Received: 15 June 2017 | Revised: 6 March 2018 | Accepted: 10 March 2018
DOI: 10.1002/fld.4507

RESEARCH ARTICLE

WILEY

Model reduction using L^1-norm minimization as an application to nonlinear hyperbolic problems

R. Abgrall | R. Crisovan

Institut für Mathematik, Universität Zürich, Zürich, Switzerland

Correspondence
R. Abgrall, Institut für Mathematik, Universität Zürich, Winterthurstrasse 190, CH-8057 Zürich, Switzerland.
Email: remi.abgrall@math.uzh.ch

Funding information
Swiss National Foundation (SNF), Grant/Award Number: 200021_153604

MSC Classification: 65M08, 65K99, 76L99

Summary

We are interested in the model reduction techniques for hyperbolic problems, particularly in fluids. This paper, which is a continuation of an earlier paper of Abgrall et al, proposes a dictionary approach coupled with an L^1 minimization approach. We develop the method and analyze it in simplified 1-dimensional cases. We show in this case that error bounds with the full model can be obtained provided that a suitable minimization approach is chosen. The capability of the algorithm is then shown on nonlinear scalar problems, 1-dimensional unsteady fluid problems, and 2-dimensional steady compressible problems. A short discussion on the cost of the method is also included in this paper.

KEYWORDS

L^1 minimization, nonlinear hyperbolic problems, reduced-order models

1 | INTRODUCTION

Model reduction is becoming an essential tool to enable applications requiring either real-time predictions or the evaluation of a large number of partial differential equations (PDEs)–based computational models. The first category encompasses optimal control[1,2] and model predictive control.[3,4] Routine analysis and parametrized studies,[5] design optimization,[6,7] and the quantification of uncertainty[8] are applications pertaining to the second category, to name just a few. In all of these applications, the large dimensionality associated with the discretized PDEs prevents their solution in real time. Model reduction reduces that cost by restricting the solution to a subspace of the solution space. This subspace is usually described by a small number of reduced basis (RB) vectors. In turn, a projection step reduces the dimensionality of the system of discrete equations considered, enabling their fast solution.

While the model reduction of elliptic and parabolic PDEs has been the subject of numerous studies[9,10] and its theory is well understood,[11-17] reducing hyperbolic equations has proved to be much more challenging.[18] More specifically, moving waves and discontinuities, such as shocks, require a large number of basis vectors to accurately approximate these features.[19] This characterizes these problems as ones with large Kolmogorov n-widths.[20] Some model reduction approaches that deal with convection dominated or hyperbolic problems have been also developed, each of them proposing different methods to deal with nonlinearities using approximations of generalized Lax pairs,[21] the solution of Monge-Kantorovich mass transfer problem,[22] the method of freezing,[23] or using the domain partitioning method, followed by an interpolation step.[24] All these prior works are mostly based on the studies and the theory of model order reduction for elliptic and parabolic equations and used an L^2 minimization norm. This norm is not the natural one for evolution problems, which involves discontinuities, being not linked to the concept of weak solutions of hyperbolic conservation laws. In our case, using L^1-norm minimization, we place ourselves exactly in this scenario, obtaining a nonoscillatory behavior of the numerical solution.

In order to reduce the Kolmogorov n-width, approaches based on local bases considering local subspaces can be found in related works.[25-28] The locality can be characterized in parameters,[25] time,[26] or state space.[27] In the present work, an approach based on dictionaries is considered.[29,30] More specifically, solutions corresponding to various time and parameter instances are collected and stored in such a dictionary using a greedy sampling method.[8,10,12] Each solution will then be considered as a RB vector. In turn, localization in time and space can be easily enforced by only considering basis vectors corresponding to restricted subdomains of the time and parameters spaces. In addition to the reduction in number of basis vectors, this paper will demonstrate that a key advantage of a dictionary approach is a better approximation of states having sharp gradients and discontinuities. In particular, it will be demonstrated that avoiding basis truncation such as the one occurring in proper orthogonal decomposition (POD)[31] or nonnegative matrix factorization[32] avoids the Gibbs phenomenon.

In addition to the choice of RB, a key ingredient in projection-based model reduction is the definition of the reduced system of equations. For symmetric systems such as those arising in elliptic and parabolic PDEs, Galerkin projection is the method of choice. For nonsymmetric systems, however, it has been shown that minimizing the L^2-norm of the residual is preferable for stability considerations.[33,34] In the present paper, model reduction based on the minimization of the L^1-norm of the residual is introduced and its advantage is demonstrated in conjunction with a dictionary approach for reducing problems with sharp gradient and shocks. More specifically, the present work demonstrates that combining a dictionary and L^1 minimization promotes sparsity in the choice of basis functions participating in the reduced-order solution and results in more accurate and physical reduced-order solutions.

As we stated in the abstract, this work is a continuation of the work of Abgrall et al.[35] In here, we develop robust error estimators, select the dictionary elements using a greedy sampling algorithm, illustrate the advantage of using L^1-norm minimization also on a 2-dimensional (2D) example, and make use of hyper-reduction ideas, discussing also the cost of the method. These achievements were only mentioned as a future work by Abgrall et al.[35] Moreover, in the current work, we are giving more implementation details of the method, we give solutions to the potential difficulties that might arise when using L^1-norm minimization and, last but not least, in the numerical applications part, we are comparing different L^1 minimization approaches with the L^2-norm minimization, we study the convergence and the sparsity of the method, and we illustrate for different targets the quality of the reconstructed solution based on the number of elements in the dictionary.

This paper is organized as follows. We first discuss the problem of interest. In the following section, an approximation of the solution of nonlinear problems by reduced-order models is presented. In the fourth section, we explain the role of L^1-norm minimization in this problem, then we present in detail the algorithm we have developed and provide an error estimate. In Section 6, we describe the greedy sampling algorithm used to select the elements in the RB. In Section 7, we describe the potential difficulties that can arise using L^1-norm minimization, we give solutions how to fix them, and in the end, we present efficient algorithms for the computation of the L^1-norm minimization, both in the cases of linear and nonlinear residuals. In Section 8, we are discussing the computational cost of the method. The last section provides several numerical examples that illustrate the behavior of our methods on linear and nonlinear problems, both in 1-dimensional (1D) and 2D case. A conclusion follows and we sketch some perspectives.

2 | PROBLEM OF INTEREST

In this work, high-dimensional models (HDM) arising from the space discretization of hyperbolic PDEs are considered. The PDEs of the following type are considered:

$$\begin{cases} \frac{\partial \mathbf{W}}{\partial t} + \mathbf{L}(\mathbf{W};\boldsymbol{\mu}) & = \mathbf{f}(t,\boldsymbol{\mu}), \quad \mathbf{x} \in \Omega, \quad t \in [0,T], \\ \mathbf{B}(\mathbf{W};\boldsymbol{\mu}) & = \mathbf{g}(t,\boldsymbol{\mu}), \quad \mathbf{x} \in \partial\Omega, \quad t \in [0,T], \\ \mathbf{W}(\mathbf{x}, t=0, \boldsymbol{\mu}) & = \mathbf{W}_0(\mathbf{x},\boldsymbol{\mu}), \quad \mathbf{x}d \in \Omega, \end{cases} \tag{1}$$

where $\mathbf{W} = \mathbf{W}(\mathbf{x}, t) \in \mathbb{R}^p$ is the state variable, $t \in [0, T]$ is the time variable, $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ is the space variable ($1 \leq d \leq 3$), and $\partial\Omega$ is the boundary of the domain. $\mathbf{L}$ is a differential operator such as the divergence of a flux and $\mathbf{B}$ a boundary operator; $\mathbf{f}$ and $\mathbf{g}$ are volume and surface forces, respectively; and $\boldsymbol{\mu} = (\mu^1, \ldots, \mu^m) \in \mathcal{P} \subset \mathbb{R}^m$ is a vector of m parameters defining the system of interest.

Discretizing the PDE (1) using finite differences approximation or finite volume formulation leads to a system of large dimension $N = p \times N_{\text{space}}$ of ordinary differential equations of the following form:

$$\begin{cases} \frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} + \mathbf{f}(\mathbf{w}, t; \boldsymbol{\mu}) & = \mathbf{g}(t, \boldsymbol{\mu}), \quad t \in [0, T] \\ \mathbf{w}(t = 0, \boldsymbol{\mu}) & = \mathbf{w}_0(\boldsymbol{\mu}), \end{cases} \tag{2}$$

where $\mathbf{w} = \mathbf{w}(t, \boldsymbol{\mu}) \in \mathbb{R}^N$ is the HDM state, t denotes the time, and $\mathbf{f}(\cdot, \cdot)$, $\mathbf{g}(\cdot)$ are the nonlinear functions of their arguments.

In the remainder of this paper, the time and parameter variables are grouped together, unless explicitly stated, as a variable $\boldsymbol{\tau} = [t; \mu]$. Hence, the HDM state is parametrized as

$$\mathbf{w}(\boldsymbol{\tau}) = \mathbf{w}(t, \boldsymbol{\mu}). \tag{3}$$

In practice, the ordinary differential equation (2) is discretized in time using a time discretization $t_0 = 0 < t_1 < \cdots < t_{N_t} = T$. Explicit and implicit time-discretization techniques are used in the present paper, resulting in a sequence of nonlinear systems of equations of large dimension N

$$\mathbf{r}^n(\mathbf{w}) = \boldsymbol{0}, \quad n = 1, \ldots, N_t, \tag{4}$$

where $\mathbf{r}^n = [r_1^n, \ldots, r_N^n]^T$. We give several examples later in the text. Note that the residual $\mathbf{r}^n$ will depend on several time instances of the solution for unsteady problems, for example, $\mathbf{w}^n$ and $\mathbf{w}^{n-1}$ in the simplest case. Steady problems can also be written in the form $\mathbf{r}(\mathbf{w}) = \mathbf{0}$.

The goal of model reduction is to approximate the high-dimensional system (4) using a much smaller number of variables while retaining the accuracy of the solution. For that purpose, projection-based model reduction techniques approximate the state $\mathbf{w}(\boldsymbol{\tau})$ in a subspace of $\mathbb{R}^N$ using a reduced-order basis (ROB)/dictionary $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_k] \in \mathbb{R}^{N \times k}$, $k << N$. The state is then approximated as

$$\mathbf{w}(\boldsymbol{\tau}) \approx \mathbf{V}\mathbf{q}(\boldsymbol{\tau}) = \sum_{i=1}^{k} \mathbf{v}_i q_i(\boldsymbol{\tau}), \tag{5}$$

where $\mathbf{q}(\boldsymbol{\tau}) = [q_1(\boldsymbol{\tau}), \ldots, q_k(\boldsymbol{\tau})]^T \in \mathbb{R}^k$ denotes the vector of k reduced coordinates. Substituting the subspace approximation (5) into (4) usually results in a nonzero residual of dimension N

$$\mathbf{r}^n(\mathbf{V}\mathbf{q}) \approx \boldsymbol{0}, \tag{6}$$

which accounts for the fact that $\mathbf{V}\mathbf{q}(\boldsymbol{\tau})$ is not in general an exact solution of the dynamical equation. Two common approaches result in the definition of a reduced system of equations.

- Galerkin projection[9,11] enforces the orthogonality of the residual to the ROB $\mathbf{V}$ as
$$\mathbf{V}^T\mathbf{r}^n(\mathbf{V}\mathbf{q}) = \boldsymbol{0}, \quad n = 1, \ldots, N_t. \tag{7}$$
This defines a set of k nonlinear equations in terms of k unknowns, which can be solved by Newton-Raphson's method.
- Residual minimization approaches[6,8,27,33,34] minimize the residual in the L^2-norm sense
$$\min_{\mathbf{q} \in \mathbb{R}^k} \|\mathbf{r}^n(\mathbf{V}\mathbf{q})\|_2^2 = \sum_{i=1}^{N} \left(r_i^n(\mathbf{V}\mathbf{q})\right)^2, \quad n = 1, \ldots, N_t. \tag{8}$$
In practice, this nonlinear least-squares problem can be solved using Gauss-Newton or Levenberg-Marquardt iterations.[36] In Section 7, alternative residual minimization approaches based on L^1-norm minimization, which are more appropriate for the reduction of hyperbolic problems, will be proposed.

3 | DICTIONARY APPROACH

Projection-based model reduction techniques[13,31,32] are based on the precomputed snapshots of the HDM for specific values of the vector $\boldsymbol{\tau} = [t; \mu]$. These snapshots are gathered in a snapshot matrix

$$\mathbf{S} = [\mathbf{w}(\boldsymbol{\tau}_1), \ldots, \mathbf{w}(\boldsymbol{\tau}_{N_s})]. \tag{9}$$

Five approaches for compressing the snapshot matrix are described as follows.

- The POD[31] computes an optimal ROB of dimension k that minimizes the projection error of the snapshots onto the basis.

- Balanced POD,[37] applicable to linear systems only, also takes into account snapshots of the dual system to construct the RB for the primal and dual systems.
- Nonnegative matrix factorization[38] was recently applied to construct a nonnegative ROB based on snapshots with positive entries in the context of contact problems.[32] The RB minimizes the positive reconstruction of the snapshots.
- Greedy sampling method was introduced in related works[10,12,39] and approaches the task of choosing parameter sample points one by one in an adaptive manner.
- The POD-greedy algorithm is a combination of the greedy algorithm with a temporal compression step. The main ingredient for time-sequence compression is the use of POD.[40]

All 5 approaches perform a compression of the information contained in the snapshot matrix **S**. More specifically, the N_s vectors contained in **S** are compressed, leading to a ROB of dimension $k \leq N_s$.

In the present paper, an approach based on a dictionary of solutions is preferred as it does not incur any loss of information by compression. As such, the vectors $\{\mathbf{v}_i\}_{i=1}^{k}$ in the RB are the solutions of the HDM

$$\mathbf{v}_i = \mathbf{w}(\boldsymbol{\tau}_i) = \mathbf{w}(t, \boldsymbol{\mu}_i), \quad i = 1, \ldots, k. \tag{10}$$

These vectors will constitute our dictionary. We denote with $\mathcal{D} = \{\boldsymbol{\mu}_i\}_{i=1}^{k}$ the set of parameters chosen by the greedy algorithm from a big training set $\mathcal{C} = \{\boldsymbol{\mu}_i\}_{i=1}^{N_c}, \quad k < N_c$ at each greedy iteration. Therefore, the number of samplings corresponds to the number of elements in the RB. In this case, the error can be controlled using the error estimate given in Section 5.2. The solution of the HDM will then be approximated as

$$\mathbf{w}(\boldsymbol{\tau}) \approx \sum_{i=1}^{k} \mathbf{w}(\boldsymbol{\tau}_i) q_i(\boldsymbol{\tau}). \tag{11}$$

In the present case, since the HDM is of very large dimension, over-complete dictionaries, as used in compressed sensing[41,42] and for which $k \geq N$, will not be considered.

4 | L^1-NORM RESIDUAL MINIMIZATION

In the present paper, model reduction based on L^1-norm residual minimization is introduced to reduce the dimensionality of hyperbolic equations as an alternative to Galerkin projection and L^2-norm minimization. Motivations for the use of the L^1-norm are provided in this section. Model reduction based on L^1-norm minimization will be introduced in Section 7 together with practical numerical procedure for their computation in Section 7.2.

Minimizing the L^1-norm of the residual is known to lead to regressions that are much more robust to outliers.[43] In the context of hyperbolic systems, Lavery's work[44,45] was probably the first one that used L^1 minimization to solve hyperbolic problems. It was followed by Guermond et al on Hamilton-Jacobi equations and transport problems[46,47] and it has been shown, at least experimentally, that the numerical solution can retain an excellent nonoscillatory behavior by minimizing the L^1-norm of the PDE residual. In the work of Guermond and Popov,[47] the schemes are designed by minimizing quantities that mimic the total variation of a functional, as in here. In the following sections, the idea is exploited in the case of model reduction. For completeness, the motivation for L^1-norm minimization is justified as follows for the problem

$$\frac{\partial \mathbf{W}}{\partial t} + \operatorname{div}\mathbf{F}(\mathbf{W}) = 0 \tag{12}$$

defined on $\Omega \subset \mathbb{R}^d \times \mathbb{R}^+$. The solution $\mathbf{W} = \mathbf{W}(\mathbf{x}, t)$ belongs here to $\mathbb{R}^p$ so that $\mathbf{F} = (F_1, \ldots, F_p)^T$. The weak form of the equation is, for any $\boldsymbol{\varphi} \in \left[C_0^1(\overset{\circ}{\Omega})\right]^p$ with compact support in the interior $\overset{\circ}{\Omega}$ of Ω,*

$$\int_\Omega \boldsymbol{\varphi}(\mathbf{x}, t) \left(\frac{\partial \mathbf{W}}{\partial t} + \operatorname{div}\mathbf{F}(\mathbf{W}) \right) \mathrm{d}t\mathrm{d}\mathbf{x} = 0. \tag{13}$$

Integrating by parts yields

$$\int_\Omega \frac{\partial \boldsymbol{\varphi}}{\partial t} \mathbf{W} \mathrm{d}t\mathrm{d}\mathbf{x} + \int_\Omega \nabla \boldsymbol{\varphi} \cdot \mathbf{F}(\mathbf{W}) \mathrm{d}t\mathrm{d}\mathbf{x} = 0. \tag{14}$$

*So that, in particular, $\forall (x, 0) \in \mathbb{R}^d \times \mathbb{R}^+$, $\phi(x, 0) = 0$.

Restricting to the set of test functions $\mathcal{T} = \left\{ \boldsymbol{\varphi} \in \left[C_0^1(\overset{\circ}{\Omega}) \right]^p, ||\boldsymbol{\varphi}||_\infty \leq 1 \right\}$, **W** is a solution if

$$\sup_{\boldsymbol{\varphi}\in\mathcal{T}} \left(\int_\Omega \frac{\partial \boldsymbol{\varphi}}{\partial t} \mathbf{W} \mathrm{d}t\mathrm{d}\mathbf{x} + \int_\Omega \nabla\boldsymbol{\varphi}\cdot\mathbf{F}(\mathbf{W})\mathrm{d}t\mathrm{d}\mathbf{x}\right) = 0. \tag{15}$$

Remember that, for any function $\mathbf{g} \in L^1(\mathbb{R}^d)$, the total variation is defined as

$$TV(\mathbf{g}) = \sup_{\boldsymbol{\varphi}\in C_0^1(\mathbb{R}^d)\cap L^\infty(\mathbb{R}^d), ||\boldsymbol{\varphi}||_\infty \leq 1} \left\{ \int_{\mathbb{R}^d} \nabla\boldsymbol{\varphi}(\mathbf{x})\cdot\mathbf{g}(\mathbf{x})\mathrm{d}\mathbf{x} \right\}, \tag{16}$$

and if, in addition, $\mathbf{g} \in C^1(\mathbb{R}^d)$, then $TV(\mathbf{g}) = \int_{\mathbb{R}^d} ||\nabla\mathbf{g}||\mathrm{d}x = ||\nabla\mathbf{g}||_{L^1(\mathbb{R}^d)}$. This shows that, defining the space-time flux $\mathcal{F} = (\mathbf{W}, \mathbf{F})$, **W** is a weak solution if and only if the space-time total variation of $\mathcal{F}$ vanishes, that is,

$$TV(\mathcal{F}(\mathbf{W})) = 0. \tag{17}$$

In other words, one can look for **W** as a function of $L^1 \cap L^\infty$ such that **W** minimizes $TV(\mathcal{F}(\mathbf{V}))$ over $\mathbf{V} \in L^1 \cap L^\infty$, ie,

$$\mathbf{W} = \mathrm{argmin}\{TV(\mathcal{F}(\mathbf{V})), \mathbf{V} \in L^1 \cap L^\infty\}. \tag{18}$$

This does not guaranty uniqueness (and thus, there is some abuse of language in this setting) because the entropy conditions are not encoded into this formulation. From (17), this nonuniqueness can also be interpreted in term on the nonstrict convexity of the L^1 unit ball.

However, (18) indicates that a natural setting is to minimize the L^1 norm of the space-time divergence of the space-time flux $\mathcal{F}$, but something must be added to generate unique solutions as the entropy criteria is a way to guaranty the uniqueness of the solution of (12).

How does it translates to the discrete setting? For simplicity, we only mention the case of explicit schemes. We discuss later the solution procedure for the case of implicit schemes. The following classical result is mentioned. Consider $\{x_j\}_{j\in\mathbb{Z}}$ a strictly increasing sequence in $\mathbb{R}$ and $x_{j+1/2} = \frac{x_j + x_{j+1}}{2}$. Assuming that $\mathbb{R} = \cup_{j\in\mathbb{Z}}[x_{j-1/2}, x_{j+1/2}[$ and considering g defined as, for any $j \in \mathbb{Z}$,

$$g(x) = g_j \text{ if } x \in [x_{j-1/2}, x_{j+1/2}[, \tag{19}$$

then

$$TV(g) = \sum_{j\in\mathbb{Z}} |g_{j+1} - g_j|. \tag{20}$$

Consider now an approximation procedure that enables, from $\mathbf{w}^n \approx \mathbf{W}(\cdot, t_n)$, to compute $\mathbf{w}^{n+1} \approx \mathbf{W}(\cdot, t_{n+1})$. For instance, assume that we have a finite volume method $d = 1$ and, for any grid point $j \in \{1, \ldots, N\}$, we define the mesh $x_{j+1/2} = j\Delta x$, $t^n = n\Delta t$, and the control volumes $c_j = (x_{j-1/2}, x_{j+1/2})$. Considering $\boldsymbol{\varphi} = \boldsymbol{\mathit{1}}_{[x_{j-1/2}, x_{j+1/2}]\times[t^n, t^{n+1}]}$ with $[x_{j-1/2}, x_{j+1/2}] \times [t^n, t^{n+1}] \subset \Omega$ in (13), we obtain the following:

$$\begin{aligned}
0 &= \int_{x_{j-1/2}}^{x_{j+1/2}} \int_{t^n}^{t^{n+1}} \left(\frac{\partial \mathbf{W}}{\partial t} + \mathrm{div}\mathbf{F}(\mathbf{W}) \right) \mathrm{d}t\mathrm{d}x \\
&= \Delta x \int_{x_{j-1/2}}^{x_{j+1/2}} \frac{1}{\Delta x} \mathbf{W}(x, t^{n+1})\mathrm{d}x - \Delta x \int_{x_{j-1/2}}^{x_{j+1/2}} \frac{1}{\Delta x} \mathbf{W}(x, t^n)\mathrm{d}x \\
&\quad + \Delta t \int_{t^n}^{t^{n+1}} \frac{1}{\Delta t} \mathbf{F}(\mathbf{W}(x_{j+1/2}, t))\mathrm{d}t - \Delta t \int_{t^n}^{t^{n+1}} \frac{1}{\Delta t} \mathbf{F}(\mathbf{W}(x_{j-1/2}, t))\mathrm{d}t.
\end{aligned}$$

Using the approximations $\mathbf{w}_j^n \approx \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{W}(x, t^n)\mathrm{d}x$ and $\mathbf{f}_{j+1/2}(\mathbf{w}^n) \approx \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{W}(x_{j+1/2}, t))\mathrm{d}t$, we obtain

$$\Delta x(\mathbf{w}_j^{n+1} - \mathbf{w}_j^n) + \Delta t\big(\mathbf{f}_{j+1/2}(\mathbf{w}^n) - \mathbf{f}_{j-1/2}(\mathbf{w}^n)\big) = 0. \tag{21}$$

In this case, the residual can be written as

$$\left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j = \mathbf{w}_j^{n+1} - \mathbf{w}_j^n + \frac{\Delta t}{\Delta x}\left(\mathbf{f}_{j+1/2}(\mathbf{w}^n) - \mathbf{f}_{j-1/2}(\mathbf{w}^n)\right), \tag{22}$$

and we can evaluate the value of $\mathbf{w}^{n+1}$ as the minimization solution of the total variation

$$TV(\boldsymbol{r}) = \sum_{j=1}^{N} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right|. \tag{23}$$

In (22), $\mathbf{f}_{j+1/2}$ is any consistent numerical flux at the cell interface $x_{j+1/2}$; see the work of Toro[48] for the classical examples. Substituting (5) into (22), for all time steps, find the coefficients q_i^n that minimizes the following residual:

$$\boldsymbol{r}_j^n(\boldsymbol{Vq}) = \sum_{i=1}^{k} q_i^{n+1} \mathbf{w}_j^{n+1}(\boldsymbol{\mu}_i) - \sum_{i=1}^{k} q_i^n \mathbf{w}_j^n(\boldsymbol{\mu}_i) + \frac{\Delta t}{\Delta x}\left(\mathbf{f}_{i+1/2}\left(\sum_{i=1}^{k} q_i^n \mathbf{w}_j^n(\boldsymbol{\mu}_i)\right) - \mathbf{f}_{i-1/2}\left(\sum_{i=1}^{k} q_i^n \mathbf{w}_j^n(\boldsymbol{\mu}_i)\right)\right). \tag{24}$$

For the case of implicit schemes, the solution is obtained similarly as for the explicit case, obtaining the following residual that has to be minimized:

$$\boldsymbol{r}_j^n(\boldsymbol{Vq}) = \sum_{i=1}^{k} q_i^{n+1} \mathbf{w}_j^{n+1}(\boldsymbol{\mu}_i) - \sum_{i=1}^{k} q_i^n \mathbf{w}_j^n(\boldsymbol{\mu}_i) + \frac{\Delta t}{\Delta x}\left(\mathbf{f}_{i+1/2}\left(\sum_{i=1}^{k} q_i^{n+1} \mathbf{w}_j^{n+1}(\boldsymbol{\mu}_i)\right) - \mathbf{f}_{i-1/2}\left(\sum_{i=1}^{k} q_i^{n+1} \mathbf{w}_j^{n+1}(\boldsymbol{\mu}_i)\right)\right). \tag{25}$$

Remark 1. The L^1 norm is convex but not strictly convex, and hence, the minimization problem (23) may not have a unique solution. For this reason, in practice, we perturb the functional (23) to make it strictly convex. Let us denote it by J, and thus, we will look for solutions that minimize

$$J(\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})). \tag{26}$$

Examples are as follows.

1. For $\nu > 0$,

$$J(\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})) = \sum_{j=1}^{N} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right| + \nu \sum_{j=1}^{N} (\mathbf{w}_j^{n+1})^2. \tag{27a}$$

2. More generally, if U is a convex entropy,

$$J(\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})) = \sum_{j=1}^{N} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right| + \nu \sum_{j=1}^{N} U(\mathbf{w}_j^{n+1}). \tag{27b}$$

The functional (27) can be used for systems and the choice of $\nu > 0$ will be discussed in Section 7.1.

Another example of strictly convex functionals, probably more linked to the characterisation of entropy solution, is, for $\nu > 0$,

$$J(\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})) = \sum_{j=1}^{N} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right| + \nu \sum_{j=1}^{N} (\mathbf{w}_j^{n+1} - \mathbf{w}_j^n)^2.$$

Remark 2. Other potential difficulties using the minimization of a L^1-norm will be presented in Section 7.

5 | ERROR ESTIMATION

In this section, we provide an error estimate (in the scalar case) between the solution obtained by projecting onto the span of the dictionary or onto the convex hull of the dictionary and the solution of the original scheme. These error estimates are another way to justify the method and are provided in a simple setting, ie, we consider a monotone scheme. In this section, we first precise the scheme settings, then we give a natural condition on the dictionary for obtaining these error estimates and, in the end, we state them and prove them.

5.1 | Scheme setting

Consider the scalar conservation law equations with the initial condition

$$\begin{aligned} \frac{\partial w}{\partial t} + \frac{\partial f(w)}{\partial x} &= 0, \quad x \in \mathbb{R}, t > 0 \\ w(x,0) &= w_0(x), \quad x \in \mathbb{R}. \end{aligned} \tag{28}$$

After discretizing, we assume that the scheme writes, for $w := (w_j)_{j\in\mathbb{Z}}$,

$$w_j^{n+1} = S(w_j^n, \lambda), \tag{29}$$

with $\lambda = \Delta t/\Delta x$ and the initial condition

$$w_j^0 = \text{given}. \tag{30}$$

We assume that the operator S is monotone for $\lambda \in [0, b[$, $b > 0$, ie, if for any sequence w and v bounded for the L^1 or L^∞ norms with $j \in \mathbb{Z}$, $w_j \leq v_j$, then $S(w_j, \lambda) \leq S(v_j, \lambda)$. Let L^1 and L^∞ norms be generically denoted by $|| \cdot ||$.

An example is given by the explicit scheme

$$S(w_j^n, \lambda) = w_j^n - \lambda\left(\hat{f}(w_{j+1}^n, w_j^n) - \hat{f}(w_j^n, w_{j-1}^n)\right), \tag{31}$$

where we assume that the numerical flux $\hat{f}(a, b)$ is monotone, ie, increasing with respect to the first variable and decreasing with respect to the second one and the operator S is monotone under a CFL-like condition.

Another example is given by the implicit scheme, where w_j^n is defined as the solution of

$$S(w_j^{n+1}, \lambda) = w_j^{n+1} + \lambda\left(\hat{f}(w_{j+1}^{n+1}, w_j^{n+1}) - \hat{f}(w_j^{n+1}, w_{j-1}^{n+1})\right), \tag{32}$$

which is unconditionally monotone.

Thanks to Crandall-Tartar lemma (for example, see the work of Godlewski and Raviart[49]), we know that, for any w and v,

$$||S(w, \lambda) - S(v, \lambda)||_{L_1} \leq ||w - v||_{L_1}.$$

The same is true in the L^∞ and L^2 norms.

5.2 | Error estimate

We collect and store into the dictionary $\mathbf{V}$ the solutions $\{w^n(\boldsymbol{\mu}_i)\}_i$ of the problem (28), which correspond to various time and parameter instances and where also the initial conditions are depending on these parameters $\{\boldsymbol{\mu}_i\}_{i=1,\dots,k} \in \mathcal{D}$. Since the minimization procedure admits a unique solution, this enables to define a projection operator p^n for any time t^n by solving the following minimization problem. Knowing $w_\star^n \in \underset{\boldsymbol{\mu}_i\in\mathcal{D}}{\text{span}}(\{w^n(\boldsymbol{\mu}_i)\})$, find $w_\star^{n+1} \in \underset{\boldsymbol{\mu}_i\in\mathcal{D}}{\text{span}}(\{w^{n+1}(\boldsymbol{\mu}_i)\})$ such that, for any target $\boldsymbol{\mu} \in \mathcal{P}$,

$$w_\star^{n+1}(\boldsymbol{\mu}) := \underset{\substack{v_\star \in \text{span}(\{w^{n+1}(\boldsymbol{\mu}_i)\}) \\ \boldsymbol{\mu}_i\in\mathcal{D}}}{\text{argmin}} \left\{J\left(v_\star - S(w_\star^n(\boldsymbol{\mu}), \lambda)\right)\right\} = p^n\left(S(w_\star^n(\boldsymbol{\mu}), \lambda)\right),$$

with J strictly convex. We consider in the following estimates that the scheme S is explicit. The same can be proven also for an implicit scheme.

Remembering that (29) applies for the elements of the dictionary, we have immediately the following estimate:

$$\begin{aligned} J\left(w_\star^{n+1}(\boldsymbol{\mu}) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) &= J\left(p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda)) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \\ &= \min_{\substack{v_\star \in \operatorname{span}(\{w^{n+1}(\mu_i)\}) \\ \mu_i \in \mathcal{D}}} J\left(v_\star - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \\ &\leq \min_{\mu_i\in\mathcal{D}} J\left(w^{n+1}(\boldsymbol{\mu}_i) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \\ &= \min_{\mu_i\in\mathcal{D}} J\left(S(w^n(\boldsymbol{\mu}_i),\lambda) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \\ &\leq \min_{\mu_i\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})\right), \end{aligned} \tag{33}$$

provided λ enables to fulfill the monotonicity property for all the elements of the dictionary. In (33), the passage from lines 2 to 3 simply comes from the fact that we are minimizing on a smaller subset of $\operatorname{span}_{\mu_i\in\mathcal{D}}(\{w^{n+1}(\boldsymbol{\mu}_i)\})$, namely, the element dictionary. The last inequality in (33) relies on (29). It is possible because of the monotonicity of the scheme for the functionals defined in Remark 1 since a monotone scheme is L^1 stable, L^2 stable, and TV stable.[49]

Next, we consider the case where we project on the convex envelop of the dictionary, ie,

$$\min_{\mathbf{q}\in\mathbb{R}^k} \|\mathbf{r}^n(\mathbf{Vq})\|_1 \quad \text{subject to} \quad \mathbf{1}^T q = 1, q \geqslant 0, \quad n = 1, \dots, N_t. \tag{34}$$

Therefore, the projection is a convex combination of the elements in the dictionary and, for any target $\boldsymbol{\mu} \in \mathcal{P}$, we define

$$w_\star^{n+1}(\boldsymbol{\mu}) = p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda)) = \sum_{i=1}^k \alpha_i^{n+1} w^{n+1}(\boldsymbol{\mu}_i).$$

with $\alpha_i^n \geq 0$ and $\sum_{i=1}^k \alpha_i^n = 1, \forall n$. We obtain a sharper error estimate of type

$$\begin{aligned} J\left(w_\star^{n+1}(\boldsymbol{\mu}) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) &\leq \min_{\mu_i\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})\right) \\ &= \min_{\mu_i\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - \sum_{j=1}^k \alpha_j^n w^n(\boldsymbol{\mu}_j)\right) \\ &\leq \min_{\mu_i\in\mathcal{D}} \left(\sum_j^k |\alpha_j^n|\right) \max_{\mu_j\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w^n(\boldsymbol{\mu}_j)\right) \\ &= \min_{\mu_i\in\mathcal{D}} \max_{\mu_j\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w^n(\boldsymbol{\mu}_j)\right) \\ &\leq \min_{\mu_i\in\mathcal{D}} \max_{\mu_j\in\mathcal{D}} J\left(w^0(\boldsymbol{\mu}_i) - w^0(\boldsymbol{\mu}_j)\right) =: \mathcal{J}(\mathcal{D}). \end{aligned} \tag{35}$$

Again, the last inequality in (35) is possible due to the monotonicity of the scheme for the functionals defined in Remark 1 because a monotone scheme is L^1 stable, L^2 stable, and TV stable.[49] We have shown the following result.

Proposition 1. *Consider J defined as in Remark 1. If $S(\cdot,\lambda)$ is a monotone scheme for $\lambda \in [0,b[$, $b > 0$, then we have the following.*

1. *At time t_{n+1} , the minimization is done onto the span of the dictionary $\{w^n(\boldsymbol{\mu}_i), \boldsymbol{\mu}_i \in \mathcal{D}\}$, for any $\boldsymbol{\mu} \in \mathcal{P}$, the reduced solution $w_\star^{n+1}(\boldsymbol{\mu}) = p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda))$ satisfies*

$$J\left(w_\star^{n+1}(\boldsymbol{\mu}) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \leq \min_{\mu_i\in\mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})\right).$$

2. *At time t_{n+1}, the minimization is done on the convex hull of the dictionary, for any $\boldsymbol{\mu} \in \mathcal{P}$, the reduced solution $w_\star^{n+1}(\boldsymbol{\mu}) = p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda))$ satisfies*

$$J\left(w_\star^{n+1}(\boldsymbol{\mu}) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \leq \min_{\mu_i\in\mathcal{D}} \max_{\mu_j\in\mathcal{D}} J\left(w^0(\boldsymbol{\mu}_i) - w^0(\boldsymbol{\mu}_j)\right).$$

For J defined as in (27a) and using a projection over the span of the dictionary, we get the following estimate:

$$\begin{aligned}
J\left(p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda)) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) &= \min_{\substack{v_\star \in \operatorname{span}(\{w^{n+1}(\boldsymbol{\mu}_i)\}) \\ \boldsymbol{\mu}_i \in \mathcal{D}}} J\left(v_\star - S(w^n(\boldsymbol{\mu}),\lambda)\right) \\
&\leq \min_{\boldsymbol{\mu}_i \in \mathcal{D}} J\left(w^{n+1}(\boldsymbol{\mu}_i) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \\
&= \min_{\boldsymbol{\mu}_i \in \mathcal{D}} J\left(S(w^n(\boldsymbol{\mu}_i),\lambda)\right) - S(w_\star^n(\boldsymbol{\mu}),\lambda)) \\
&\leq \min_{\boldsymbol{\mu}_i \in \mathcal{D}} J\left(w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})\right) \\
&= \min_{\boldsymbol{\mu}_i \in \mathcal{D}} \left(||w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})||_1 + \nu||w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})||_2^2\right) \\
&\leq \min_{\boldsymbol{\mu}_i \in \mathcal{D}} \left[\left(1 + \nu||w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})||_\infty\right) ||w^n(\boldsymbol{\mu}_i) - w_\star^n(\boldsymbol{\mu})||_1\right],
\end{aligned}$$

where the last inequality holds due to the L^p norms inequality $||f||_q \leq ||f||_r^{r/q}||f||_\infty^{1-r/q}$, with $q = 2$ and $r = 1$.

Using the same technique as for the proof of Proposition 1, we obtain in the case of the convex hull projection the following estimate:

$$J\left(p^n(S(w_\star^n(\boldsymbol{\mu}),\lambda)) - S(w_\star^n(\boldsymbol{\mu}),\lambda)\right) \leq \min_{\boldsymbol{\mu}_i \in \mathcal{D}} \max_{\boldsymbol{\mu}_j \in \mathcal{D}} \left[\left(1 + \nu||w^n(\boldsymbol{\mu}_i) - w^n(\boldsymbol{\mu}_j)||_\infty\right) ||w^n(\boldsymbol{\mu}_i) - w^n(\boldsymbol{\mu}_j)||_1\right].$$

6 | TRAINING BY GREEDY SAMPLING

An essential step in the construction of a parametric ROM is the selection of the sampled snapshots in the time and parameter domains. Greedy approaches[8,10,12] proceed by iteratively selecting the location in the parameter space where the error between the HDM and the ROM is the largest. Then, solve the full model for the chosen sampling and update the reduced model. These steps are repeated until the error is acceptable. As computing the error requires the expensive solution of the HDM, cheaper error indicators are used instead. In the present work, the cumulative L^1-norm of the residual vector corresponding to the ROM solution is used as error indicator

$$\mathcal{E}(\boldsymbol{\mu}) = \frac{1}{N_t}\sum_{n=1}^{N_t}||\mathbf{r}^n(\mathbf{V}\mathbf{q}^n(\boldsymbol{\mu}))||_1, \tag{36}$$

where $\mathbf{q}^n(\mu)$ denotes the reduced coordinates for the parameter μ at time iteration t^n. The greedy procedure is recalled in Algorithm 1.

Algorithm 1 Greedy sampling of the parameter space

Input: Residual function $\mathbf{r}(\cdot)$, tolerance for convergence ϵ, candidate parameter set $\mathcal{C} = \{\boldsymbol{\mu}_i\}_{i=1}^{N_c}$
Output: Dictionary $\mathbf{V}$

```
Randomly chose an initial sample parameter $\boldsymbol{\mu}_0 \in \mathcal{C}$ and compute the associated HDM solution $\{\mathbf{w}^n(\boldsymbol{\mu}_0)\}_{n=1}^{N_t}$
Construct an initial dictionary $\mathbf{V} = \{\mathbf{w}^n(\boldsymbol{\mu}_0)\}_{n=1}^{N_t}$
for $i_c = 1, \dots, N_c$ do
    Solve for the ROM solution $\{\mathbf{q}^n(\boldsymbol{\mu}_{i_c})\}_{n=1}^{N_t}$ and evaluate the error indicator $\mathcal{E}(\boldsymbol{\mu}_{i_c})$
end for
$j = 1$
while $\max_{i_c=1,\dots,N_c} \mathcal{E}(\boldsymbol{\mu}_{i_c}) > \epsilon$ do
    Select $\boldsymbol{\mu}_j = \arg\max_{i_c=1,\dots,N_c} \mathcal{E}(\boldsymbol{\mu}_{i_c})$
    Compute the associated HDM solution $\{\mathbf{w}^n(\boldsymbol{\mu}_j)\}_{n=1}^{N_t}$
    Update the dictionary $\mathbf{V} = \mathbf{V} \bigcup \{\mathbf{w}^n(\boldsymbol{\mu}_j)\}_{n=1}^{N_t}$
    for $i_c = 1, \dots, N_c$ do
        Solve for the ROM solution $\{\mathbf{q}^n(\boldsymbol{\mu}_{i_c})\}_{n=1}^{N_t}$ and evaluate the error indicator $\mathcal{E}(\boldsymbol{\mu}_{i_c})$
    end for
    $j = j + 1$
end while
```

Remark 3. In the case of a monotone scheme and convex hull minimization, the result of Proposition 1 shows that the error indicator (36) can be bounded by $\mathcal{J}(\mathcal{D})$. An intuitive idea of this bound can be interpreted as in the work of Willcox and Peraire[37] and can also be observed from the numerical experiments in Section 9, where the chosen greedy parameters in the dictionary are maximally separated.

7 | POTENTIAL DIFFICULTIES USING L^1-NORM AND ALGORITHMS

In this section, we are firstly illustrating the potential difficulties that one can encounter when using L^1-norm minimization and how to fix them. In the second part of this section, model reduction based on minimizing the residual in the L^1-norm is combined with the dictionary approach presented in Section 3, which leads to different algorithms used to solve the minimization problem.

7.1 | Potential difficulties and procedures

As an alternative to the Galerkin projection and residual minimization in the least-squares sense, a reduced system of equation is here obtained by minimizing, at each time step $n = 1, \dots, N_t$, the L^1-norm of the residual vector as

$$\min_{\mathbf{q}\in\mathbb{R}^k} \|\mathbf{r}^n(\mathbf{Vq})\|_1 = \sum_{i=1}^{n} \left|r_i^n(\mathbf{Vq})\right|, \quad n = 1, \dots, N_t, \tag{37}$$

or the projection on the convex envelop of the dictionary defined in (34). There are at least three difficulties associated with minimizing the L^1-norm. The first one is because of the L^1 nondifferentiability at zero. To circumvent this issue, the Huber function,[50] defined as follows, can be introduced:

$$\phi_M(x) = \begin{cases} x^2, & \text{if} \quad |x| \le M \\ M(2|x| - M) & \text{otherwise.} \end{cases} \tag{38}$$

Then, the sequence of the reduced systems of equations based on the Huber function is

$$\min_{\mathbf{q}\in\mathbb{R}^k} \sum_{i=1}^{n} \phi_M\left(r_i^n(\mathbf{Vq})\right), \quad n = 1, \dots, N_t. \tag{39}$$

The Huber function ϕ_M behaves as a parabola close to $x = 0$ and as the L^1-norm for large values of x. It is continuously differentiable on $\mathbb{R}$ ($\phi_M \in C^1(\mathbb{R})$). It is also used in regressions as a loss function because of its nonsensitivity to outliers. In the present work, it will be used as a continuously differentiable alternative to the L^1-norm.

Figure 1 compares, in the scalar case, the L^2 and L^1-norms to the norm based on the Huber function for the particular case $M = 1$. Practical algorithm for solving the systems of Equations (37) and (39), both in the case of linear and nonlinear residual functions, are presented in the following section.

A second difficulty is that the L^1 norm is not strictly convex so that the uniqueness is not guaranteed. This difficulty is taken into account in the solution procedure by adding a strictly convex penalization term, for example, a L^2 constraint (as in (27a) and (27b)).

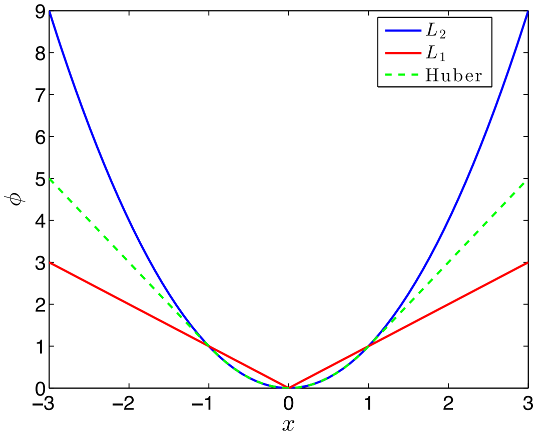


FIGURE 1 Comparison of the L^2, L^1, and the Huber function ($M = 1$) norms [Colour figure can be viewed at wileyonlinelibrary.com]

A third potential issue using a dictionary, as opposed to a RB, is the fact that the dictionary may be rank deficient. One option to address this issue is to perform a Gramm-Schmidt orthogonalization or a rank-revealing QR factorization. A drawback of that approach is that dictionary members are then linearly combined. Alternatively, a regularization term and a random perturbation are here added to the minimization functionals to ensure a system with full rank and a unique solution as follows.

- For L^1-norm minimization, the functional becomes

$$\min_{\mathbf{q}\in\mathbb{R}^k} \|\mathbf{r}^n(\mathbf{V}\mathbf{q})\|_1 + \nu\|\mathbf{q}\|_2^2 = \min_{\mathbf{q}} \sum_{i=1}^{N} \left|r_i^n(\mathbf{V}\mathbf{q})\right| + \nu\sum_{j=1}^{k} q_j^2, \quad n = 1, \dots, N_t, \quad \nu > 0. \tag{40}$$

- For Huber function minimization, the functional becomes

$$\min_{\mathbf{q}\in\mathbb{R}^k} \sum_{i=1}^{N} \phi_M\left(r_i^n(\mathbf{V}\mathbf{q})\right) + \nu\|\mathbf{q}\|_2^2, \quad n = 1, \dots, N_t, \quad \nu > 0. \tag{41}$$

Remark 4. In numerical examples, we are using $\nu = 10^{-6}$ for L^1-norm minimization and $\nu = 10^{-8}$ for the Huber function minimization. These values have been found to be robust throughout applications.

7.2 | Algorithms

A classical solution to minimizing a linear residual vector in the L^1-norm is by recasting the problem as a linear program (LP). More specifically, assuming that the residual is linear $\mathbf{r}^n(\mathbf{V}\mathbf{q}) = \mathbf{A}^n\mathbf{V}\mathbf{q} + \mathbf{b}^n$ with $\mathbf{A}^n \in \mathbb{R}^{N\times N}$ and $\mathbf{b}^n \in \mathbb{R}^N$, a solution to (37) is given by the solution $\mathbf{q} \in \mathbb{R}^k$ of the LP

$$\begin{aligned} \min_{\mathbf{q},\mathbf{s},\mathbf{t}} \quad & \mathbf{1}^T(\mathbf{s}+\mathbf{t}) \\ \text{s.t.} \quad \mathbf{A}^n\mathbf{V}\mathbf{q} + \mathbf{b}^n - \mathbf{s} + \mathbf{t} &= \mathbf{0} \\ \mathbf{s} &\geq \mathbf{0} \\ \mathbf{t} &\geq \mathbf{0}. \end{aligned} \tag{42}$$

Unfortunately, this LP involves $k + 2N$ variables and $3N$ constraints, including N equality constraints, rendering this approach intractable in the case of model reduction.

Alternatively, the L^1-norm minimization problem can be solved by iteratively reweighted least squares (IRLS).[51] This approach proceeds iteratively by solving a sequence of weighted least-squares problem. An advantage of this approach is that its implementation can rely entirely on existing least-squares solvers. Furthermore, its complexity is similar to that of the L^2-norm minimization problem. The procedure is presented in Algorithm 2 in the case of a nonlinear residual vector. At each iteration l, a weighted least-squares problem is solved, where the weight depend on the current value of the residual vector $\mathbf{r}^l$ as follows: $\mathbf{\Theta}^l = \mathbf{diag}\left(|r_i^l|^{-\frac{1}{2}}\right)$.

Algorithm 2 L^1-norm minimization by Iteratively Reweighted Least-Squares

Input: Residual function $\mathbf{r}(\cdot)$ and associated Jacobian $\mathbf{J}(\cdot)$, reduced basis $\mathbf{V}$, initial guess $\mathbf{q}^0$, tolerance for convergence ϵ
Output: Solution $\mathbf{q}$

```
$l = 0$
Compute $\mathbf{r}^0 = \mathbf{r}(\mathbf{V}\mathbf{q}^0)$ and $\mathbf{Z}^0 = \mathbf{J}(\mathbf{V}\mathbf{q}^0)\mathbf{V}$
while $l = 0$ OR $\|\Delta\mathbf{q}^{l-1}\|_1 > \epsilon(1 + \|\mathbf{q}^{l-1}\|_1)$ do
    Compute the weights $\Theta^l = \text{diag}\left(|r_i^l|^{-\frac{1}{2}}\right)$
    Solve the weighted least-squares problem
        $\Delta\mathbf{q}^l = \arg\min_{\mathbf{y}} \|\Theta^l\mathbf{Z}^l\mathbf{y} + \Theta^l\mathbf{r}^l\|_2^2$
    $\mathbf{q}^{l+1} = \mathbf{q}^l + \Delta\mathbf{q}^l$
    Compute $\mathbf{r}^{l+1} = \mathbf{r}(\mathbf{V}\mathbf{q}^{l+1})$ and $\mathbf{Z}^{l+1} = \mathbf{J}(\mathbf{V}\mathbf{q}^{l+1})\mathbf{V}$
    $l = l + 1$
end while
$\mathbf{q} = \mathbf{q}^l$
```

Similarly, the minimization of the Huber function can also be done by an IRLS procedure, as described in Algorithm 3. The procedure only differs from its L^1-norm counterpart by the choice of weights. In the present work, the following choice of weights is proposed for a given residual vector $\mathbf{r}^l$:

$$\Theta^l = \text{diag}(\Theta_i^l), \tag{43}$$

where

$$\Theta_i^l = \begin{cases} l1, \text{ if } |r_i^l| < M \\ \frac{M}{\sqrt{|r_i^l|}} \quad \text{else}, \end{cases} \tag{44}$$

and setting

$$M = \varepsilon_2 \max\left(1, \max_i \left(|r_i^l|\right)\right). \tag{45}$$

The value $\epsilon_2 = 10^{-6}$ has been found to be a robust choice across different applications.

Algorithm 3 Huber function minimization by Iteratively Reweighted Least-Squares

```
Input: Residual function $\mathbf{r}(\cdot)$ and associated Jacobian $\mathbf{J}(\cdot)$, reduced basis $\mathbf{V}$, initial guess $\mathbf{q}^0$, tolerance for convergence $\epsilon$
Output: Solution $\mathbf{q}$
$l = 0$
Compute $\mathbf{r}^0 = \mathbf{r}(\mathbf{V}\mathbf{q}^0)$ and $\mathbf{Z}^0 = \mathbf{J}(\mathbf{V}\mathbf{q}^0)\mathbf{V}$
while $l = 0$ OR $\|\Delta\mathbf{q}^{l-1}|_1 > \epsilon(1 + \|\mathbf{q}^{l-1}\|_1)$ do
    Compute the weights $\boldsymbol{\Theta}^l = \textbf{diag}\left((|r_i^l| < M) + M|r_i^l|^{-\frac{1}{2}}(|r_i^l| \geq M)\right)$
    Let $M = \epsilon_2 \max(1, \max(|r_i^l|))$
    Solve the weighted least-squares problem
        $\Delta\mathbf{q}^l = \arg\min_{\mathbf{y}} \|\boldsymbol{\Theta}^l\mathbf{Z}^l\mathbf{y} + \boldsymbol{\Theta}^l\mathbf{r}^l\|_2^2$
    $\mathbf{q}^{l+1} = \mathbf{q}^l + \Delta\mathbf{q}^l$
    Compute $\mathbf{r}^{l+1} = \mathbf{r}(\mathbf{V}\mathbf{q}^{l+1})$ and $\mathbf{Z}^{l+1} = \mathbf{J}(\mathbf{V}\mathbf{q}^{l+1})\mathbf{V}$
    $l = l + 1$
end while
$\mathbf{q} = \mathbf{q}^l$
```

8 | COMPUTATIONAL COST

Let us make some comments on the computational cost of the method.

The minimization procedure consists in looking for the minimum of functionals of the type (28), where all the degrees of freedom describing the solution appear. This is a challenge because, in general, the algorithms to solve this kind of minimization procedure are much more expensive than those of the least-square type. In all the numerical experiments that will be presented in Section 9, we consider a slightly different approach, namely, instead of using all the degrees of freedoms, we use only a small subset of them $\mathcal{I}$. Hence, instead of minimizing the total variation in (23), one can minimize the total variation over $\mathcal{I}$, ie,

$$TV(\boldsymbol{r}) = \sum_{j \in \mathcal{I}} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right|. \tag{46}$$

In practice, because of the uniqueness issues listed earlier, instead of the functional (27b), we minimize

$$J(\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})) = \sum_{j \in \mathcal{I}} \left| \left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_j \right| + \nu \sum_{j \in \mathcal{I}} \left(\mathbf{w}_j^{n+1}\right)^2, \tag{47}$$

where $\mathcal{I}$ is a small subset of the set of degrees of freedom defined as in (46). Of course, the question is how to choose this set. Clearly, if $\mathcal{I}$ is equal to the full set of grid points, the solution is given by (23). We discuss how to choose $\mathcal{I}$ later in this paragraph and focus first on the evaluation of $\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})$. We describe this in the steady 2D case, the extension to the unsteady case or the 1D case is quite straightforward. In our simulation, we have chosen to use the scheme developed in

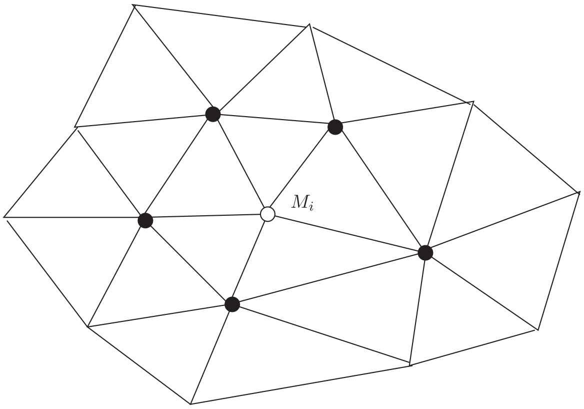


FIGURE 2 Representation of the set $\mathcal{V}_i$. The vertex M_i is indicated with a ○ and the elements of $\mathcal{V}_i$ by ●

related works,[52-54] but this is not essential and what matters is that the stencil of the method is relatively compact because we are integrating a PDE of the form (12)

8.1 | Evaluation of $\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})$

The computational domain is covered by an unstructured mesh, which elements K are triangles in 2D. The same method could deal with an hybrid mesh. Consider a vertex M_i in the computational mesh. We denote by $\mathcal{V}_i$ the set of vertices that are connected to M_i by an edge, and by $\mathcal{W}_i$ the set of points that are needed to evaluate $\left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_i$, see Figure 2. In the case of the scheme described in the following, the 2 sets coincide, and we also have that the set $\mathcal{W} = \mathcal{V}_i \cup \{M_i\}$ is the set of vertices contained in $\cup_{K, M_i \in K} K$. The full order model writes

$$\left[\boldsymbol{r}(\mathbf{w}^n, \mathbf{w}^{n+1})\right]_i = |C_i| \frac{\mathbf{w}_i^{n+1} - \mathbf{w}_i^n}{\Delta t} + \sum_{K, M_i \in K} \Phi(\mathbf{w}_i^n, \mathbf{w}_j^n, \mathbf{w}_k^n).$$

Here, the vertices of K are M_i, M_j, and M_k; and $|C_i|$ is the area of the dual control volume. Though the stencil of the method is reduced to $\mathcal{V}_i \cup \{M_i\}$, the method is second order in space.

In the case of the reduced-order model, we proceed as follows. Knowing $\mathbf{q}^n$, we evaluate the components of $V\mathbf{q}^n$ corresponding to the vertices of $\{M_i, i \in \mathcal{I}\} \cup \cup_{i \in \mathcal{I}} \mathcal{V}_i$. For a regular mesh, this amounts to $\approx 7|\mathcal{I}|$ evaluations. Then, we evaluate $\mathbf{q}^{n+1}$ by the minimization of J defined in (47).

8.2 | Evaluation of $\mathcal{I}$

In the dictionary approach, a lot of information is encoded and the degrees of freedom are linked if they are in the same cone of dependence. In the fluid dynamics case and for a subsonic solution, the problem to solve is essentially elliptic; hence, 2 different degrees of freedom are linked together, ie, if one consider the grid points M_1 and M_2, the flow field at M_1 has some knowledge of what occurs at M_2. In the case of a transonic flow, the subsonic and supersonic pockets are somehow disconnected. Using this rational, one can select randomly points in the mesh, taking into account an a priori knowledge of the location of the supersonic pockets when they exist.

This hyper-reduction strategy is applied in all our numerical results. In 2D, we use only about 100 points in each case. Several stencils will be chosen and the presented results seem not to be sensitive to the choices. In addition, in the 1D case where the full grid can be used, we have not experienced any change in the solution.

9 | NUMERICAL APPLICATIONS

In the next numerical applications, we consider the parameter space $\mathcal{P}$ to be included in $\mathbb{R}^m$, with $m = 1$. As a consequence, we denote the parameters in this section by μ.

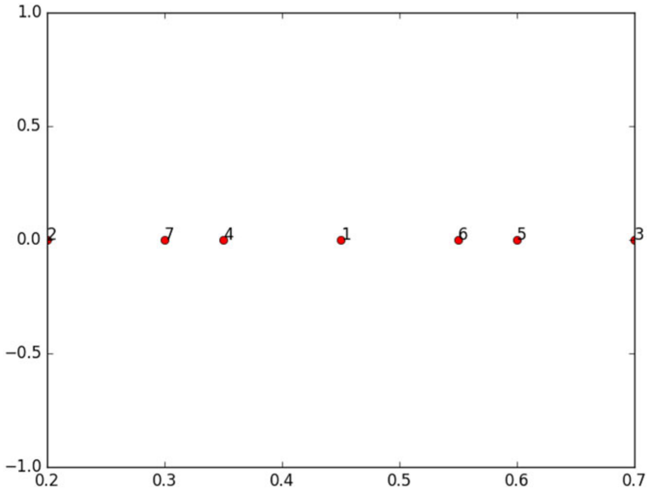


FIGURE 3 Parameters contained in $\mathcal{D}$, which are selected by the greedy algorithm for Burgers' equation [Colour figure can be viewed at wileyonlinelibrary.com]

9.1 | Unsteady Burgers′ equation

We consider here the system

$$\frac{\partial w}{\partial t} + \operatorname{div} f(w) = 0 \tag{48}$$

defined on $\Omega = [0, \pi] \subset \mathbb{R}$, $t > 0$ with periodic boundary conditions, ie, $f(u) = \frac{w^2}{2}$. The initial conditions are parametrized by

$$w_0(x; \mu) = \mu \ \ |\sin(2\,x)| + 0.1, \qquad \mu \in [0.2, 0.7].$$

This initial condition is chosen such that a moving shock is generated in a finite time for $\mu \neq 0$. The shock moves with velocity is $\sigma_\mu = 0.5\mu + 0.1$. The PDE is discretized by an upwind first-order scheme using a uniform mesh, resulting in an HDM of dimension $N = 10^2$, the CFL condition is 0.5, the number of iterations is $N_t = 200$, and the time step equals 0.0157.

The greedy sampling algorithm proposed in the Section 6 is firstly applied in an offline stage to construct a set of parameters $\mathcal{D}$, which is accurate in the parametric domain $\mathcal{P} = [0.2, 0.7]$. For that purpose, a set C containing $N_c = 11$ random training parameters is considered. We start with an initial element in the dictionary and then 6 greedy iterations are performed using a reduced-order method based on L^1-norm minimization by LP and with the error indicator (36), resulting in a dictionary $\mathbf{V}$ with $k = 7$ members.

The values of $\mathcal{D}$ that represent the parameters selected by the greedy approach are reported in Figure 3 and the dictionary members of $\mathbf{V}$ are depicted in Figure 4. One can observe that the greedy algorithm selects in practice new samples that are maximally separated from the previously sampled dictionary members.

For the online stage, 2 target parameters $\mu_1 = 0.575$ and $\mu_2 = 0.65$ are randomly selected and the dictionary approach based on the previously constructed 7 sampled is tested together with the following 4 model reduction approaches:

1. minimization of the L^2-norm of the residual;
2. minimization of the L^1-norm of the residual by linear programming;
3. minimization of the L^1-norm of the residual by IRLS with tolerance $\epsilon = 10^{-8}$; and
4. minimization of the Huber function applied to the residual by IRLS with tolerance $\epsilon = 10^{-8}$.

The solutions obtained using each model-order reduction approach are compared with the target solution in Figures 5 and 6 at time $t = \pi$. Qualitatively, one can observe that the L^1-norm and Huber function-based approaches approximate the target solution the best by providing solutions with steep discontinuities. On the other hand, the L^2-norm minimization of the residual leads to a solution that presents undershoots and overshoots before and after the discontinuity, respectively.

The relative errors in L^1-norm between the true solution and each ROM solution

$$E_{\text{rel}}(\mu) = \frac{||w_{\text{true}}(\mu) - w_{\text{ROM}}(\mu)||_1}{||w_{\text{true}}(\mu)||_1} \tag{49}$$

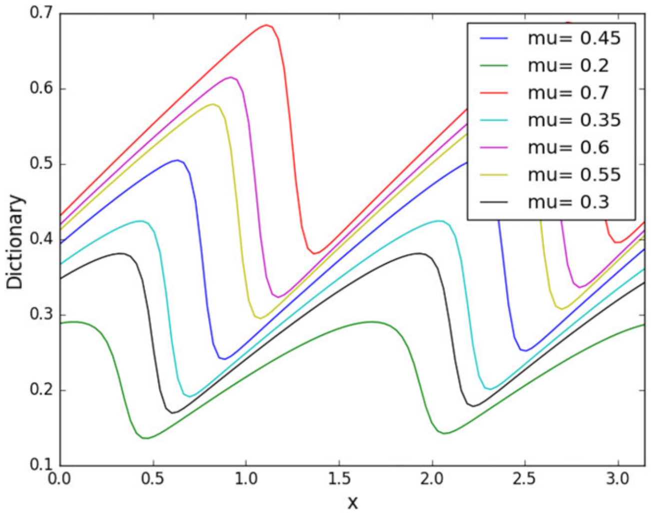


FIGURE 4 Members of the dictionary $\mathbf{V}$ at $t = \pi$ for Burgers' equation [Colour figure can be viewed at wileyonlinelibrary.com]

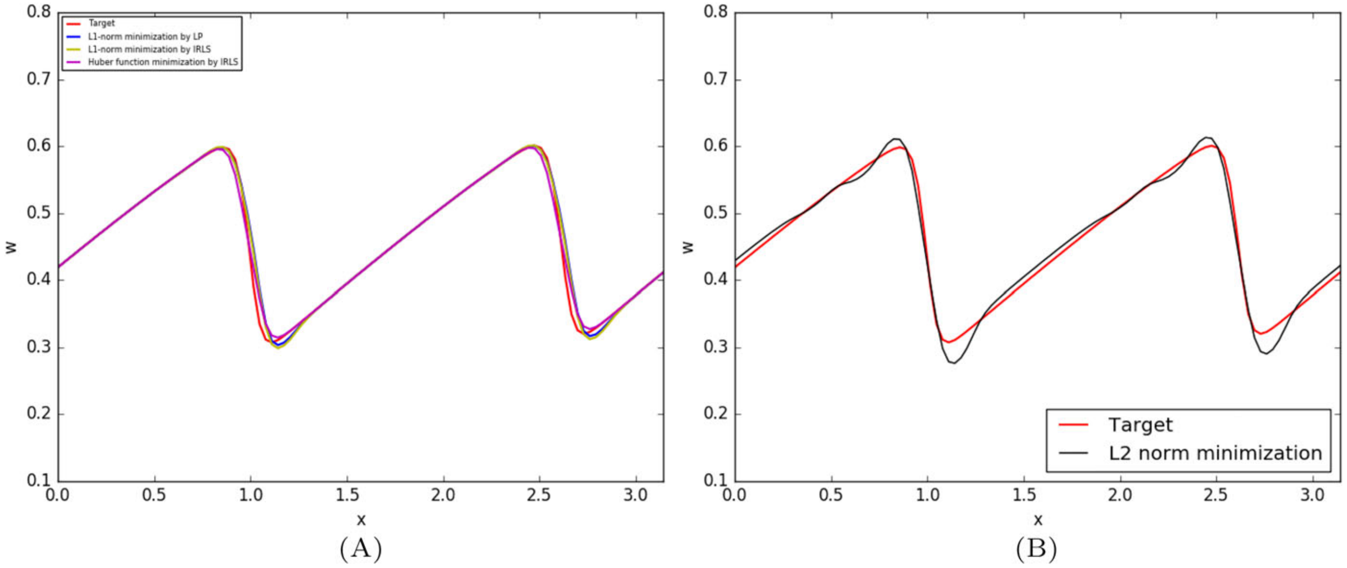
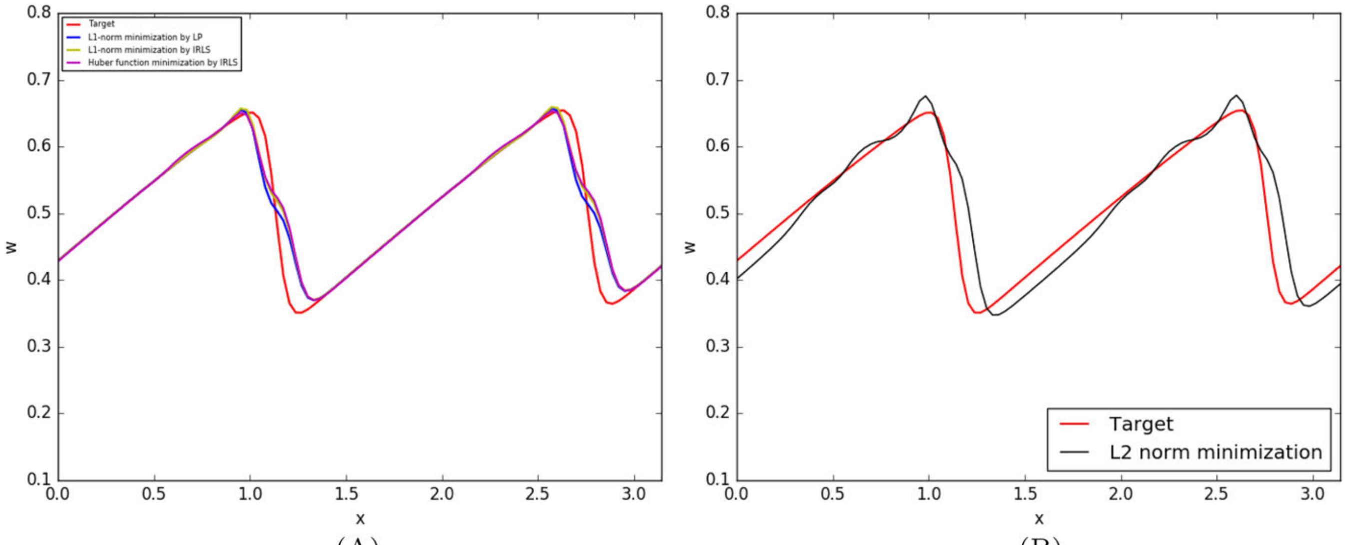


FIGURE 5 Comparison of all model-order reduction approaches at $t = \pi$ for the target solution $\mu_1 = 0.575$. IRLS, Iteratively reweighted least squares; LP, Linear program [Colour figure can be viewed at wileyonlinelibrary.com]

that computed for the target $\mu \in \mathcal{P}$ using different minimization procedures are reported in Table 1. One can observe that the approaches based on L^1-norm minimization (including the Huber function) lead to the smallest errors. In that same table, the CPU timings are reported. One can observe that the linear programming procedure is less computationally expensive than the IRLS approach and more than 3 times faster than the Huber approach. Nevertheless, even if the Huber function minimization approach is more expensive, it leads to much more accurate reduced solutions, as observed in Figures 5 and 6. These figures show the qualitative behavior of the L^1-norm types minimization in comparison with the quantitative results, which are more or less in the same range for each norm considered, including L^2-norm. We can use the RB method to approximate the solution of the problem (48) for a target parameter range from 0.2 to 0.7. In Figure 7, we report the error (49) as a function of the target parameter $\mu \in \mathcal{P}$. More precisely, we show the linear interpolation of the RB approximation error computed for 98 equidistant target values between 0.2 and 0.7. The vertical dashed lines are plotted in correspondence with the parameter values selected by the greedy algorithm 1. It is obvious that the RB approximation error tends to vanish close to the values selected by the greedy algorithm. We also consider different mesh sizes and, performing the relative error in L^1-norm using L^1-norm minimization by LP and L^2-norm minimization, we can observe a slightly better order of convergence when using L^1-norm–type minimizations than L^2-norm minimization (see Figure 8).



(A)

(B)

FIGURE 6 Comparison of all model-order reduction approaches at $t = \pi$ for the target solution $\mu_2 = 0.65$. IRLS, Iteratively reweighted least squares; LP, Linear program [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Unsteady Burgers' equation: relative errors in L^1-norm and CPUs for solutions at time $t = \pi$, using different minimization techniques for a mesh with $N = 100$ points and the target $\mu_1 = 0.575$

	HDM	L^2-norm	L^1-norm (LP)	L^1-norm (IRLS)	Huber function (IRLS)
$E_{\mathrm{rel}}(\mu)$	-	3.304e-6	1.510e-6	1.499e-6	1.229e-6
CPU timings (s)	7.391e-2	4.412e-4	6.108e-4	8.874e-4	1.959e-3

Abbreviations: IRLS, Iteratively reweighted least squares; LP, linear program.

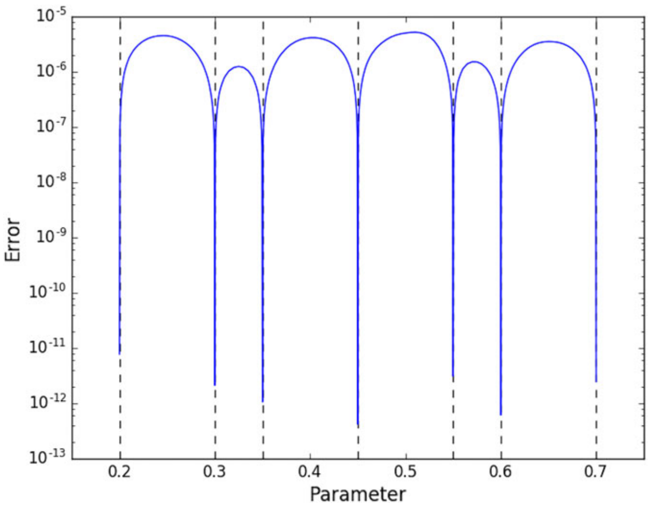


FIGURE 7 The reduced basis approximation error as a function of the target parameter in the range [0.2, 0.7] [Colour figure can be viewed at wileyonlinelibrary.com]

Finally, the reduced coordinates associated with each ROM are reported in Figure 9. The L^1-norm and Huber function minimization types lead to sparse solutions whether L^2-norm minimization has only nonzero contributions from all dictionary members.

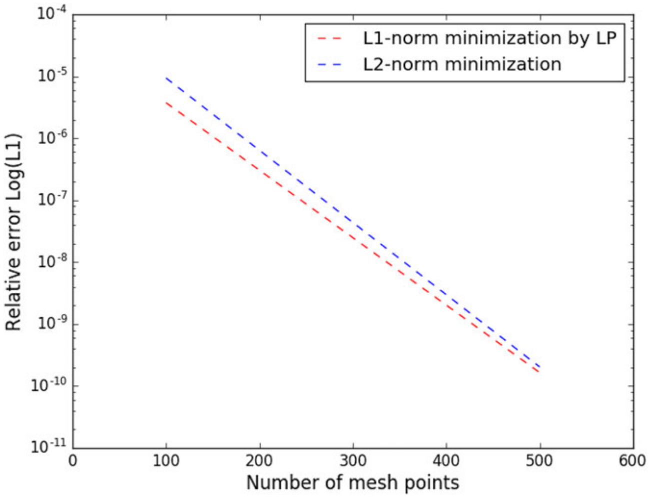


FIGURE 8 Convergence plots of the true relative errors using L^1-norm minimization by linear program (LP) and L^2-norm minimization [Colour figure can be viewed at wileyonlinelibrary.com]

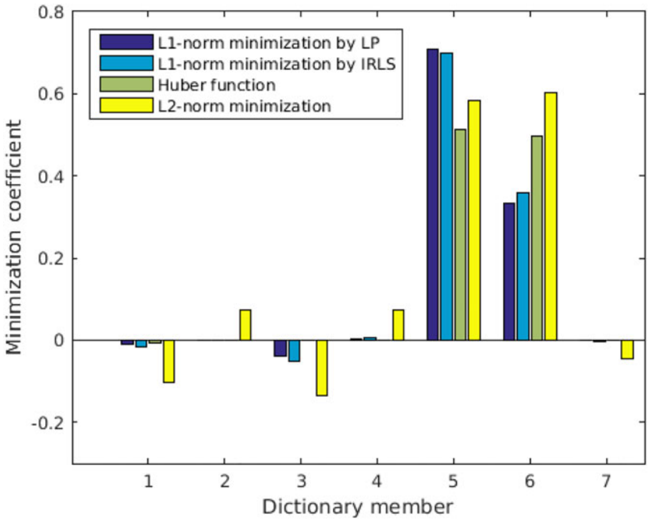


FIGURE 9 Reduced coordinates of the solutions for all model-order reduction approaches at $\mu_1 = 0.575$. IRLS, Iteratively reweighted least squares; LP, Linear program [Colour figure can be viewed at wileyonlinelibrary.com]

9.2 | Euler equations

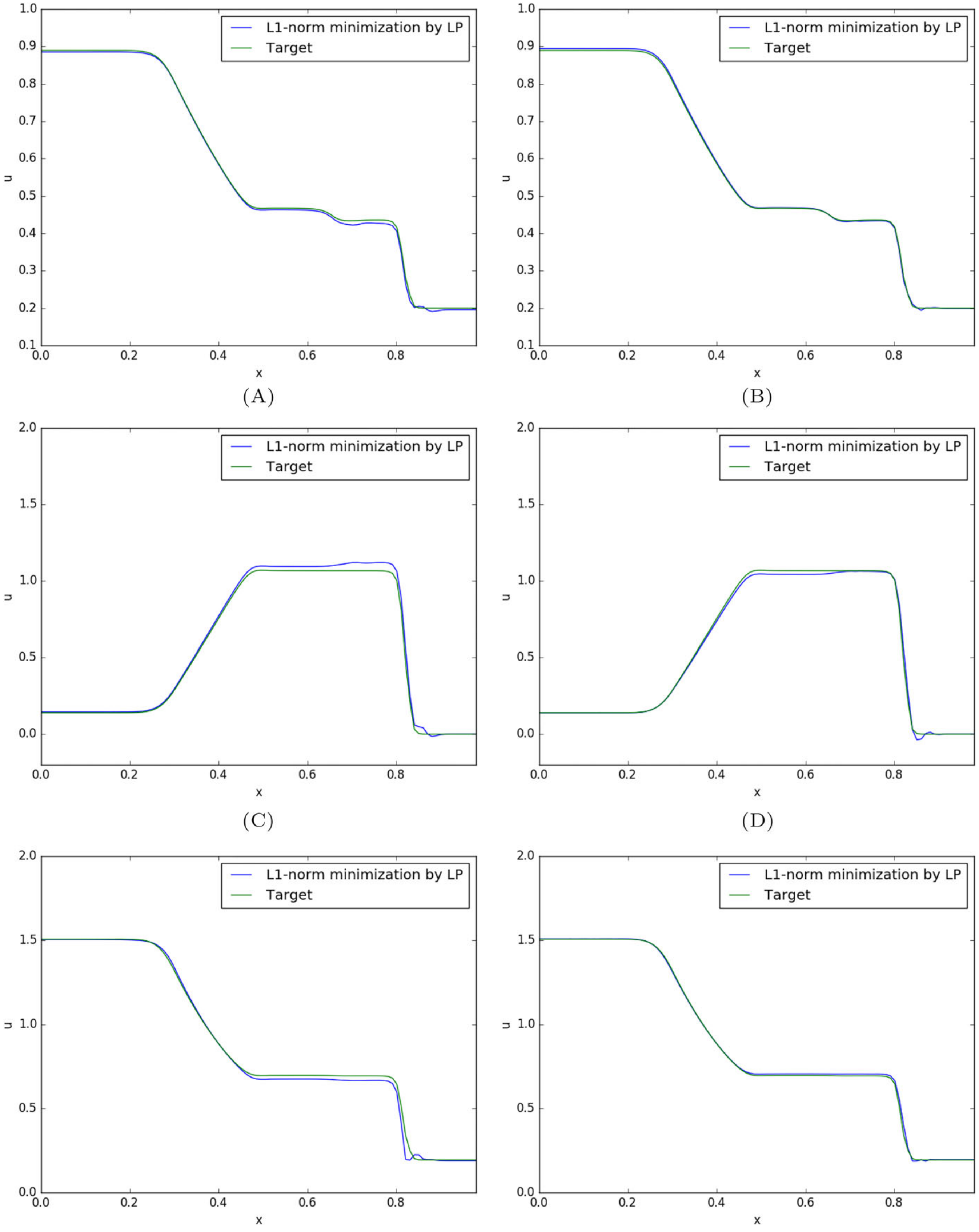
The 1D Euler equations are considered on $\Omega = [0, 1]$

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho w\\ E\end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix}\rho w\\ \rho w^2 + p\\ w(E+p)\end{pmatrix} = 0, \tag{50}$$

for which $\mathbf{W} = (\rho, \rho w, E)^T$ and the pressure is given by

$$p = (\gamma - 1)\left(E - \frac{1}{2}\rho w^2\right), \tag{51}$$

with $\gamma = 1.4$. For this problem, we have taken a second-order finite volume scheme with MUSCL extrapolation on the characteristic variables and the limiter is minmod on all waves. The simulations are displayed at the final time $T_{\text{fin}} = 0.16$ and it is obtained with 100 grid points and a time step of 0.001. As in the work of Abgrall et al,[35] we choose that the conservative initial conditions $\mathbf{W}_0(x; \mu)$, $\mu \in [0, 1]$ are constructed from the linear combination of the Sod and Lax shock tube problems. As suggested in the aforementioned work,[35] we have reconstructed the density, momentum, and total energy independently, and here, we are using the L^1-norm minimization by LP. The purpose of this section is to show the behavior of the reconstructed solution based on the number of elements in the dictionary $\mathbf{V}$ and for different targets.



(A)

(B)

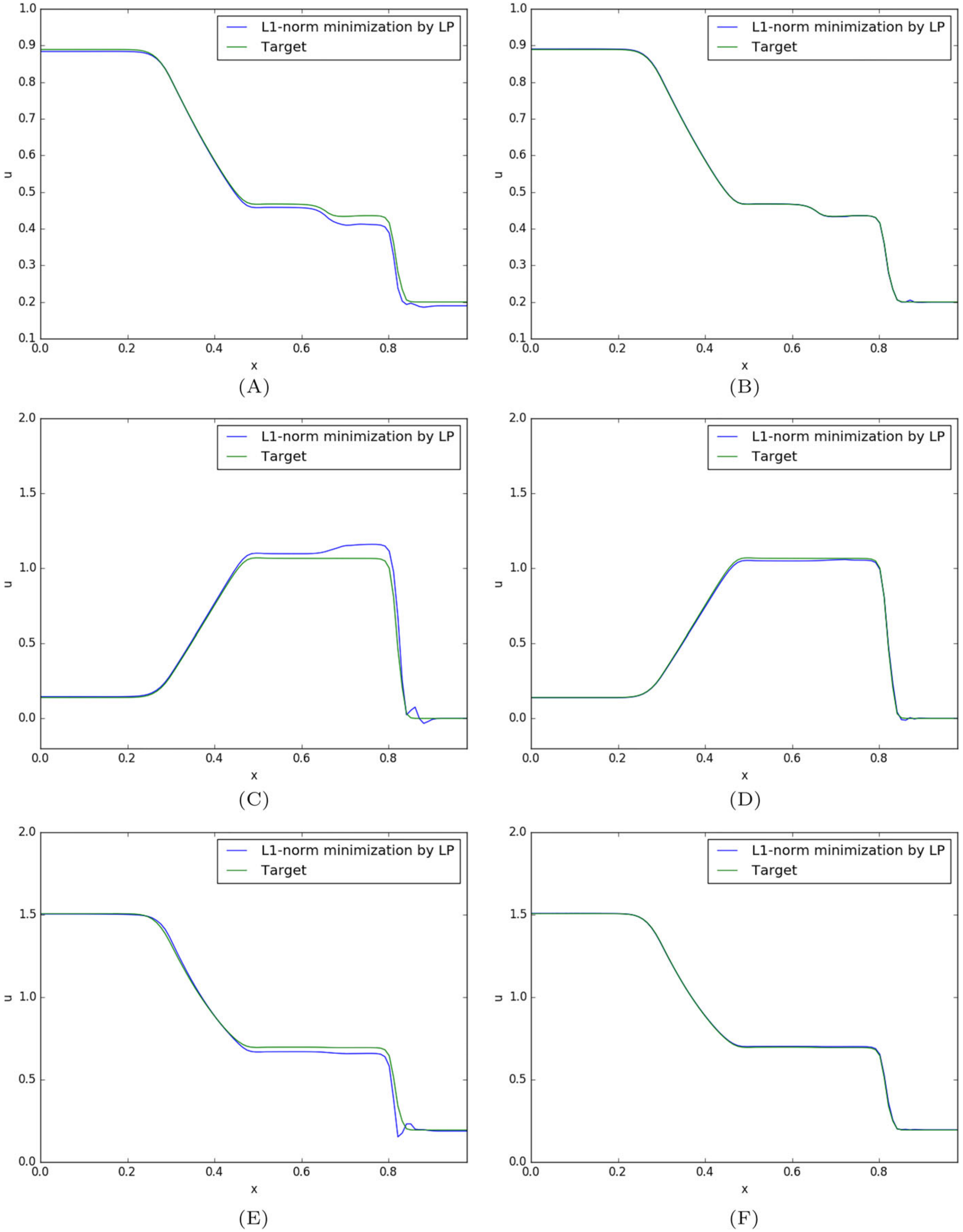
(C)

(D)

(E)

(F)

FIGURE 10 Reconstructed solution of density, momentum, and total energy with 5 (left) and 7 (right) elements in the dictionary for the target $\mu_1 = 0.3$ at time $t = 0.16$. LP, Linear program [Colour figure can be viewed at wileyonlinelibrary.com]



(A)

(B)

(C)

(D)

(E)

(F)

FIGURE 11 Reconstructed solution of density, momentum, and total energy with 5 (left) and 7 (right) elements in the dictionary for the target. $\mu_2 = 0.5$ at time $t = 0.16$. LP, Linear program [Colour figure can be viewed at wileyonlinelibrary.com]

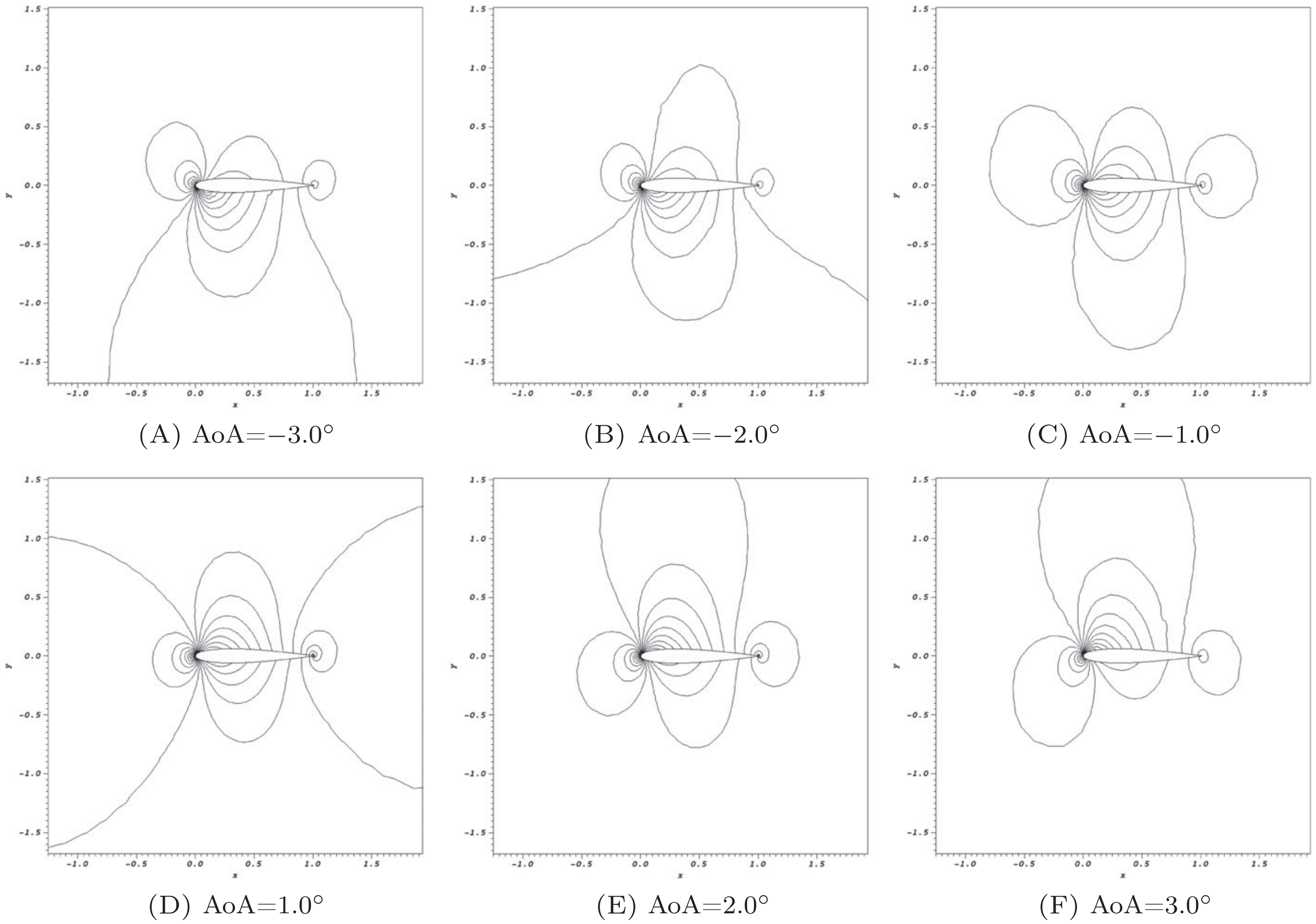
The greedy sampling algorithm is applied to construct a set of parameters $\mathcal{D}$ that is accurate in the parametric domain $\mathcal{P} = [0.1, 0.8]$. For that purpose, a set C containing $N_c = 15$ training parameters randomly distributed are considered. An initial dictionary is considered and then 4 and 6, respectively, greedy iterations are performed, resulting in a dictionary $\mathbf{V}$ with $k = 5$ and $k = 7$, respectively, members.

We start with the reconstruction of the solution for the target $\mu_1 = 0.3$ (see Figure 10) and then, in order to show that our L^1 minimization type can handle other parameters as a target, we are also illustrating the solution for $\mu_2 = 0.5$ (see Figure 11). One can observe that increasing just a little the number of elements in the dictionary, in our example only with two more RB, will result in a more accurate reconstructed solution.

Remark 5. Besides our effort to show, by plotting the convergence, the sparsity, evaluating the errors, and computing the CPUs, that L^1-norm minimization combined with a greedy algorithm improves the quality of the reduced solution, one can also compare these results with the ones obtained in the work of Abgrall et al[35] and a big improvement will be observed.

9.3 | Nonlinear steady problems: a 2D example

The extension to multidimensions is straightforward. We have started from a code using a second-order oscillation-free method for solving the 2D Euler equations. A description of this method can be found, for example, in the work of Abgrall.[55] Note that this specific choice has no impact on the reduced-order model because this algorithm is coded in Python on top of the CFD code, which is called as a black box. Any other CFD method would do the job. The used CFD mesh has 4510 grid points, which corresponds to a total of 18040 unknowns. For this numerical experiment, the hyper-reduction is performed using only a set of $\mathcal{I} = 100$ degrees of freedom, instead of $N = 4510$. Hence, only 2.2% of the total points is used.



(A) AoA=−3.0°

(B) AoA=−2.0°

(C) AoA=−1.0°

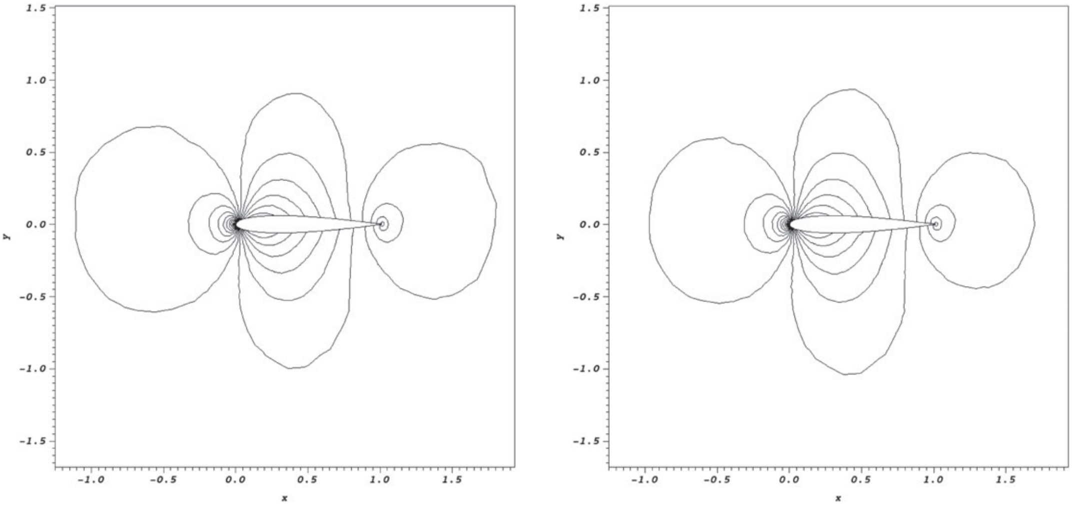
(D) AoA=1.0°

(E) AoA=2.0°

(F) AoA=3.0°

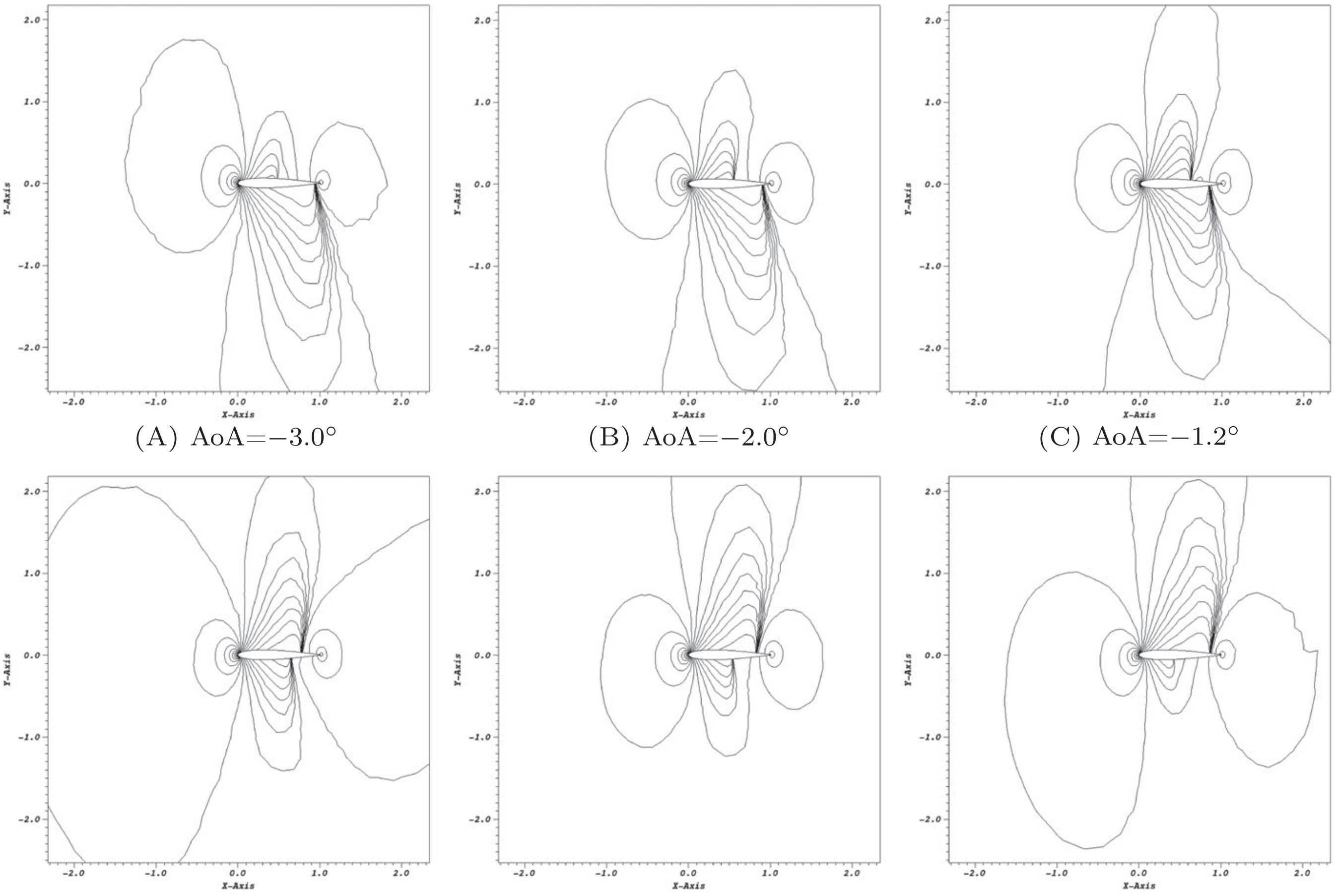
FIGURE 12 The components of the dictionary $\mathbf{V}$ constructed from the parameters $\mathcal{D} = \{-3.0°, -2.0°, -1.0°, 1.0°, 2.0°, 3.0°\}$ for Mach=0.65

The minimization is done on the mass, momentum components, and total energy. We introduce 4 sets of independent parameters that are the expansion coefficients for the dictionaries. In other words, we first run the CFD code to get a finite number of CFD solutions. Each solution is described as a vector of state variables (ρ, m_x, m_y, E). From this, we form 4 dictionaries, ie, one for the density, two for the velocities, and one for the total energy on which the reduced solutions are (independently) expanded. The minimization method is the straight L^1 minimization.



(A) ROM Solution (B) Exact Numerical Solution

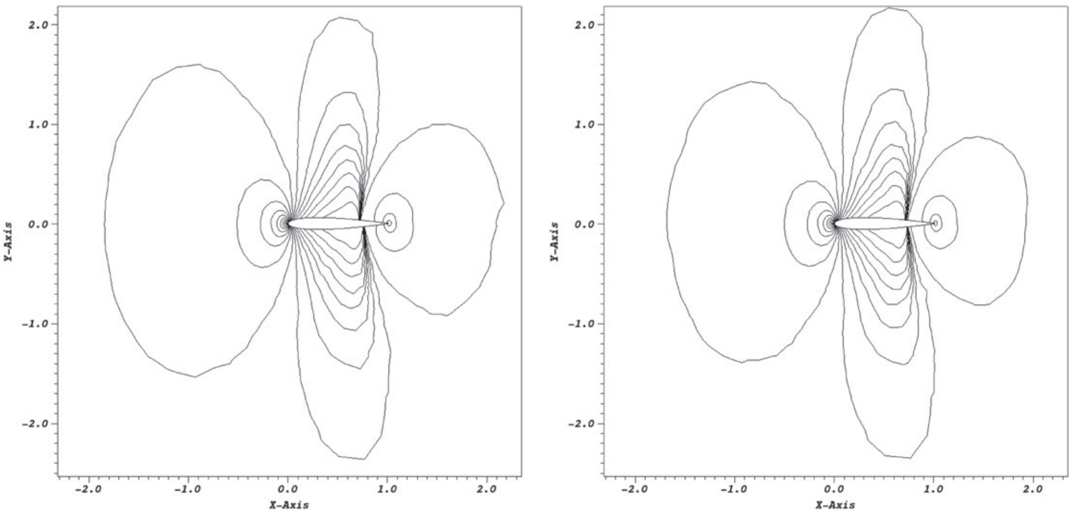
FIGURE 13 The ROM solution and the exact numerical solution for Mach= 0.65 and AoA=0.0°



(A) AoA=−3.0° (B) AoA=−2.0° (C) AoA=−1.2°

(D) AoA=1.0° (E) AoA=2.0° (F) AoA=3.0°

FIGURE 14 The components of the dictionary **V** constructed from the parameters $\mathcal{D} = \{-3.0°, -2.0°, -1.2°, 1.0°, 2.0°, 3.0°\}$ for Mach=0.85



(A) ROM Solution (B) Exact Numerical Solution

FIGURE 15 The ROM solution and the exact numerical solution for Mach= 0.85 and AoA=0.0°

In order to illustrate the technique, we have chosen a NACA012 profile with subsonic and transonic conditions. In the first case, the inflow mach number is $M_\infty = 0.65$ and the angle of attack (AoA) may change. The dictionary $\mathbf{V}$ is constructed by sampling parameters $\mathcal{D} = \{-3.0°, -2.0°, -1.0°, 1.0°, 2.0°, 3.0°\}$, representing the AoA, and the solution is sought for the predictive case $\mu = 0.0°$. We illustrate the elements in the dictionary (Figure 12), and then, using the L^1-norm minimization onto the convex hull of the dictionary, we obtain a ROM solution, which is comparable with the exact numerical solution (Figure 13).

In the second case, we consider that a shock exists ($M_\infty = 0.85$) to illustrate that our method can also deal with this kind of problems. The dictionary $\mathbf{V}$ is constructed by sampling the parameters $\mathcal{D} = \{-3.0°, -2.0°, -1.2°, 1.0°, 2.0°, 3.0°\}$ representing the AoA. Note that the symmetry is intentionally broken to show that the symmetry does not have any influence on the result. The solution is sought for the predictive case $\mu = 0.0°$. As in the first case, we illustrate the elements in the dictionary $\mathbf{V}$ to show that they are different (Figure 14). We obtain the following ROM solution using the L^1-norm minimization onto the convex hull of the dictionary (Figure 15). This one proves to be more robust. In this case, we initialize by the projection on the dictionary of a uniform flow. We may have positivity (of the density and/or the pressure) issues, and because of that, the projection on the convex hull revealed to be an efficient tool to control and avoid this issue. There is some discrepancy between the ROM solution and the exact numerical solution because the problem is very sensitive to the angle (this can be seen from the dictionary elements in Figure 14). As shown in Section 9.2, if we increase the number of elements in the dictionary, the ROM solution will be more similar to the exact numerical one, but in this case, there is no simple strategy to obtain error bounds as in the scalar case.

10 | CONCLUSIONS, PERSPECTIVES

We have presented a general framework to approximate the solution of steady and unsteady hyperbolic problems. The solution can be smooth or discontinuous and, in the unsteady case, moving waves may exist. Starting from any standard scheme (explicit or implicit), the reduced-order solution is obtained at each time step (or each iteration in the implicit case) from a minimization problem in the L^1-norm. We gave a sufficient condition to be able to solve the problem and discuss the practical aspects of the method. It is illustrated by several examples dealing with linear and nonlinear problems, scalar, and systems in 1- and 2-space dimensions. A rough error estimate is given based on the successive projections and the initial solution.

ACKNOWLEDGEMENTS

The first author was funded in part by the MECASIF project (2013-2017), funded by the French, and SNF grant 200021_153604 of the Swiss National Foundation.

The second author has been supported in part by the SNF grant 200021_153604 of the Swiss National Foundation.

The authors would like to thank Y. Maday, LJLL, Université Pierre et Marie Curie, and D. Amsallem for very insightful conversations. T. Kaman, University of Arkansas (USA), is also warmly acknowledged for her insight in several coding issues. RC is also thankful to D. Torlo, University of Zürich, for very productive discussions.

ORCID

R. Abgrall http://orcid.org/0000-0002-5553-7476

REFERENCES

1. Ito K, Ravindran S. A reduced-order method for simulation and control of fluid flows. *J Comput Phys*. 1998;143(2):403-425.
2. Ly HV, Tran HT. Modeling and control of physical processes using proper orthogonal decomposition. *Math Comput Model*. 2001;33(1-3):223-236.
3. Hovland S, Gravdahl JT, Willcox K. Explicit model predictive control for large-scale systems via model reduction. *J Guid Control Dyn*. 2008;31(4):918-926.
4. Amsallem D, Deolalikar S, Gurrola F, Farhat C. Model predictive control under coupled fluid-structure constraints using a database of reduced-order models on a tablet. Paper presented at: 21st AIAA Computational Fluid Dynamics Conference; 2013; San Diego, CA.
5. Amsallem D, Cortial J, Farhat C. Toward real-time computational-fluid-dynamics-based aeroelastic computations using a database of reduced-order information. *AIAA J*. 2010;48(9):2029-2037.
6. LeGresley PA, Alonso JJ. Airfoil design optimization using reduced order models based on proper orthogonal decomposition. Paper presented at: Fluids 2000 Conference and Exhibit; 2000; Denver, CO.
7. Amsallem D, Zahr MJ, Choi Y, Farhat C. Design optimization using hyper-reduced-order models. *Struct Multidiscip Optim*. 2015;51(4):919-940.
8. Bui-Thanh T, Willcox K, Ghattas O. Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications. *AIAA J*. 2008;46(10):2520-2529.
9. Kunisch K, Volkwein S. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM J Numer Anal*. 2003;40(2):492-515.
10. Veroy K, Prud'homme C, Rovas DV, Patera AT. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. Paper presented at: 16th Computational Fluid Dynamics Conference; 2003; Orlando, FL.
11. Kunisch K, Volkwein S. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numer Math*. 2001;90(1):117-148.
12. Grepl MA, Patera AT. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Math Model Numer Anal*. 2005;39(1):157-181.
13. Rozza G, Huynh DBP, Patera AT. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Arch Comput Methods Eng*. 2008;15(3):229-275.
14. Quarteroni A, Manzoni A, Negri F. *Reduced Basis Methods for Partial Differential Equations*. Cham, Switzerland: Springer International Publishing; 2016.
15. Hesthaven JS, Rozza G, Stamm B. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Cham, Switzerland: Springer International Publishing; 2016.
16. Benner P, Ohlberger M, Patera A, Rozza G, Urban K. *Model Reduction of Parametrized Systems*. Cham, Switzerland: Springer International Publishing; 2017.
17. Benner P, Ohlberger M, Cohen A, Willcox K. *Model Reduction and Approximation: Theory and Algorithms*. Philadelphia, PA: SIAM; 2017.
18. Barone MF, Kalashnikova I, Segalman D, Thornquist H. Stable Galerkin reduced order models for linearized compressible flow. *J Comput Phys*. 2009;228(6):1932-1946.
19. Dahmen W, Plesken C, Welper G. Double greedy algorithms: Reduced basis methods for transport dominated problems. *ESAIM: Math Model Numer Anal*. 2014;48(3):623-663.
20. Binev P, Cohen A, Dahmen W, DeVore R, Petrova G, Wojtaszczyk P. Convergence rates for greedy algorithms in reduced basis methods. *SIAM J Math Anal*. 2011;43(3):1457-1472.
21. Gerbeau JF, Lombardi D. Approximated Lax pairs for the reduced order integration of nonlinear evolution equations. *J Comput Phys*. 2014;265:246-269.
22. Iollo A, Lombardi D. Advection modes by optimal mass transfer. *Phys Rev E*. 2014;89(2):022923.
23. Ohlberger M, Rave S. Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing. *Comptes Rendus Math*. 2013;351(23-24):901-906.
24. Taddei T, Perotto S, Quarteroni A. Reduced basis techniques for nonlinear conservation laws. *ESAIM: Math Model Numer Anal*. 2015;49(3):787-814.
25. Amsallem D, Farhat C. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA J*. 2008;46(7):1803-1813.
26. Dihlmann M, Drohmann M, Haasdonk B. Model reduction of parametrized evolution problems using the reduced basis method with adaptive time-partitioning. Paper presented at: Proceedings of ADMOS; 2011; Paris, France.

27. Amsallem D, Zahr MJ, Farhat C. Nonlinear model order reduction based on local reduced-order bases. *Int J Numer Methods Eng*. 2012;92(10):891-916.
28. Maday Y, Stamm B. Locally adaptive greedy approximations for anisotropic parameter reduced basis spaces. *SIAM J Sci Comput*. 2013;35(6):A2417-A2441.
29. Kaulmann S, Haasdonk B. Online greedy reduced basis construction using dictionaries. Paper presented at: Proceedings of the VI International Conference on Adaptive Modeling and Simulation; 2013; Lisbon, Portugal.
30. Brunton SL, Tu JH, Bright I, Kutz JN. Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. *SIAM J Appl Dyn Syst*. 2014;13(4):1716-1732.
31. Sirovich L. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Q Appl Math*. 1987;45(3):561-571.
32. Balajewicz M, Amsallem D, Farhat C. Projection-based model reduction for contact problems. *Int J Numer Methods Eng*. 2016;106(8):644-663.
33. Carlberg K, Bou-Mosleh C, Farhat C. Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *Int J Numer Methods Eng*. 2011;86(2):155-181.
34. Carlberg K, Farhat C, Cortial J, Amsallem D. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *J Comput Phys*. 2013;242:623-647.
35. Abgrall R, Amsallem D, Crisovan R. Robust model reduction by L^1-norm minimization and approximation via dictionaries: application to non linear hyperbolic problems. *Adv Model Simul Eng Sci*. 2016;3(1).
36. Nocedal J, Wright SJ. *Numerical optimization*. New York, NY: Springer; 2006.
37. Willcox K, Peraire J. Balanced model reduction via the proper orthogonal decomposition. Paper presented at: 15th AIAA Computational Fluid Dynamics Conference; 2001; Anaheim,CA.
38. Lee DD, Seung HS. Learning the parts of objects by non-negative matrix factorization. *Nature*. 1999;401(6755):788-791.
39. Veroy K, Patera AT. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigurous reduced-basis a posteriori error bounds. *Int J Numer Methods Fluids*. 2005;47(8-9):773-788.
40. Haasdonk B, Ohlberger M. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Math Model Numer Anal*. 2008;42(2):277-302. https://doi.org/10.1051/m2an:2008001
41. Candes E, Romberg J. Robust signal recovery from incomplete observations. Paper presented at: International Conference on Image Processing; 2006; Atlanta, GA.
42. Donoho DL. Compressed sensing. *IEEE Trans Inf Theory*. 2006;52(4):1289-1306.
43. Boyd S, Vandenberghe L. *Convex Optimization*. Cambridge, UK: Cambridge University Press; 2004.
44. Lavery JE. Nonoscillatory solution of the steady-state inviscid burgers' equation by mathematical programming. *J Comput Phys*. 1988;79(2):436-448.
45. Lavery JE. Solution of steady-state one-dimensional conservation laws by mathematical programming. *SIAM J Numer Anal*. 1989;26(5):1081-1089.
46. Guermond JL, Marpeau F, Popov B. A fast algorithm for solving first-order pdes by l1-minimization. *Commun Math Sci*. 2008;6(1):199-216.
47. Guermond JL, Popov B. L^1-approximation of stationary Hamilton–Jacobi equations. *SIAM J Numer Anal*. jan 2009;47(1):339-362.
48. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Berlin, Germany: Springer; 1997.
49. Godlewski E, Raviart P. *Hyperbolic Systems of Conservation Laws*. Paris, France: Ellipses; 1991.
50. Huber PJ, Ronchetti EM. *Robust Statistics*. Hoboken, NJ: Wiley; 2011.
51. Daubechies I, DeVore R, Fornasier M, Gunturk CS. Iteratively re-weighted least squares minimization for sparse recovery. *Commun Pure Appl Math*. 2010;63(1):1-38.
52. Abgrall R, Larat A, Ricchiuto M. Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes. *J Comput Phys*. 2011;230(11):4103-4136.
53. Ricchiuto M. An explicit residual based approach for shallow water flows. *J Comput Phys*. 2015;280:306-344.
54. Abgrall R, De Santis D. High order residual distribution scheme for navier-stokes equations. Paper presented at: 20th AIAA Computational Fluid Dynamics Conference; 2011; Honolulu, HI.
55. Abgrall R. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *J Comput Phys*. 2006;214(2):773-808.

How to cite this article: Abgrall R, Crisovan R. Model reduction using L^1-norm minimization as an application to nonlinear hyperbolic problems. *Int J Numer Meth Fluids*. 2018;87:628–651. https://doi.org/10.1002/fld.4507