i)} - \mathbf{v}_e^{(j)})^T\mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j) \tag{A.5}$$

we note that, in order to satisfy entropy conservation, the following should hold:

$$(\mathbf{v}_e^{(i)} - \mathbf{v}_e^{(j)})^T\mathbf{f}_S((\mathbf{u}_h)_i, (\mathbf{u}_h)_j) = \psi((\mathbf{u}_h)_i) - \psi((\mathbf{u}_h)_j)$$

However, this is not guaranteed because, unlike in the complete case, $\mathbf{v}_e$ is not generally a function of $\mathbf{u}_h$ but is obtained by projection and is therefore decoupled from the map $\mathbf{u} \mapsto \mathbf{v}$.

To correct this defect, the entropy map is inverted. A new projected conservative variable $\mathbf{u}_e$ is defined, obtained as:

$$\mathbf{u}_e := \mathbf{u}(\mathbf{v}_e) = \mathbf{u}(\mathbf{V}\mathbf{V}^\dagger\mathbf{v}(\mathbf{V}\mathbf{u}_N))$$

i.e., the values of $\mathbf{u}$ are reconstructed starting from the projected values of the entropy variables. This new variable $\mathbf{u}_e$ still depends on $\mathbf{u}_N$, but through the transition $\mathbf{u}_N \mapsto \mathbf{v} \mapsto \mathbf{v}_N \mapsto \mathbf{v}_e \mapsto \mathbf{u}_e$.

Using $\mathbf{u}_e$ to construct the flux matrix $\mathbf{F}$, i.e.:

$$\mathbf{F}_{ij} = \mathbf{f}_s(\mathbf{u}_e^{(i)}, \mathbf{u}_e^{(j)})$$

and substituting in (A.2), we obtain a new entropy conservative formulation in semi-discrete form:

$$\Delta x \mathbf{V}^T \mathbf{V} \frac{d\mathbf{u}_N}{dt} + 2\mathbf{V}^T(\mathbf{Q} \circ \mathbf{F})\mathbf{1} = 0 \tag{A.6}$$

where $\mathbf{F}$ is constructed from $\mathbf{u}_e$ and therefore from the conservative values derived from the projected entropy variables.

We now want to demonstrate semi-discrete conservation of entropy. Let $\mathbf{u}_N$ be the coefficients of the reduced solution that satisfies (A.6) with:

$$\mathbf{u}_e = \mathbf{u}(\mathbf{V}\mathbf{V}^\dagger \mathbf{v}(\mathbf{V}\mathbf{u}_N))$$
$$\mathbf{F}_{ij} = \mathbf{f}_S(\mathbf{u}_e^{(i)}, \mathbf{u}_e^{(j)})$$

Then, the following entropy conservation equation is verified:

$$\Delta x \mathbf{1}^T \frac{dS(\mathbf{V}\mathbf{u}_N)}{dt} = 0 \tag{A.7}$$

Furthermore, if the constant vector $\mathbf{1}$ belongs to the image of $\mathbf{V}$, i.e. $\exists \mathbf{e}$ such that $\mathbf{V}\mathbf{e} = \mathbf{1}$, then the global average of the conservative variables is also conserved:

$$\Delta x \frac{d}{dt} \mathbf{1}^T (V\mathbf{u}_N) = 0$$

Entropy conservation follows directly by testing the equation with $\mathbf{v}_N$ and applying the relation:

$$\mathbf{v}_N^T \mathbf{V}^T \mathbf{V} \frac{d\mathbf{u}_N}{dt} = \mathbf{1}^T \frac{dS(\mathbf{V}\mathbf{u}_N)}{dt}$$

The conservation of the mean follows by testing with $\mathbf{e}$ such that $\mathbf{V}\mathbf{e} = \mathbf{1}$:

$$\Delta x \mathbf{e}^T \mathbf{V}^T \mathbf{V} \frac{d\mathbf{u}_N}{dt} + 2\mathbf{e}^T \mathbf{V}^T (\mathbf{Q} \circ \mathbf{F})\mathbf{1} = \Delta x \frac{d}{dt} \mathbf{1}^T (\mathbf{V}\mathbf{u}_N) + 2\mathbf{1}^T (\mathbf{Q} \circ \mathbf{F})\mathbf{1} = 0$$

since $\mathbf{Q}$ is skew-symmetric, $\mathbf{F}$ is symmetric and therefore $\mathbf{Q} \circ \mathbf{F}$ is also skew-symmetric [1]. It follows that:

$$\mathbf{1}^T (\mathbf{Q} \circ \mathbf{F})\mathbf{1} = 0$$

and therefore:

$$\Delta x \mathbf{1}^T \frac{d(\mathbf{V}\mathbf{u}_N)}{dt} = 0.$$

A.0.3 Entropy conservative hyper-reduction

While the Galerkin-projected formulation presented in equation (A.2) is entropy stable, its computational cost remains tied to the full order mode due to the presence of nonlinear terms such as $\mathbf{V}^T(\mathbf{Q} \circ \mathbf{F})$. Since $\mathbf{Q}$ and $\mathbf{F}$ are derived from the full model, evaluating such terms is still expensive and scales poorly with respect to the reduced bases dimension.

To mitigate this cost, we apply a second hyper-reduction step. In particular, we adopt a sampling and weighting strategy which enables entropy stable hyper-reduction through the approximation of nonlinear terms as:

$$\mathbf{V}^T g(\mathbf{V}\mathbf{u}_N) \approx \mathbf{V}(I,:)^T \mathbf{W}\, g(\mathbf{V}(I,:)\mathbf{u}_N), \tag{A.8}$$

where:

- $g(\mathbf{u})$ is a nonlinear function (e.g., flux),
- I is a subset of N_s indices corresponding to sampled spatial points,
- $\mathbf{V}(I,:)$ is the submatrix of $\mathbf{V}$ restricted to rows indexed by I,
- $\mathbf{W} = \mathrm{diag}(\mathbf{w}) \in \mathbb{R}^{N_s \times N_s}$ is a diagonal matrix of positive weights.

Instead of hyper-reducing the nonlinear vector term directly, we consider a matrix-based hyper-reduction that respects the structure of the product $\mathbf{Q} \circ \mathbf{F}$. To this end, we define a reduced matrix $\mathbf{Q_s}$ and approximate:

$$\mathbf{V}^T(\mathbf{Q} \circ \mathbf{F}) \approx \mathbf{V}(I,:)^T \mathbf{W}(\mathbf{Q_s} \circ \mathbf{F_s}), \tag{A.9}$$

where $\mathbf{Q_s} \in \mathbb{R}^{N_s \times N_s}$ is a reduced version of $\mathbf{Q}$ and $\mathbf{F_s} \in \mathbb{R}^{N_s \times N_s}$ contains nonlinear flux evaluations at hyper-reduced points.

To guarantee entropy conservation, the matrix $\mathbf{Q_s}$ must satisfy:

$$\mathbf{Q_s} = -\mathbf{Q_s}^T, \quad \mathbf{Q_s}\mathbf{1} = 0, \tag{A.10}$$

i.e., $\mathbf{Q_s}$ must be skew-symmetric with zero row sum.

Unfortunately, typical hyper-reduction techniques preserve at most one of these properties. To ensure both, we adopt a two-step hyper-reduction process that combines modal compression with projection.

The first step consists in constructing a test bases $\mathbf{V_t} \in \mathbb{R}^{N \times r_t}$ such that the reduced bases $\mathbf{V} \subseteq \mathrm{range}(\mathbf{V_t})$. This allows us to define an intermediate operator that satisfies the necessary structural conditions:

$$\hat{\mathbf{Q}} = \mathbf{V_t}^T \mathbf{Q} \mathbf{V_t}. \tag{A.11}$$

This operator is skew-symmetric by construction and if $\mathbf{1} \in \mathrm{range}(\mathbf{V_t})$, then:

$$\hat{\mathbf{Q}}\mathbf{e} = \mathbf{0} \quad \text{for} \quad \mathbf{1} = \mathbf{V_t}\mathbf{e}.$$

The second step consists in projecting from a selected subset of hyper-reduced points. Let I be the index set of sampled spatial locations and define the corresponding test mass matrix:

$$\mathbf{M_t} = \mathbf{V_t}(I,:)^T \mathbf{W} \mathbf{V_t}(I,:) \in \mathbb{R}^{r_t \times r_t}. \tag{A.12}$$

We assume the matrix $\mathbf{M_t}$ is invertible. Under this assumption, we define the projection matrix:

$$\mathbf{P_t} = \mathbf{M_t}^{-1}\mathbf{V_t}(I,:)^T\mathbf{W}. \tag{A.13}$$

The projection property ensures that for any vector $\mathbf{f} = \mathbf{V_t}(I,:)\mathbf{c}$, we recover $\mathbf{c}$ exactly:

$$\mathbf{P_t}\mathbf{f} = \mathbf{M_t}^{-1}\mathbf{V_t}(I,:)^T\mathbf{W}\mathbf{V_t}(I,:)\mathbf{c} = \mathbf{c}. \tag{A.14}$$

Based on this, we proceed to define the nodal differentiation matrix, which serves as the discrete counterpart of the continuous operator evaluated at the sampled points:

$$\mathbf{Q_t} = \mathbf{P_t}^T\hat{\mathbf{Q}}\mathbf{P_t} = \mathbf{P_t}^T\mathbf{V_t}^T\mathbf{Q}\mathbf{V_t}\mathbf{P_t}. \tag{A.15}$$

Suppose $\mathbf{1} \in \text{range}(\mathbf{V_t})$ and Assumption 1 holds. Then, the matrix $\mathbf{Q_t}$ is skew-symmetric and satisfies:

$$\mathbf{Q_t} = -\mathbf{Q_t}^T, \quad \mathbf{Q_t}\mathbf{1} = \mathbf{0}.$$

From equation (A.15), skew-symmetry follows from the skew-symmetry of $\mathbf{Q}$. Also, if $\mathbf{1} = \mathbf{V_t}(I,:)\mathbf{e}$, then $\mathbf{P_t}\mathbf{1} = \mathbf{e}$ and:

$$\mathbf{Q_t}\mathbf{1} = \mathbf{P_t}^T\hat{\mathbf{Q}}\mathbf{e} = \mathbf{P_t}^T \cdot \mathbf{0} = \mathbf{0}.$$

This result ensures that the hyper-reduced matrix $\mathbf{Q_t}$ retains the essential properties required for the entropy conservation proof, just as in the full order model.

Note that if the hyper-reduction is performed over all grid points (i.e., $I = \{1, \ldots, N\}$) and $\mathbf{W} = \Delta x \cdot \mathbf{I}$, then $\mathbf{P_t}$ reduces to the pseudoinverse:

$$\mathbf{P_t} = (\mathbf{V_t}^T\mathbf{V_t})^{-1}\mathbf{V_t}^T.$$

The construction above enables an entropy conservative semi-discrete ROM whose nonlinear terms are evaluated using only a subset of spatial points.

To reduce the computational cost associated with evaluating the nonlinear term in an entropy conservative reduced model, we propose an approach that preserves structural properties, namely skew-symmetry and entropy conservation, while using a reduced number of points.

A reduced test bases $\mathbf{V_t}$ is introduced, constructed such that:

$$\mathcal{R}(\mathbf{V_t}) = R([\mathbf{1}, \mathbf{V}, \mathbf{Q}\mathbf{V}])$$

and uses $\mathbf{V_t}$ to construct the reduced modal matrix $\hat{\mathbf{Q}}$, see (A.11).

This matrix inherits the skew-symmetry of $\mathbf{Q}$, i.e. $\hat{\mathbf{Q}}^T = -\hat{\mathbf{Q}}$ and satisfies $\hat{\mathbf{Q}}\mathbf{1} = \mathbf{0}$.

To apply hyperreduction, select a subset $I \subset \{1, \ldots, N\}$ of indices associated with hyper-reduced points and construct the projection matrix:

$$\mathbf{P_t} = \left(\mathbf{V_t}(I,:)^T\mathbf{W}\mathbf{V_t}(I,:)\right)^{-1}\mathbf{V_t}(I,:)^T\mathbf{W}$$

where $\mathbf{W}$ is a diagonal matrix with quadrature weights.

The hyper-reduced differentiation matrix is then defined as:

$$\mathbf{Q_t} = \mathbf{P_t}^T \hat{\mathbf{Q}} \mathbf{P_t}$$

This matrix maintains skew-symmetry and the property $\mathbf{Q_t 1} = \mathbf{0}$, which are essential for ensuring entropy conservation.

Once $\mathbf{V}$, $\mathbf{V_t}$, $\mathbf{P_t}$ and $\mathbf{Q_t}$ have been constructed, the hyper-reduced entropy conservative formulation can be written as:

$$\mathbf{M_N} \frac{d\mathbf{u}_N}{dt} + 2\mathbf{V}(I,:)^T (\mathbf{Q_t} \circ \mathbf{F})\mathbf{1} = \mathbf{0}. \tag{A.16}$$

Here:

$$\begin{aligned}
\mathbf{M_N} &= \mathbf{V}(I,:)^T \, \mathbf{W} \, \mathbf{V}(I,:), \\
\mathbf{P} &= {\mathbf{M_N}}^{-1} \, \mathbf{V}(I,:)^T \, \mathbf{W}, \\
\mathbf{v}_N &= \mathbf{P} \, \mathbf{v}(\mathbf{V}(I,:)\mathbf{u}_N), \quad \mathbf{v}_e = \mathbf{V}(I,:)\mathbf{v}_N, \\
\mathbf{u}_e &= u(\mathbf{v}_e), \\
\mathbf{F}_{ij} &= \mathbf{f}_s(\mathbf{u}_e^{(i)}, \mathbf{u}_e^{(j)}),
\end{aligned}$$

where $v(u)$ is the entropy variable associated with u while $\mathbf{f}_s(a,b)$ is the entropy conservative flow symmetrical between a and b.

Assuming that the hyper-reduced weights $\mathbf{W}$ and the projection $\mathbf{P_t}$ are chosen correctly, the formulation (A.16) preserves the entropy in the form:

$$\mathbf{1}^T \mathbf{W} \frac{dS(\mathbf{V}(I,:)\mathbf{u}_N)}{dt} = 0$$

Using the diagonal structure of W and the properties of the chain rule:

$$\begin{aligned}
{\mathbf{v_N}}^T \mathbf{M_N} \frac{d\mathbf{u}_N}{dt} &= \mathbf{v}(\mathbf{V}(I,:)\,\mathbf{u}_h)^T \, \mathbf{P}^T \, \mathbf{M_N} \frac{d\mathbf{u}_N}{dt} \\
&= \mathbf{v}(\mathbf{V}(I,:)\,\mathbf{u}_h)^T \, \mathbf{W} \, \mathbf{V}(I,:) \, {\mathbf{M_N}}^{-1} \mathbf{M_N} \frac{d\mathbf{u}_N}{dt} \\
&= \mathbf{v}(\mathbf{V}(I,:)\,\mathbf{u}_h)^T \, \mathbf{W} \, \mathbf{V}(I,:) \, \frac{d\mathbf{u}_N}{dt} \\
&= \mathbf{v}(\mathbf{V}(I,:)\,\mathbf{u}_h)^T \, \mathbf{W} \, \frac{d(\mathbf{V}(I,:)\,\mathbf{u}_N)}{dt} \\
&= \mathbf{1}^T \, \mathbf{W} \, \frac{dS(\mathbf{V}(I,:)\,\mathbf{u}_N)}{dt}
\end{aligned}$$

The hyper-reduced formulation (A.16) is obtained starting from the reduced model (A.2), namely:

$$\Delta x \mathbf{V}^T \mathbf{V} \frac{d\mathbf{u}_N}{dt} + 2\mathbf{V}^T (\mathbf{Q} \circ \mathbf{F})\mathbf{1} = 0$$

Given that $\mathbf{M} = \mathbf{V}(I,:)^T\ \mathbf{W}\ \mathbf{V}(I,:)$, if $\mathbf{W} = \Delta x \mathbf{I}$, then the first term becomes:

$$\mathbf{M_N} \frac{d\mathbf{u}_N}{dt}$$

The second term is approximated using a double projection:

$$\mathbf{V}^T(\mathbf{Q} \circ \mathbf{F})\mathbf{1} \approx \mathbf{V}^T \left(\mathbf{V_t}{\mathbf{V_t}}^T \mathbf{Q} \mathbf{V_t}{\mathbf{V_t}}^T \circ \mathbf{F}\right)\mathbf{1} = \mathbf{V}^T \left(\mathbf{V_t}\mathbf{P_t}\hat{\mathbf{Q}}{\mathbf{P_t}}^T{\mathbf{V_t}}^T \circ \mathbf{F}\right)\mathbf{1} \tag{A.17}$$

where $\mathbf{P_t}$ is defined with the discrete scalar product induced by $\mathbf{W}$:

$$\mathbf{P_t} = ({\mathbf{V_t}}^T \mathbf{W} \mathbf{V_t})^{-1} {\mathbf{V_t}}^T \mathbf{W}$$

If $\mathbf{W} = \Delta x\, \mathbf{I}$, then $\mathbf{P_t} = {\mathbf{V_t}}^\top$.

When hyperreduction is introduced, the projection $\mathbf{P_t}$ becomes:

$$\mathbf{P_t} = \left(\mathbf{V_t}(I,:)^T \mathbf{W} \mathbf{V_t}(I,:)\right)^{-1} \mathbf{V_t}(I,:)^T \mathbf{W}$$

which defines the interpolation at the reduced points and allows us to write:

$$\mathbf{Q_t} = {\mathbf{P_t}}^T \hat{\mathbf{Q}} \mathbf{P_t}.$$

Substituting in A.17 we obtain:

$$\mathbf{V}(I,:)^T(\mathbf{Q_t} \circ \mathbf{F})\mathbf{1}$$

from which we obtain the formulation (A.16) [1].

Bibliography

[1] Jesse Chan. Entropy stable reduced order modeling of nonlinear conservation laws. *arXiv:1909.09103*, 2019.

[2] Ami Harten, Peter D. Lax, and Bram van Leer. On upstream differencing and godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25(1):35–61, 1983.

[3] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.

[4] Randall J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. SIAM, 2007.

[5] K. W. Morton and D. F. Mayers. *Numerical Solution of Partial Differential Equations: An Introduction*. Cambridge University Press, 2005.

[6] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical Mathematics*. Springer, 2007.

[7] Sébastien Riffaud. *Reduced-order models: convergence between scientific computing and data for fluid mechanics*. Phd thesis, Université de Bordeaux, 2020. Numerical Analysis [math.NA], ffNNT: 2020BORD0334ff.

[8] Philip L. Roe. Characteristic-based schemes for the euler equations. *Annual Review of Fluid Mechanics*, 18:337–365, 1986.

[9] Eitan Tadmor. The numerical viscosity of entropy stable schemes for systems of conservation laws. i. *Mathematics of Computation*, 49(179):91–103, 1987.

[10] Eitan Tadmor. Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems. *Acta Numerica*, 12:451–512, 2003.

[11] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer, 3rd edition, 2009.

[12] Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*. SIAM, 1997.