A Greedy Non-Intrusive Reduced Order Model for Fluid Dynamics

Wang Chen[1]
Northwestern Polytechnical University, Xi'an 710072, P.R. China

Jan S Hesthaven[2]
Swiss federal Institute of Technology in Lausanne(EPFL), CH-1015 Lausanne, Switzerland

Bai Junqiang[3], Yasong Qiu[4], and Yang Tihao[5]
Northwestern Polytechnical University, Xi'an 710072, P.R. China

Zhang Yang[6]
Xi'an Jiaotong University, Xi'an 710018, P.R. China

A greedy non-intrusive reduced order method (ROM) is proposed for parameterized time-dependent problems with an emphasis on problems in fluid dynamics. The non-intrusive ROM (NIROM) bases on a two-level proper orthogonal decomposition (POD) to extract temporal and spatial reduced basis from a set of candidates, and adopts the radial basis function (RBF) to approximate undetermined coefficients of extracted reduced basis. Instead of adopting uniform or random sampling strategies, the candidates are determined by an adaptive greedy approach to minimize the overall offline computational cost. Numerical studies are presented for a two-dimensional diffusion problem as well as a lid-driven cavity problem governed by incompressible Navier-Stokes equations. The results demonstrate that the greedy non-intrusive ROM (GNIROM) predicts the flow field accurately and efficiently.

---

[1] PH.D candidate, School of Aeronautics, datouroony@163.com.
[2] Professor, Chair of Computational Mathematics and Simulation Science(MCSS); jan.hesthaven@epfl.ch .
[3] Professor, School of Aeronautics, junqiang@nwpu.edu.cn (Corresponding Author).
[4] Assistant research fellow, School of Aerospace, qiuyasong@nwpu.edu.cn
[5] PH.D candidate, School of Aeronautics, xiaoyaoyangtihao@163.com .
[6] Lecturer, School of Aerospace, youngz@xjtu.edu.cn

Nomenclature

$\alpha^{\theta}(k,m)$ = undetermined coefficients in ROM approximation, $k = 1, 2, \cdots K$, $m = 1, 2, \cdots M$

$g^{\theta}(x,t)$ = boundary conditions

M = correlation matrix in POD

N_t^* = coarse temporal mesh size

N_t = number of time steps

N_x^* = coarse spatial mesh size

N_x = spatial mesh size

t = time

θ = vector of parameters in researched problem

u^{θ} = exact solution of the parameterized nonlinear PDE problem

$\hat{u}^{\theta}$ = approximated solutions with ROM

$u_0^{\theta}(x)$ = initial conditions

$\hat{v}^{\theta}(x,t)$ = auxiliary term in ROM approximation

x = spatial coordinates

Δ = error estimator of Greedy approach

Ω = physical space

$\partial\Omega$ = boundary of physical space

$\varphi^k(x)$ = spatial basis of the parameterized nonlinear problem, $k = 1, 2, \cdots, K$

$\xi^m(t)$ = temporal basis of the parameterized nonlinear problem, $m = 1, 2, \cdots, M$

λ_k = the k-th eigen value for Correlation matrix

μ = penalized parameter in least square approach

$\phi(r)$ = RBF kernel

γ_i^{km} = RBF coefficients

σ = scaling factor of RBF

$\psi(r)$ = RBF-QR kernel

I. Introduction

Although the ability to numerically model complex phenomena is sustainably improved, numerical simulation in optimization, design and control with lots of degree of freedoms (DOFs) still remains challenging due to the high computational cost. As an engineering compromise to approximate the full problem, the development of reduced-order method (ROMs) [1, 2] that enables approximating solutions with an acceptable loss of accuracy has gained a substantial interest. ROMs aim to approximate the full problem by projecting it onto a specific subspace with much smaller DOFs. Once properly chosen, the reduced linear space can adequately represent the dynamics of the full system at a substantially reduced cost. Examples of these developments include proper orthogonal decomposition (POD) [3], Harmonic Balance approach (HB) [4], and Volterra theory [5].

Usually, ROMs extract a set of reduced basis functions from snapshots of the full solutions and then adopt Galerkin projection [6] to approximate the full system with the reduced basis. As the projection approaches depend on governing equations, the source codes have to be modified, leading to the fact that the process is inconvenient for complex nonlinear problems and even impossible when the source code is unavailable. To address this concern, there is a recent interest in non-intrusive ROMs (NIROMs) which seeks to develop ROMs based solely on access to snapshots and doesn't utilize governing equations. In the context of fluid dynamics, [7] presents a second-order Taylor series method as well as an approach based on a Smolyak sparse grid collocation method, and applies the two NIROMs to simulate the flow past a cylinder with wind driven gyre. In [8], the authors explore the use of empirical interpolation method [9] to recover a reduced model based on a non-intrusive approach. Other researchers [10–13] seek to develop NIROMs by basing on POD and radial basis function (RBF). In these approaches, RBF is used to interpolate undetermined coefficients of reduced basis. Although there exist other nonlinear interpolation approaches such as BP network or Kriging, RBF is more popular in non-intrusive ROMs' development. The reason is that RBF is flexible, convenient and accurate when approximating a function through scattered data, demonstrated in [14]. It has also been validated in numerical cases that the non-intrusive approaches using RBF can present a good performance in fluid dynamics, even for strong nonlinear flow fields. For example, [11] adopts POD and RBF onto unsteady flow over an oscillating ONERA

M6 wing, and [12] demonstrates that the NIROM also performs well in a lock exchange problem as well as a flow past a cylinder.

Development of ROMs for parameterized time-dependent problems is another issue. For such problems, ROMs for time-dependent parameterized partial differential equations (PDEs) have to approximate solutions as a function of time, spatial coordinates and a parameter vector, which turns out to be more challenging especially when the boundary conditions vary with time. Some work on this topic is dealing with simple parameterized time-dependent PDEs such as convection-diffusion problems [15] and boussinesq equations [16]. To solve unsteady parameterized Navier-Stokes equations, [17] and [18] both utilize POD-Galerkin and achieve a good approximation in adaptation of Mach and Reynolds number. [19] also adopts intrusive ROMs in patient-specific haemodynamics with shape parameters. Discrete empirical interpolation method (DEIM) proposed in [20] is an intrusive ROM for nonlinear problems. DEIM reduces the complexity as well as the dimension of nonlinear terms and has proven to be efficient and accurate to approximate nonlinear problems. Although intrusive ROMs have made great achievements in model adaption, this paper prefer to adopt NIROMs since those don't rely on governing equations as presented above. [21] presents a general NIROM for parameterized time-dependent nonlinear PDEs. This approach constructs snapshots and local basis functions for any parameter point by interpolating those from samples and then generates hyper-surfaces to represent the dynamical system over the reduced space. [13] proposes a NIROM consisting of two-level POD as well as RBF, which can deal with general problems with time-dependent boundary conditions. Considering that the NIROM in [13] is simpler on implementation and requires less memory than that in [21], the former is adopted.

Besides, it is equally important to carefully selecting appropriate snapshots in model adaption. The process to generate snapshots by a numerical solver or experiments contributes a lot to the computational cost for ROMs. Thus, under the premise that requirements of ROMs' approximation accuracy are reached, fewer candidates are chosen, more efficient the ROMs perform. However, due to the variation of possible solutions, it is impossible to determine the optimal candidates for snapshot generation. Usually a uniform sampling or a random strategy such as Latin Hypercube sampling [22], Centroidal voronoi tessellations (CVT) [23], Monte Carlo methods [24], etc., is used

to generate snapshots for ROMs. As these approaches don't utilize the useful information from the researched problem, they are not optimal. To guarantee the accuracy of the reduced model, they often require an excessive number of snapshots, which leads to an unacceptable computational cost. To deal with this difficulty, recently, some researches have been devoted to propose adaptive sampling approaches for guiding the selection of snapshots. In [25], an algorithm that iteratively applies surrogate-model optimization in the generation of snapshots is developed. Other related approaches are based on a greedy algorithm. In such approaches, the snapshots are adaptively determined by finding the locations in the predetermined parameter space where the error estimator reaches the maximum, quantifying the quality of the ROM. In [26, 27], a POD-Greedy approach is introduced to recover the ROMs, albeit in an intrusive manner, with a discussion of convergence offered in [28, 29]. Patera et al. [30–33] has researched a lot on Greedy sampling for POD-Galerkin method when approximating parameterized problems. Tan Bui-Thanh and Karen Willcox [34] also propose a model-constrained adaptive Greedy sampling method for large-scale systems with high-dimensional parametric input spaces [35]. However, to the authors' knowledge, little work has been done to develop greedy sampling for non-intrusive methods.

The paper aims to develop a greedy sampling approach for the NIROM put forward in [13], which is applicable for time-dependent parameterized nonlinear problems. The non-intrusive ROM employs a two-level approach to construct the spatial and temporal basis functions. And RBF is adopted to estimate the undetermined coefficients of reduced basis because of its features stated above. While the discussion in this paper is appropriate for problems in a general degree, the paper mainly focus on problems and examples in the context of fluid dynamics. Different from previous work in [13], the reduced model in this paper adaptively generate snapshots basing on a greedy approach to minimize the overall computational cost, thus forming a greedy NIROM (GNIROM).

The structure of this paper is organized as follows. In Section II, a NIROM is developed, including a two-level basis development and RBF approximation. The use of the RBF-QR method [36] is also briefly discussed to improve RBF's robustness. Afterwards, a Greedy sampling approach is proposed in Section III to reduce the computational cost when generating the reduced basis. This sets the stage for Section IV where the efficiency and accuracy of present ROM is confirmed through

problems of heat conduction and lid-driven cavity flows. Finally, Section V lists a few concluding remarks and plans in our follow-up work.

II. Construction of the non-intrusive reduced order model

The following part presents the construction of the NIROM to approximate time-dependent problems, which is parameter dependent. It begins the discussion by focusing on the approach put forward in [13] in which the basis is extracted using a two-level POD. And these are subsequently combined with a RBF to estimate undetermined coefficients of the reduced basis.

A. Problem statement

Consider the general parameterized time-dependent PDEs with initial and boundary conditions:

$$\begin{cases} \frac{\partial u^\theta}{\partial t} + F(u^\theta) = f^\theta & x \in \Omega \times t \in (0, T] \\ u^\theta_{|t=0} = u^\theta_0(x) & x \in \Omega \\ u^\theta(x,t) = g^\theta(x,t) & x \in \partial\Omega \times t \in (0, T] \end{cases} \tag{1}$$

where $t \in [0, T]$ denotes time, $x \in \Omega$ are the spatial coordinate vector, and Ω is the physical domain with a boundary $\partial\Omega$. θ represents a vector of control parameters with dimension $p(p \geq 1)$. It is assumed that θ is within the range Θ. u^θ represents the state variables, which is a function of space, time and θ.

The general problems take three spaces into consideration, namely, the temporal space,the physics space and the parameter space. To deal with this kind of problems, the basic assumption is that $u^\theta(x,t)$ can be expressed as the following ansatz:

$$u^\theta(x,t) \approx \hat{u}^\theta(x,t) = \hat{v}^\theta(x,t) + \sum_{k=1}^{K}\sum_{m=1}^{M} \alpha^\theta(k,m)\varphi^k(x)\xi^m(t) \tag{2}$$

In (2), two types of basis functions are introduced: spatial basis functions $\varphi^k(x)$ and temporal basis functions $\xi^m(t)$. The coefficients $\alpha^\theta(k,m)(k = 1, 2, \cdots, K; m = 1, 2, \cdots, M)$ denote the undetermined coefficients in the approximation. $\hat{v}^\theta(x,t)$ denotes the auxiliary term created to properly account for boundary and initial condition thus the term is required to meet:

$$\begin{cases} \frac{\partial \hat{v}^\theta}{\partial t} - \triangle \hat{v}^\theta = 0 \qquad x \in \Omega \times t \in (0, T] \\ \hat{v}^\theta_{|t=0} = u_0^\theta(x) \qquad\qquad x \in \Omega \\ \hat{v}^\theta(x,t) = g^\theta(x,t) \qquad x \in \partial\Omega \times t \in (0, T] \end{cases} \tag{3}$$

and at the same time the reduced basis have to satisfy

$$\begin{cases} \varphi^k_{|\partial\Omega} = 0 \\ \xi^m(0) = 0 \end{cases} \tag{4}$$

The reason to adopt the ansatz (2) is that it can directly deal with general time-dependent problems with design parameters. And introducing temporal and spatial basis at the same time can help approximate the spatio-temporal dynamic system well, just like the work of biorthogonal decomposition (BOD) [37].

In this paper's first case, although the boundary conditions vary with time, the governing equation is the diffusion problem, the same to that of auxiliary PDEs, and adopting the NIROM without the axillary term can perform well in this case.

And in the second case, a problem with constant boundary conditions is considered, and $\hat{v}^\theta(x,t)$ should be correspondingly set as $u_0(x)$ to meet (3). The following is the proof:

Proof. Concretely, consider the initial condition,

$$\hat{u}^\theta(x,0) = \hat{v}^\theta(x,0) + \sum_{k=1}^{K}\sum_{m=1}^{M} \alpha^\theta(k,m)\varphi^k(x)\xi^m(0) \tag{5}$$

The second term on the right hand side in (5) vanishes because of the assumption that $\xi^m(0)_{|m=1,2,\cdots,M} = 0$. By introducing $\hat{v}^\theta(x,t) = u_0(x)$ into (5),

$$\hat{u}^\theta(x,0) = u_0(x) \tag{6}$$

For the boundary condition, the term $\sum_{k=1}^{K}\sum_{m=1}^{M} \alpha^\theta(k,m)\varphi^k(x)\xi^m(0)$ can also be ignored with $\varphi^k_{|\partial\Omega} = 0$, thus

$$\hat{u}^\theta_{|\partial\Omega}(t) = \hat{u}^\theta_{|\partial\Omega}(t = t_0) = g^\theta \tag{7}$$

Therefore, for problems with constant boundary conditions, $\hat{v}^\theta(x,t) = u_0(x)$ satisfies (3) and the boundary and initial conditions are satisfied. □

B. Basis creation by POD

POD, first developed to correlate statistical data [38], has been widely used to recover eigenmodes for spatio-temporal problems. The central idea is to derive an optimal linear basis from ensembles of statistical data with optimality measured in a least-square sense.

Consider a state variable $y(x,t)(x \in \Omega, t \in [0,T])$ with snapshots $y(x,t_i)(t_i \in [0,T], i = 1,2,\cdots,n; x \in \Omega)$. POD approximate solutions as

$$y(x,t) \approx y_0(x) + \sum_{i=1}^{K} a_i(t) \cdot \Phi_i(x) \tag{8}$$

where $y_0(x)$ represents the mean part. Orthogonal basis $\Phi_i(x)$ is extracted to approximate the fluctuating part $y'(x,t) = y'(x,t) - y_0(x)$ and K is the number of retained POD basis. POD seeks an optimal basis $\Phi_i(i = 1,2,\cdots K)$ in a Hilbert space such that

$$\max_{\Psi \in L^2} \frac{< (y', \Psi) >}{(\Psi, \Psi)} = \frac{< (y', \Phi) >}{(\Phi, \Phi)} \tag{9}$$

The inner product $(\cdot,\cdot)$ is defined as $(f,g) = \int_{x\in\Omega} f(x)g(x)dx$ and $< \cdot >$ denotes a statistically average operator. Assume that reduced basis functions $\Phi_i(x)$ are unit and orthogonal, $(\Phi_i, \Phi_j) = \delta_{ij}$. The maximization problem (9) is transformed equivalently to solve the following eigen problem of a two-point time correlation matrix. Define the correlation matrix $M \in R_{n\times n}$ with the elements as

$$M(i,j) = \frac{1}{n}(y_i, y_j) \qquad 0 \le i,j \le n \tag{10}$$

As M is a symmetric positive semi-definite matrix, according to the Hilbert-Schmidt theorem, there exists n nonnegative eigenvalues $\lambda_i(i = 1,2,\cdots n)$ for matrix M. Assume that $\lambda_i \ge \lambda_2 \ge \cdots \ge \lambda_n$. v_i denotes the relative eigenvector of λ_i, which satisfies that $M \cdot v_i = \lambda_i v_i$. The values of λ_i determines the "energy" which the relative basis vector captures. Define a small threshold $0 < \varepsilon < 1$, the retained modes number K is truncated through

$$K = min\{k \in [1,n] | \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \ge 1 - \varepsilon\} \tag{11}$$

Then the POD reduced basis can be determined by $\Phi_i(x) = \sum_{j=1}^{n} v_i(j) \cdot y(x,t_j)$.

C. Two-level basis development

To develop a ROM for the parameterized time-dependent PDEs (1), two-level approach, based on POD, is adopted in [13] to construct both the temporal and the spatial basis. The reason to

extract spatial and temporal modes separately is to introduce the coarse grids which aims to speed up ROM's offline computations. As the generations for spatial and temporal basis are similar, only the process of generating spatial modes is discussed here.

This paper defines a coarse spatial grid as $\chi = \{\chi_j \in \Omega_h(j = 1, 2, \cdots, N_x^*)\}$ and a coarse temporal one as $\Upsilon = \{t_n | 0 = t_1 < t_2 < \cdots t_{N_t^*} = T\}$. Furthermore, a set of samples $\theta_i \in \Theta(i = 1, 2, \cdots, N)$ are defined for generating snapshots from a given parameter space.

Through execution of the numerical solver, the shifted spatial snapshots can be recovered as $\Psi = \{u^{\theta_i}(\cdot, t_j) - v^{\theta_i}(\cdot, t_j), j = 1, 2, \cdots N_t^*\}$. For each sample θ_i, POD is adopted onto the temporal snapshots with the same threshold denoted by ε_1. K^i represents the number of modes truncated by ε_1. And $r^{k,i}(k = 1, 2, \cdots K^i)$ are the relative eigenvectors. The corresponding spatial modes are finally recovered as

$$\varphi^{k,i}(x) = \sum_{n=1}^{N_t^*} (r^{k,i})_n (u^{\theta_i}(x, t_n) - \hat{v}^{\theta_i}(x, t_n)) \tag{12}$$

With the aforementioned work, a total of $\sum_{i=1}^{N} K^i$ spatial basis can be obtained. And by application of SVD with another threshold ε_2, the eigenfunction would be further compressed while K spatial modes $\varphi^k(k = 1, 2, \cdots K)$ retains.

To achieve temporal modes, temporal snapshots are given by $\{u^{\theta_i}(\chi_j, \cdot) - \hat{v}^{\theta_i}(\chi_j, \cdot), j = 1, 2, \cdots N_x^*\}$. By using the similar approach, M temporal modes $\xi^m(m = 1, 2, \cdots M)$ can be recovered.

In the above process, coarse meshes are introduced and spatial and temporal modes are extracted independently, which aims to speed up ROM's offline computations. Detailed computation complexity for extracting reduced modes is listed in Table 1. If no coarse meshes are introduced and spatial and temporal modes are extracted at the same time by POD with the snapshot, the complexity is also listed in Table 2, where $\hat{K}^i$ denotes the extracted basis number for θ^i and $\hat{K}^i$ denotes extracted basis number compressed by SVD. For simple cases like the two cases in this paper, N_x^* and N_t^* are both much less than N_t^* . As a result, $N \cdot {N_x^*}^2 \cdot N_t + N \cdot {N_t^*}^2 \cdot N_x < N \cdot {N_t}^2 \cdot N_x$, ${N_t^*}^3 \cdot \sum_{i=1}^{N} K^i + {N_t}^3 \cdot \sum_{i=1}^{N} \hat{K}^i < {N_t}^3 \cdot \sum_{i=1}^{N} \hat{K}^i$, and a reduction of offline computational cost is achieved by comparing the two tables above. This is the reason for introducing coarse meshes and extracting spatial and temporal modes separately.

Table 1: Computation complexity to extract spatial and temporal modes separately

Process	Detailed Steps	Computational complexity
Achieve spatial modes	Assemble correlation matrix	$o(N \cdot {N_t^*}^2 \cdot N_x)$
	Extract modes for θ_i	$o({N_t^*}^3 \cdot \sum_{i=1}^{N} K^i)$
	Extract φ^k	$o(K \cdot (\sum_{i=1}^{N} K^i)^3)$
Achieve temporal modes	Assemble correlation matrix	$o(N \cdot {N_x^*}^2 \cdot N_t)$
	Extract modes for θ_i	$o({N_x^*}^3 \cdot \sum_{i=1}^{N} M^i)$
	Extract ξ^k	$o(M \cdot (\sum_{i=1}^{N} M^i)^3)$

Table 2: Computation complexity to extract temporal modes at the same time

Detailed Steps	Computational complexity
Assemble correlation matrix	$o(N \cdot {N_t}^2 \cdot N_x)$
Extract modes for θ_i	$o({N_t}^3 \cdot \sum_{i=1}^{N} \hat{K}^i)$
Extract φ^k	$o(\hat{K} \cdot (\sum_{i=1}^{N} \hat{K}^i)^3)$

However, for more complex problems, where N_x is very big, setting N_x^* less than N_t will cause a loss in accuracy. Thus, the approach above loses its advantage in efficiency. For such problems when extracting temporal modes, although the same snapshots $\{u^{\theta_i}(\chi_j, \cdot) - \hat{v}^{\theta_i}(\chi_j, \cdot), j = 1, 2, \cdots N_x^*\}$ are used, the correlation matrix M is defined as $M \in R_{N_t \times N_t}$ rather than $M \in R_{N_x^* \times N_x^*}$ and the temporal modes are just the eigenvectors of correlation matrix. The complexities for extracting temporal modes are listed as Table 3 and at this moment the total offline computational cost can also be reduced by extracting spatial and temporal modes separately.

Table 3: Computation complexity to extract temporal modes for complex problems

Detailed Steps	Computational complexity
Assemble correlation matrix	$o(N \cdot {N_t}^2 \cdot N_x^*)$
Extract modes for θ_i	$o({N_t}^3 \cdot \sum_{i=1}^{N} M^i)$
Extract φ^k	$o(M \cdot (\sum_{i=1}^{N} M^i)^3)$

D. RBF approximation

The final challenge for developing a ROM is to estimate the coefficients α^{θ} at any $\theta \in \Theta$ in (2) while retaining a good approximation accuracy. To deal with this, a RBF method is employed to interpolate α^{θ} using the sample data.

At first, direct training data for RBF should be provided: the samples' coefficients denoted by $\alpha^{\theta_i}(k,m)(i=1,2,\cdots N; k=1,2,\cdots K; m=1,2,\cdots M)$. Substitute shifted snapshots $u^{\theta_i}-v^{\theta_i}$ into (2),

$$u^{\theta_i}(x,t) \approx v^{\theta_i}(x,t) + \sum_{k=1}^{K}\sum_{m=1}^{M} \alpha^{\theta_i}(k,m)\varphi^k(x)\xi^m(t) \qquad (i=1,2,\cdots N) \tag{13}$$

and rewrite (13) in a simpler form as

$$u^{\theta_i} \approx v^{\theta_i} + \varphi\alpha^{\theta_i}\xi^T \tag{14}$$

At this time, a penalized least square approach [39] is adopted to determine α^{θ_i}: Define a residual as the difference between an observed solution and the fitted solution. α^{θ_i} should be determined to minimize the sum of squared residuals in the least-squares sense:

$$\min_{\alpha^{\theta_i}\in R_{K\times M}} (||u^{\theta_i} - v^{\theta_i} - \varphi\alpha^{\theta_i}\xi^T||_2^2 + \mu||\alpha^{\theta_i}||_2^2) \tag{15}$$

where $\mu \geq 0$ is a penalized parameter to avoid overfitting and $||\cdot||_2$ represents the 2-norm for a matrix with the definition $||A||_2 = tr(A^TA)$.

Coefficients α^{θ_i} corresponding to θ_i can be computed by

$$(\varphi^T\varphi)\alpha^{\theta_i}(\xi^T\xi) + \mu\alpha^{\theta_i} = \varphi^T(u^i - v^i)\xi \tag{16}$$

With the computed data α^{θ_i}, RBF approximates $\alpha^{\theta}(k,m)$ as follows:

$$\alpha^{\theta}(k,m) \approx \sum_{i=1}^{N} \gamma_i^{km}\phi(||\theta-\theta_i||) \qquad (k=1,2,\cdots K; m=1,2,\cdots M) \tag{17}$$

where $\phi(r)$ is the chosen RBF kernel function and γ_i^{km} denotes the undetermined coefficients, which are independent on θ. $||\cdot||$ denotes the 2-norm for a vector. By substituting θ_j and relating α^{θ_j} into (17), $\gamma_i^{km}(i=1,2,\cdots N; k=1,2,\cdots K; m=1,2,\cdots M)$ are computed though

$$\begin{gathered}\alpha^{\theta_j}(k,m) \approx \sum_{i=1}^{N} \gamma_i^{km}\phi(||\theta_j-\theta_i||) \\ (i=1,2,\cdots N; k=1,2,\cdots K; m=1,2,\cdots M)\end{gathered} \tag{18}$$

Finally from $\gamma_i^{km}(i = 1, 2, \cdots N; k = 1, 2, \cdots K; m = 1, 2, \cdots M)$, α can be recovered for arbitrary θ value in parameter space though (17). There are variety of kernel functions and in this work a Gaussian radial basis function[40] is used, defined as

$$\phi(r) = \exp\left(-(\sigma r)^2\right) \tag{19}$$

with a parameter $\sigma > 0$. Define a matrix as:

$$\Phi_{i,j} = \phi(||\theta_i - \theta_j||) \quad (i, j = 1, 2, \cdots N) \tag{20}$$

When σ is very large, the matrix Φ defined in (20) to be used in (18), will be ill-conditioning. As a result, the approximation will break down. On the contrary, if σ is very small, the approximation accuracy gets affected.

This phenomenon would become particularly problematic when the number of candidates increases or an unstructured sampling pattern in parameter space is used. To address this, RBF-QR proposed in [36] is adopted in this paper. In this approach, one eliminates the effect of σ by QR decomposition and restores high accuracy even for small σ.

Although derivation processes of RBF-QR are similar for different kernel functions or different number of design parameters, it is difficult to give a general description. In this paper, only derivation process for Gauss kernel function and two design parameters $\theta = (\theta_1, \theta_2)$ are presented. Gauss RBF centered on θ_0 can be expanded as

$$\phi(r) = \phi(||\theta - \theta_0||) = e^{\left(-\sigma \cdot ||\theta - \theta_0||^2\right)} = e^{\left(\sigma \cdot \theta^2\right)} \cdot e^{\left(\sigma \cdot {\theta_0}^2\right)} \cdot e^{(2\sigma \cdot \theta_0 \cdot \theta)} \tag{21}$$

The term $e^{(2\sigma \cdot \theta_0 \cdot \theta)}$ can be expanded by Taylor series $e^{(2\sigma \cdot \theta_0 \cdot \theta)} = \sum_{j=0}^{\infty} \frac{(2\sigma^2)^j}{j!}(\theta \cdot \theta_0)^j$. And $\phi(||\theta - \theta_0||)$ is expanded as

$$\phi(||\theta - \theta_0||) = e^{\left(\sigma \cdot \theta^2\right)} \cdot e^{\left(\sigma \cdot {\theta_0}^2\right)} \cdot \sum_{j=0}^{\infty} \left(\frac{(2\sigma^2)^j}{j!}(\theta \cdot \theta_0)^j\right) \tag{22}$$

When further expanding (22), [36] points out that the monomials may be nearly linearly dependent with each other. To deal with this, two approaches are adopted: Firstly, transform the expression form orthogonal coordinate (θ_1, θ_2) to polar coordinate (r, α) and the centered point in the polar coordinate is denoted by (r_0, α_0):

$$\phi(||\theta - \theta_0||) = 2e^{-\sigma^2 r_0^2} \cdot e^{-\sigma^2 r^2} \cdot (A + B) \tag{23}$$

where

$$\begin{cases} A = \sum_{i=0}^{\infty} \left(\sigma^2 r_0 r\right)^{2i} \sum_{j=0}^{i} \frac{1}{2} \frac{1}{(i+j)!(i-j)!} G(2j) \\ B = \sum_{i=0}^{\infty} \left(\sigma^2 r_0 r\right)^{2i+1} \sum_{j=0}^{i} \left(\frac{1}{2} \frac{1}{(i+1+j)!(i-j)!} G(2j+1) \right) \end{cases} \tag{24}$$

and function G is defined as $G(m) = \cos(m \cdot \alpha_0)\cos(m \cdot \alpha) + \sin(m \cdot \alpha_0)\sin(m \cdot \alpha)$.

Secondly, express r's powers with Chebyshev splines $T_i(r)$, see [41],

$$r^{2j+p} = \sum_{i=0}^{\infty} h_l \cdot T_{2l+p}(r) \qquad p, l = 0, 1, \cdots, \infty \tag{25}$$

and $\phi(\theta)$ is finally expanded as

$$\phi(||\theta - \theta_0||) = \sum_{j=0}^{\infty} \sum_{m=0}^{\frac{j-p}{2}} d_{j,m} \cdot c_{j,m}(\theta_0) \cdot T^c_{j,m}(\theta) + \sum_{j=0}^{\infty} \sum_{m=0}^{1-p} d_{j,m} \cdot s_{j,m}(\theta_0) \cdot T^s_{j,m}(\theta) \tag{26}$$

Where

$$\begin{cases} d_{j,m} = \frac{\sigma^{2j}}{2^{j-2m-1}(\frac{j+2m+p}{2})!\left(\frac{j-2m-p}{2}\right)!} \\ T^c_{j,m}(\theta_0) = e^{-\sigma^2 r^2} r^{2m} T_{j-2m}(r) cos\left((2m+p)\alpha\right) \\ T^s_{j,m}(\theta_0) = e^{-\sigma^2 r^2} r^{2m} T_{j-2m}(r) sin\left((2m+p)\alpha\right) \\ c_{j,m}(\theta_0) = f(2m+p)t(j-2m)e^{-\sigma^2 {r_0}^2} {r_0}^j \cos\left((2m+p)\alpha_0\right) {}_1F_2(a_{jm}, b_{jm}, \sigma^4 {r_0}^2) \\ s_{j,m}(\theta_0) = f(2m+p)t(j-2m)e^{-\sigma^2 {r_0}^2} {r_0}^j \sin\left((2m+p)\alpha_0\right) {}_1F_2(a_{jm}, b_{jm}, \sigma^4 {r_0}^2) \end{cases} \tag{27}$$

$a_{jm} = \frac{j-2m+p+1}{2}$,$b_{jm} = [j-2m+1, \frac{j+2m+p+1}{2}]$, and ${}_1F_2$ denotes hypergeometric function. Function f(x) and t(x) is defined as:

$$f(x) = \begin{cases} 1 & x = 0, \\ 2 & x > 0. \end{cases}$$

,

$$t(x) = \begin{cases} 0.5 & x = 0, \\ 2 & x > 0. \end{cases}$$

The expansion in (26) can be written into a matrix form:

$$\phi(r) = C \cdot D \cdot T(r) \tag{28}$$

where $T(r)$ is a matrix consisting of Chebyshev splines, D is a diagonal matrix of σ's increasing powers $o(\sigma^{2j})(j = 1, 2, \cdots \infty)$, and C is a matrix with elements $o(1)$. And the σ's effect mainly stems

from the diagonal matrix D. [36] proposes a new form of RBF, by adopting QR decomposition on C and multiplying $\phi(r)$ with $D_N^{-1}R_N^{-1}Q^T$:

$$\begin{aligned}\psi(r) &= D_N^{-1} \cdot R_N^{-1} \cdot Q^T \cdot \phi(r) \\ &= D_N^{-1} \cdot R_N^{-1} \cdot Q^T \cdot Q \cdot R \cdot D \cdot T(r) \\ &= D_N^{-1} \cdot R_N^{-1} \cdot R \cdot D \cdot T(r)\end{aligned} \tag{29}$$

D_N^{-1} and R_N^{-1} is the first $N \times N$ sub-matrix of D and R respectively. It is predicted in [36] that the matrix $D_N^{-1} \cdot R_N^{-1} \cdot R \cdot D$ is less senstive to σ, leading to the fact that $\psi(r)$ is more stable than $\phi(r)$.

Section IV further compares the performance of the GNIROM with RBF-QR and that of the standard RBF through a specific problem.

E. Overview

Once the spatial and temporal basis functions are determined and the coefficients α^θ are expressed through the expansion of RBF, the solutions at any θ in the parameter space can be approximated through (2). Algorithm 1 outlines the steps for the NIROM. It is also worth noting that, for problems with multiple state variables the NIROM is used for each variable independently, which may cause different mode number achieved. However, in intrusive ROMs, it is common to consider a vector mode (u, v, p) all together. [10] compares the two methods in NIROMs consisting of POD and RBF in SECTION 2.2 and concludes that: "Although costlier, the approach considering the state variables independently allows for a more detailed exploitation of the information contained in the snapshots, with the space of representable solutions being enlarged." And for the NIROMs, if only the relative reduced basis can predict the flow field accurately for each state variable, no physical and mathematical concerns would raise from a forced decoupling in governing equation. Thus in this paper, it is preferred to adopt the NIROM for different state variable independently.

Algorithm 1 Reduced-order method approach

1: Generate snapshots with an appropriate numerical solver.

2: **for** each $\theta_i (i = 1, 2, \cdots, N)$ **do**

3: set a threshold ε_1 and apply POD to extract a set of temporal modes $\varphi^{k,i}$ and spatial modes $\xi^{m,i}$;

4: **end for**

5: Collect all the temporal and spatial basis functions, and then apply SVD to determine the final required reduced basis $\varphi^k (k = 1, 2, \cdots K)$ and $\xi^m (m = 1, 2, \cdots M)$ with another threshold ε_2.

6: Utilize the reduced basis, snapshots and appropriate RBF $\phi(r)$ to compute γ_i^{km}.

7: Detemine coefficients α^θ at any condition using (17) and then approximate the solution with (2).

III. A greedy approach for model development

The remaining problem in the algorithm discussed so far is the sampling. In [13], samples are selected in a simple manner. For further improvements in sample selection, this paper conducts application of a greedy algorithm to propose adaptive sampling. The core idea of the greedy algorithm is to select samples adaptively and iteratively by locating the new sample at each iteration where the estimated error in the reduced model reaches the maximal in the predetermined parameter space.

A. Initial Sampling

To start the adaptive sampling, N^0 initial candidates should be given. As the Greedy approach selects samples locally, the sampling selection is very sensitive to the initialization. Considering that Greedy approach proposed in this Section aims to provide optimum candidates at each step, fewer initial points are selected, more efficient of GNIROM performs. On the other hand, the NIROM proposed in Section III performs well in interpolation, but not very well in extrapolation (which means that the point to be approximated is outside the range in the parameter space which the samples span). Therefore, it is suggested to select all the corner points in the parameter space as initial points, to ensure a good start of the iteration.

B. Error estimator

From the ROM presented above, the error estimator at the iter-th iterative step can be directly proposed as

$$\Delta(iter) = ||u^\theta - \hat{u}^\theta||_2 = ||u^\theta - v^\theta - \varphi^e \alpha^\theta \xi^{eT}||_2 \tag{30}$$

where φ^e and ξ^e represents basis functions at the current iterative step. Here $||\cdot||_2$ still denotes the squared norm for a matrix with $||A||_2 = tr(A^T A)$. And θ in parameter space which makes $||(u^\theta - v^\theta - \varphi^e \alpha^\theta \xi^{eT})||_2$ reach the maximal is selected as the new candidate, denoted by θ_s. However, as u^θ and v^θ is undetermined except when the numerical solver is used, it would be too expensive to determine θ_s directly with this error estimator (30).

To develop an efficient alternative, recall the process of computing α^θ using the least square as well as RBF approximation in Section II. To recover the undetermined matrix γ_i^{km}, α^{θ_l} is substituted by (16) into (17) as

$$\begin{aligned} \alpha^{\theta_l}(k,m) &= \sum_{i=1}^{N^e} \gamma_i^{km} \left(\phi \left(\frac{|\theta_l - \theta_i|}{\sigma} \right) \right) \\ (l &= 1, 2, \cdots N^e; k = 1, 2, \cdots K; m = 1, 2, \cdots M) \end{aligned} \tag{31}$$

Where $N^e = iter + N^0$ denotes the candidate number at the current step. From (31), γ_i^{km} is easy to be solved, and then α^θ can be computed for any θ using (17). Finally, the approximated solution $\hat{u}^\theta$ is recovered through (2) and α^θ. Clearly, the approximated solution for any θ is recovered similarly to the interpolation of snapshots in parameter space. The difference is that the former interpolates the coefficients $\alpha^\theta(k,m)$ of each candidate, instead of interpolating the snapshots directly. Therefore it is suggested to take the point where the mean distance between α^θ and α^{θ_l} is the largest as the new candidate.

The distance between $\alpha^{\theta_l}(l = 1, 2, \cdots, N^e)$ and α^θ can be expressed by

$$\begin{aligned} \alpha^\theta(k,m) - \alpha^{\theta_l}(k,m) &= \sum_{i=1}^{N^e} \gamma_i^{km} (\phi(\frac{|\theta - \theta_i|}{\sigma}) - \phi(\frac{|\theta_l - \theta_i|}{\sigma})) \\ (l &= 1, 2, \cdots, N^e, k = 1, 2, \cdots K; m = 1, 2, \cdots M) \end{aligned} \tag{32}$$

Thus an alternative for error estimator on the form in 2-norm sense based on α distance is given as:

$$\begin{aligned}\Delta_1(iter) &= \frac{\sum_{l=1}^{N^e} || \sum_{k=1}^{K} \sum_{m=1}^{M} [(\alpha^{\theta}(k,m) - \alpha^{\theta_l}(k,m))\xi^{ek}(x)\xi^{em}(t)]||_2}{N^e} \\ &= \frac{\sum_{l=1}^{N^e} || \sum_{k=1}^{K} \sum_{m=1}^{M} [(\sum_{i=1}^{N^e} \gamma_i^{km}(\phi(\frac{|\theta-\theta_i|}{\sigma}) - \phi(\frac{|\theta_l-\theta_i|}{\sigma})))\varphi^{ek}(x)\xi^{em}(t)]||_2}{N^e} \\ &= \frac{\sum_{l=1}^{N^e} ||\sum_{i=1}^{N^e}(\phi(\frac{|\theta-\theta_i|}{\sigma}) - \phi(\frac{|\theta_l-\theta_i|}{\sigma}))(\sum_{k=1}^{K} \sum_{m=1}^{M} (\gamma_i^{km})\varphi^{ek}(x)\xi^{em}(t))||_2}{N^e}\end{aligned} \tag{33}$$

where $\sum_{k=1}^{K} \sum_{m=1}^{M} (\gamma_i^{km}\varphi^{ek}(x)\xi^{em}(t))$ is independent on θ and considered as coefficients.

However, if θ is considered as a continuous variable vector, the Greedy method would be impractical for high cost of direct computation (33). To solve this, a series of discrete points $\theta_{ii}^*(ii = 1, 2, \cdots N^*)$ are selected in the parameter space, from which the new candidate can be identified. Subsequently, with Δ_1 for each sample point computed, the new candidate would be easily obtained by searching the one with the maximum value of Δ_1.

To further reduce the computational cost when choosing the candidate from the N^* points, a simpler error estimator ignoring basis function is also proposed as:

$$\Delta_2(iter) = \sqrt{\frac{\sum_{l=1}^{N^e} ||\alpha^{\theta}(k,m) - \alpha^{\theta_l}(k,m)||_2^2}{N^e}} = \sqrt{\frac{\sum_{l=1}^{N^e} ||\sum_{i=1}^{N^e} \gamma_i^{km}(\phi(\frac{|\theta-\theta_i|}{\sigma}) - \phi(\frac{|\theta_l-\theta_i|}{\sigma}))||_2^2}{N^e}} \tag{34}$$

Δ_2 is just the mean error of α distance. Strictly speaking, Δ_1 and Δ_2 don't belong to error estimators but could be an estimator for us to find points with large errors. The computational complexity of Δ_1 is $N^{*2} \times K \times M$ while that of Δ_2 is N^{*2}, predicting that computational cost for both error estimators are very small. And the two error estimators are compared and discussed in the diffusion problem in Section IV.

For problems with multi-state variables $u(q)(q = 1, 2, \cdots Q)$, GNIROM is adopted to each state variable independently and the error estimator (33) or (34) is computed for each state variable, denoted by $\Delta_1(q, iter)$ or $\Delta_2(q, iter)$. The key point to deal with this issue is to propose an error estimator which can consider all the state variables and provide a new candidate at each iterative step. Drawing the ideas form multi-objective optimization, priority weightings $w(q)(q = 1, 2, \cdots Q)$ are set for each state variable, and the error estimator for this problem denoted by Δ_{mul} is defined

as

$$\Delta_{mul}(iter) = \sqrt{\sum_{q=1}^{Q} w(q) * \Delta_i(q, iter)^2} \qquad (i = 1 \quad or \quad 2) \tag{35}$$

Generally, the chosen of $w(q)$ is subjective and problem dependent.

C. Termination criterion

To terminate the growth of the reduced basis, there are various available strategies. For example, the adaptive sampling can terminate the iteration by just setting the number of iterative steps to a constant, or basing on the decrease of the approximation error below a certain threshold. Under the goal of reducing the overall approximation error, the present criterion includes the following two steps:

At first, an appropriate step number N_{pre} is set to run the GNIROM's iteration. As Greedy Approach aims to reduce the offline computational cost, the approximation accuracy is the key factor which determines whether to terminate the sampling. However, parameters set in the ROM, such as ε_1, ε_2 in Section II (C) and σ in equation (19), may not be appropriate which will cause more unnecessary iterative steps needed to meet the accuracy requirements, especially when the iterations have proceeded for 10 or more steps. Thus, it is preferred to set a certain value N_{pre} and then check whether the adaptive sampling goes well. If not, the thresholds will be changed at N_{pre}-th step. To make the approach efficient, the value N_{pre} should not be too small or too big. 10 for 1-parameter problem and 20 for 2-parameter problems are empirically suggested. And then the iteration continues.

Secondly, iteration proceeds until when the approximation error between the solver and G-NIROM is less than a preset value. And once this assumption is invalid, the GNIROM terminates. As to which kind of error to focus on, it is preferred in this paper to use the 2-norm of errors on a set of test points as the error indicator. However, in concrete application, this often results in a substantial computational cost. As an alternative, the mean error on the determined new candidates is chosen as the indicator.

D. Summery of GNIROM Approach

As presented above, two alternatives for error indicators are proposed to decide sampling, however, both are heuristic and it is difficult to provide a rigorous bound for them. In numerical cases of section IV, the ROM performs well with candidates determined by the indicator, demonstrating that those may not be optimal but at least very good samples for building reduced basis.

By adopting the error indicator as well as the criterion for termination, the whole GNIROM scheme can be expressed and summarized in Algorithm 2. As the first N_{pre} steps are set to check the thresholds, this process is not mentioned in Algorithm 2.

Algorithm 2 GNIROM Scheme

1: Choose initial samples as a start.

2: **for** $iter = 1$; $iter++$ **do**

3: Apply two-level POD to extract reduced basis ξ^e and φ^e, and utilize RBF to compute γ_i^{km}.

4: Use the error indicator to determine the new candidate in the whole parameter space.

5: Add the determined candidate to the existing snapshots.

6: Once the 2-norm of errors on a set of test points is less than a preset value, terminate the Greedy sampling iteration.

7: **end for**

8: Use the ROM in section II to extract basis functions from the snapshots determined above and approximate required solutions .

IV. Numerical test-case

In the following, the efficiency and accuracy of GNIROM are demonstrated through two numerical cases on 4 cores of a workstation running at 3.50GHZ. Although both validation cases belong to problems of fluid dynamics, it is worth emphasizing that the proposed algorithm is generic and applicable to general problems.

A. Two-dimensional heat diffusion

Consider the time-dependent diffusion equation with only one parameter:

$$\frac{\partial u}{\partial t} = v\triangle u \qquad in \qquad \Omega = [0,1]^2, t \in [0,T] \tag{36}$$

where T=5.0s, and viscosity coefficient $\upsilon = 1$. The single parameter is located in the boundary condition:

On the upper boundary,

$$g^{\theta}(x,t) = sin(\frac{\pi t}{2T})(1 - \theta sin(2\pi(x - 0.5))) + \frac{\theta t}{T} \tag{37}$$

and on the other three boundaries,

$$g^{\theta}(x,t) = 0 \tag{38}$$

where $\theta \in [0.5, 1.5]$. Initial solutions are set 0 everywhere in $\Omega = [0,1]^2$: $u_0^{\theta}(x) = 0$.

The problem is solved using a second order finite difference method with a uniform spatial grid of 41×41 points. The time-step dt is set 0.004 seconds and thus $N_t = 1251$. $N_t^* = 101$ points are selected uniformly in [0,T] to generate spatial snapshots, and a coarse spatial mesh of 11×11 points is used to generate temporal snapshots.

The parameters in the NIROM like ε_1 in (11) and σ in (19), are important for the ROM's approximation. However, when the parameters' values are adjusted at the initial phase of GNIROM, those values may be not appropriate with more candidates added. As a result, before adopting GNIROM, the paper applies the NIROM in Section II with uniform sampling (25 samples spread uniformly in the parameter space in this case) at first and adjusts the parameters' values to make the ROM more efficient. The adjusted parameters are listed: the thresholds ε_1 in (11) to recover temporal and spatial modes are both set 10^{-3}; when applying the SVD on these different sets of modes in Section II (C), the threshold ε_2 for temporal modes is set to 10^{-3}, and ε_2 for spatial modes equals to 10^{-4}; gaussian radial basis functions with $\sigma = 141.4$ in (19) is adopted as the kernel function. When GNIROM is used, those parameters are fixed to the same values.

To evaluate the accuracy, $N^* = 26$ test points θ^* locate uniformly in the parameter space and three different errors between the approximated solutions $\hat{u}^{\theta}$ and direct solutons u^{θ} are defined as

$$\text{Maximal error}: \quad Error1 = \max_{t_n \in [0,T]} ||\hat{u}^{\theta}(\cdot, t_n) - u^{\theta}(\cdot, t_n)||_{\Omega} \tag{39}$$

$$\text{Mean errors at } T \geq t \geq 0: \quad Error2 = \frac{1}{N_t} \sum_{n=1}^{N_t} ||\hat{u}^{\theta}(\cdot, t_n) - u^{\theta}(\cdot, t_n)||_{\Omega} \tag{40}$$

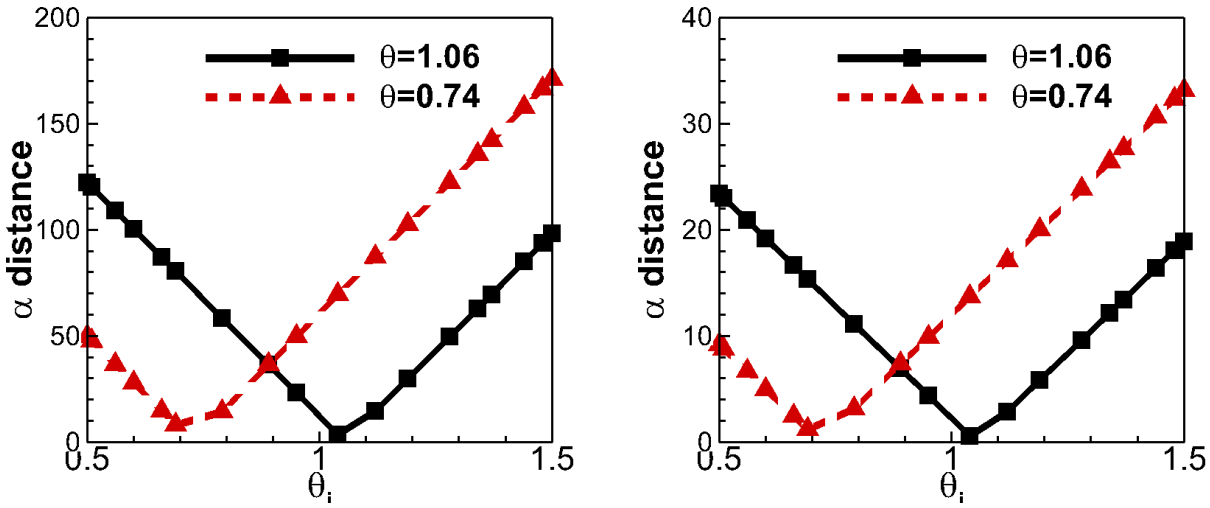
$$\text{Mean error at t=1s}: \quad Error3 = ||\hat{u}^{\theta}(\cdot, t=1) - u^{\theta}(\cdot, t=1)||_{\Omega} \tag{41}$$

The norm $||\cdot||_{\Omega}$ is defined as $||f||_{\Omega} = \sqrt{(f,f)}$ and the inner product $(\cdot,\cdot)$ is defined as $(f,g) = \int_{x\in\Omega} f(x)g(x)dx$. For fixed θ and time instant t_n, $\hat{u}^{\theta}(\cdot,t_n) = u^{\theta}_{t=t_n}$ is a vector of spatial coordinates $x \in \Omega$.

The greedy approach is initiated from two candidates $\theta = 0.5, 1.5$ as stated in Section III, and at each iteration it provides a new one. In Section III, two error estimators are presented in (33) and (34), and the GNIROM's results with the two estimators are compared. As both error estimators base on the distance between α^{θ} and $\alpha^{\theta_i}(i = 1, 2, \cdots, N^e)$, plots of $|\alpha^{\theta}(k,m) - \alpha^{\theta_i}(k,m)|$ for $\theta = 1.06, 0.74$ are made when k=1,m=1 and k=2,m=2 as presented in Figure 1. All the solutions, such as θ_i, $\alpha^{\theta}(k,m)$ and $\alpha^{\theta}(k,m)$, come from the GNIROM's solution where Δ_1 in (34) acts as the error estimator. From the figures, when θ_i is closer to θ, the distance is smaller. And if k or m differs and Δ_2 is used, the plot is similar and the trend is the same. From the Table 4, the two error estimators provide the same candidates before 8 iterations, but the subsequent chosen candidates are totally different. To judge whether error estimator performs better, the error distributions of test points for the two error estimators are presented at the same iterative step in Figure 2 and Figure 3. And the errors for test points are computed through eqn. (41). Blue vertical solid lines mark the candidates at the current iteration and red vertical dashed lines mark the new candidates at the next iteration. As the candidates are the same before 8 iterations, there is no need to compare error distributions before 8 iterations. Figure 3 shows that Δ_2 provides good samples when the step order is less than 13, as the new candidates are located at one of the peaks in the relative error distribution of test points. But the candidates determined by Δ_2 at 13th, 14th and 15th step are not appropriate, as the optimal candidates, of which the errors reach the maximal, are most likely located at $0.5 < \theta < 0.6$ by observing the error distributions. But Δ_1 always find a point which is very close to the "optimal" candidate, as presented in Figure 2. On the other hand, although compared to Δ_2, computing Δ_1 for discrete points is more time-consuming, the timing of Δ_1 is also very small, as stated in Section III (B). Going forward, the error estimator Δ_1 in (34) is adopted to drive the greedy selection in θ and only solutions with Δ_1 are presented.

Table 4: θ of chosen candidates at each iteration by the two error estimators

Iteration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Δ_1	0.89	1.19	0.69	1.34	1.04	0.79	0.6	1.48	0.51	1.44	1.28	0.95	0.56	1.12	1.37
Δ_2	0.89	1.19	0.69	1.34	1.04	0.79	0.6	0.96	1.11	1.26	1.41	1.47	0.63	1.49	0.61

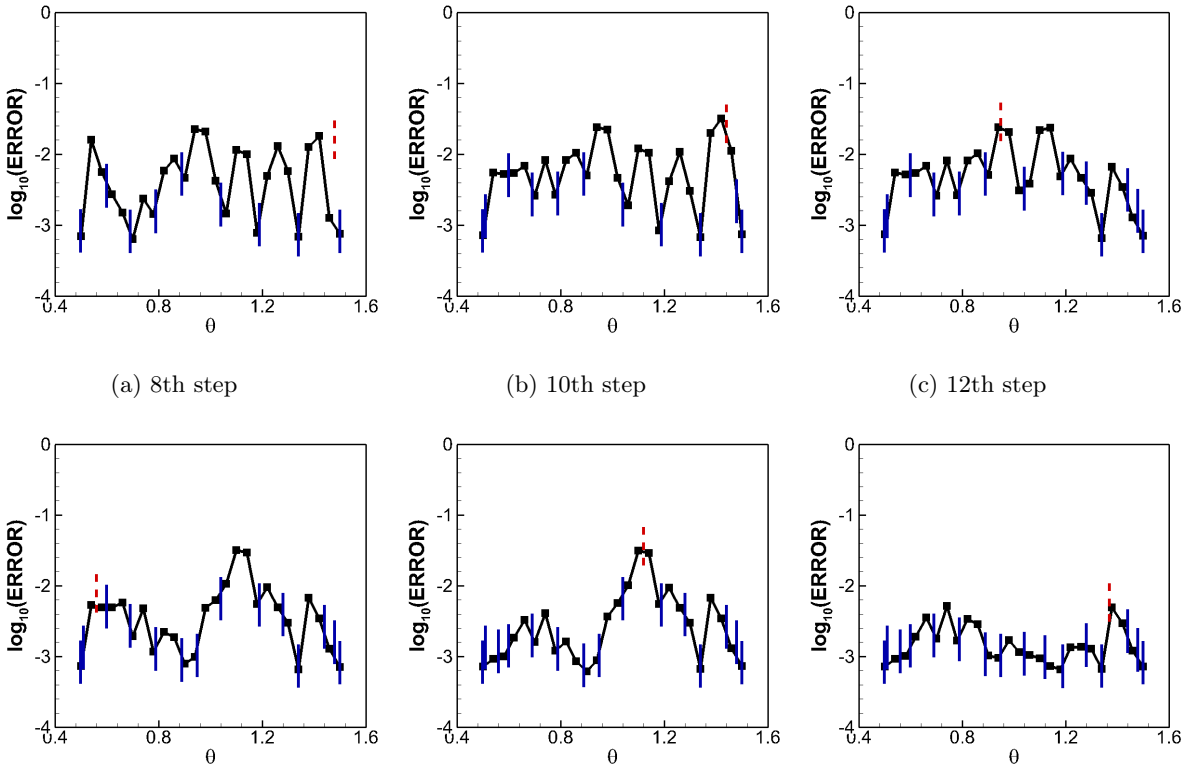


(a) k=1,m=1 (b) k=2,m=2

Fig. 1: Plot of α distance $|\alpha^{\theta}(k,m) - \alpha^{\theta_i}(k,m)|$ for $\theta = 1.06, 0.74$

After the error estimator determined, the adaptive sampling finally terminates at the 16th iteration with the termination criterion stated in Section III (C). In the above approximation, RBF-QR is used in the GNIROM approach. To validate RBF-QR's advantange over standard RBF, solutions of standard RBF are also presented as a comparison. Table 5 lists the samples determined in the iterative approach, using the standard RBF approximation as well as the RBF-QR technique and it can be noted that the candidates selected by the two approach are different, in particular as the iteration count increases. The difference is explained by the fact that the kernel function of RBF is introduced into the error estimator of adaptive sampling. When the standard RBF is replaced by RBF-QR, the error estimator varies, leading to a totally different sampling.

Figure 4 shows the trend of Root Mean Square (RMS) for the test samples' errors (39-41) in the greedy scheme using both the standard RBF approximation and the improved RBF-QR. From



(a) 8th step (b) 10th step (c) 12th step

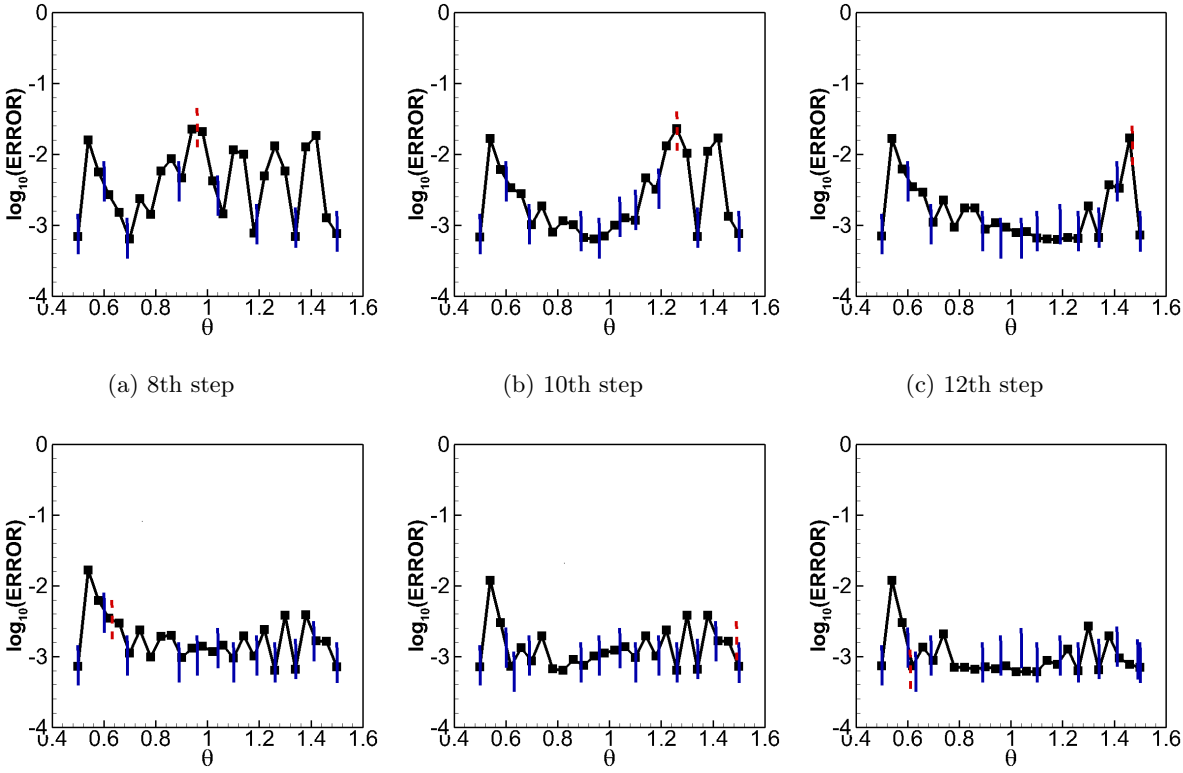
(d) 13th step (e) 14th step (f) 15th step

Fig. 2: Error distribution(eqn. (41)) in log10 scale between ROM and numerical solver for test points θ^* when error estimator Δ_1 in eqn (33) is used

Table 5: θ of chosen candidates at each iteration

Iteration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Standard RBF	1	1.25	0.75	1.44	1.19	0.94	0.7	0.55	1.47	1.41	1.49	1.43	1.39	1.36	1.34	1.33
RBF_QR	0.89	1.19	0.69	1.34	1.04	0.79	0.6	1.48	0.51	1.44	1.28	0.95	0.56	1.12	1.37	0.66

the figure, GNIROM using RBF-QR reduces the errors to values much less than that of standard RBF. Actually, the code of GNIROM reports an ill-condition problem when standard RBF is used. The reason is explained: Considering that it is inconvenient to adjust σ in the adaptive sampling, the value of σ is fixed in the iteration. But as standard RBF is very sensitive to σ, when the iteration proceeds, fixed σ may be inappropriate and will either cause ill-conditioning problem or affect approximation accuracy. In contrast, RBF-QR is less sensitive, and thus performs much better with fixed σ in the GNIROM. This comparison clearly confirms the importance of using improved



(a) 8th step (b) 10th step (c) 12th step

(d) 13th step (e) 14th step (f) 15th step

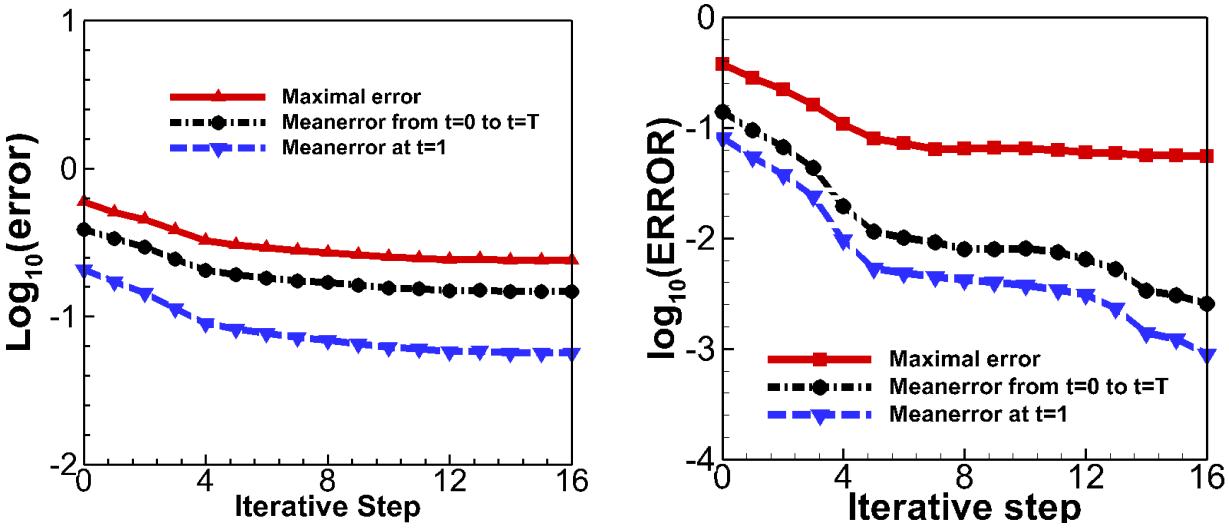
Fig. 3: Error distribution(eqn. (41)) in log10 scale between ROM and numerical solver for test points θ^* when error estimator Δ_2 in eqn (34) is used

technique to ensure the accuracy. Going forward, only solutions with RBF-QR are presented.

In Figure 5, the errors for each test sample all drop when iterative step n grows, demonstrating a global error reduction achieved.

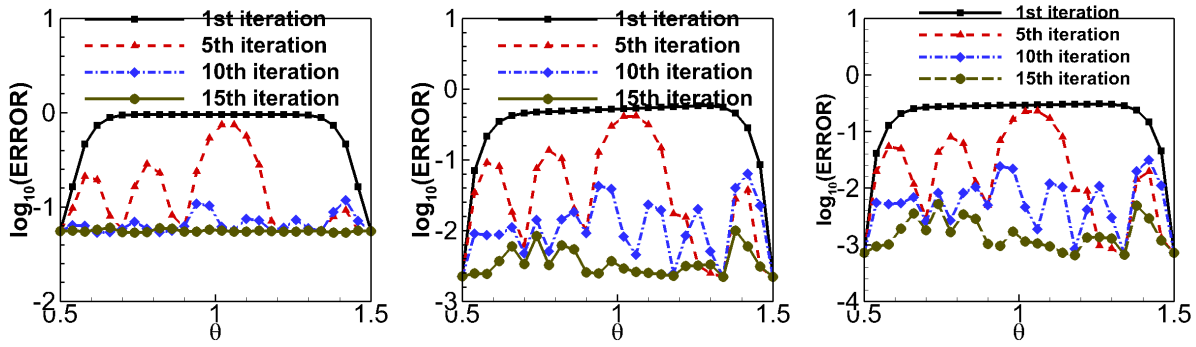
To further evaluate efficiency, approximations of the NIROM presented in Section II with 18 samples uniformly distributed in the parameter space is considered as a comparison. It is worth noting that this is the same number with candidates provided by Greedy sampling. The errors' RMS of both approximation at test points are compared in Table 6. From the table, the errors' RMS of the GNIROM are a little smaller than that of the ROM with uniform points. Due to the simplicity of the boundary condition and the single parameter, GNIROM does not present much advantage over the NIROM with uniform samples in this case.

With those 18 candidates, the GNIROM generates a total of $\sum_{i=1}^{N} M^i = 216$ temporal basis and $\sum_{i=1}^{N} K^i = 285$ spatial basis. After the SVD compression, 10 temporal basis $\xi^m (m = 1, 2, \cdots M)$ -



(a) Standard RBF (b) RBF-QR

Fig. 4: Trend of RMS for test points' errors(Eq. (39-41)) in the Greedy iteration process. In (a) the standard RBF is used in the reconstruction while in (b) the RBF-QR is used.



(a) Maximal error (b) Mean error at $T \geq t \geq 0$ (c) Mean error at t=1s

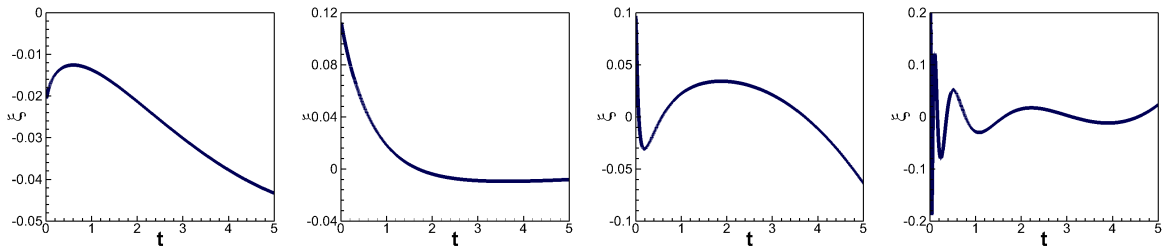
Fig. 5: Error distribution(Eq. (39-41)) in log10 scale between GNIROM and numerical solver for test points θ^*

see Figure 6 for an illustration of these, and 12 spatial ones $\varphi^k(k = 1, 2, \cdots K)$, illustrated in Figure 7, are retained. It can be observed that all temporal modes in Figure 6 change rapidly close to 0, consistent with the rapid changes of the solution during the initial temporal phase.

After the reduced basis built, it takes 0.12 seconds to approximate one point with those reduced basis while the numerical solver has to spend 251.97 seconds, illustrating that a great online benefit is achieved in computational cost with the proposed approach.

Table 6: Errors' comparison in log10 scale of GNIROM and the ROM with uniform sampling

Approach	Maximal error	Mean error at $T \geq t \geq 0$	Mean error at t=1s
GNIROM	-1.257	-2.592	-3.053
ROM with uniform snapshots	-1.257	-2.575	-3.046



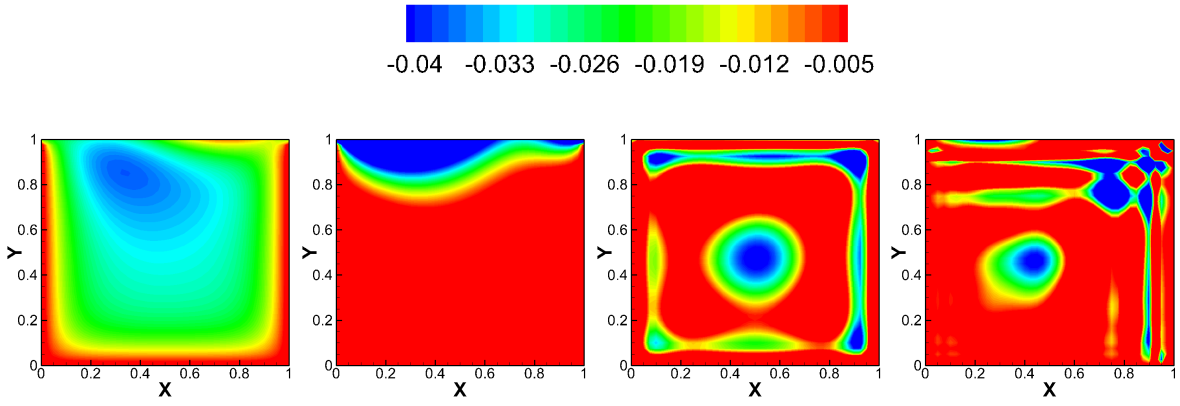
$m = 1$ $m = 2$ $m = 4$ $m = 10$

Fig. 6: The typical temporal modes

It can be seen from Figure 4 that the mean errors become very small after greedy sampling. However, it is premature to conclude that the GNIROM's approximation is accurate. To further evaluate this, the direct error in flow field's prediction across variations in θ requires further study. Considering the flow field at two points, neither of which belong to the candidates, one is $\theta = 0.55$ and the other is $\theta = 1.15$. Contours predicted by the numerical solver are presented in Figure 8 at four time instants and Figure 9 gives the error distributions $|u^{\theta}(x,t) - \hat{u}^{\theta}(x,t)|$ in the whole flow field between the direct solutions and those approximated by GNIROM. Observed from the figures, errors in the whole field are small, demonstrating a good accuracy of GNIROM when approximating the flow fields. By observing Figure 8 together with Figure 9, a high direct error is more likely to appear at the regions where there is a dramatic change in the state variable u. For example, in the regions close to boundary conditions, the relative errors are often very big.

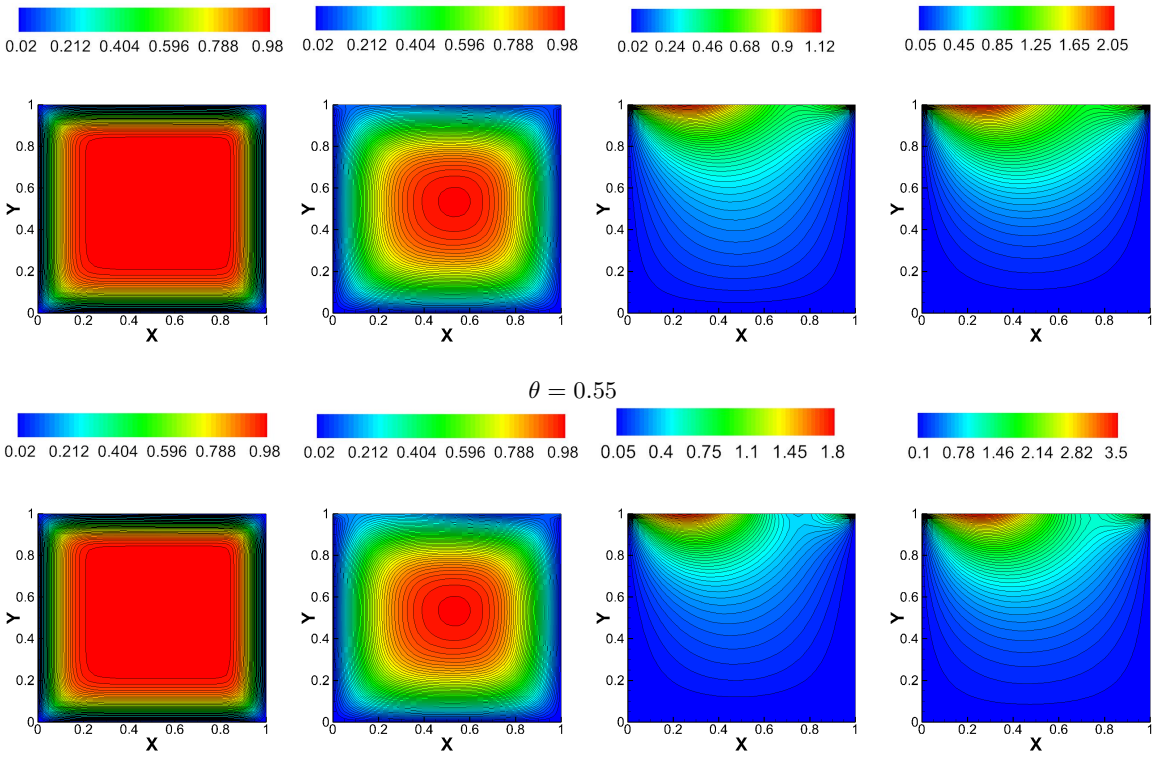
B. Lid-driven cavity flow

As a more complex problem in fluid dynamics, a two-dimensional driven cavity flow with two parameters θ_1 and θ_2 is considered as the second verification case. In this case, incompressible



$k = 1$ $k = 2$ $k = 6$ $k = 12$

Fig. 7: The typical spatial modes



$\theta = 0.55$

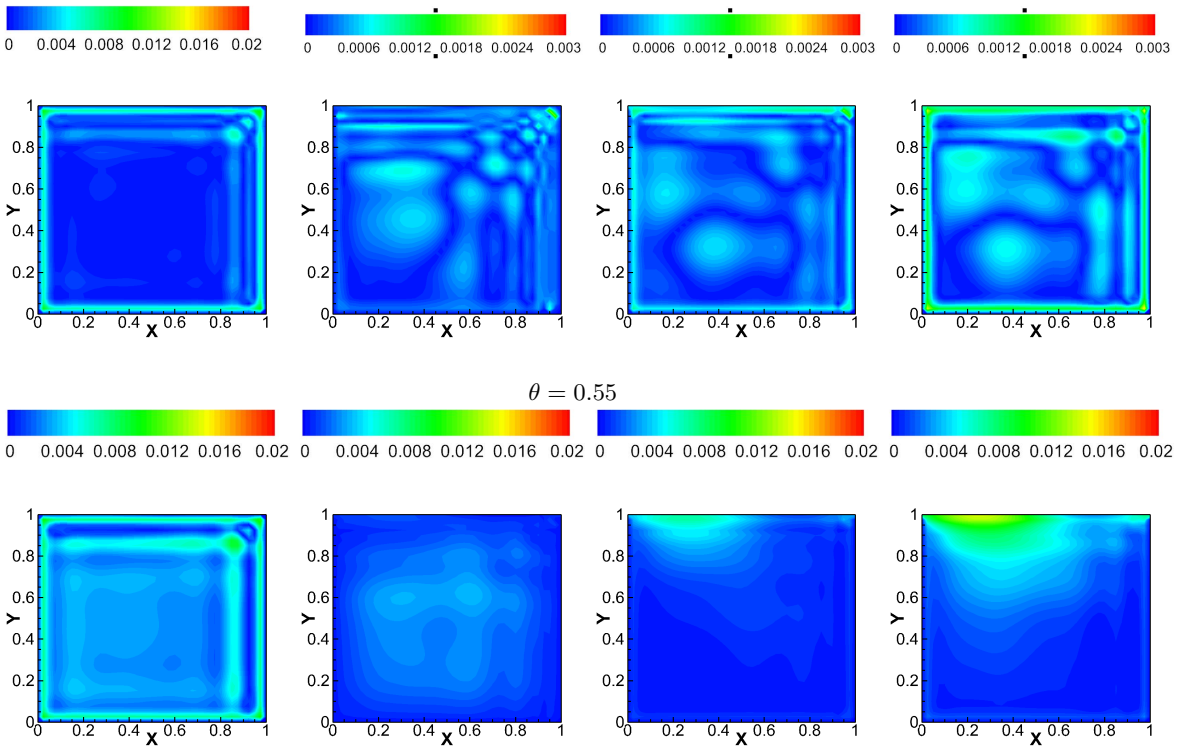
$\theta = 1.15$

(a)t=0.04s (b)t=0.2s (c)t=2s (d)t=5s

Fig. 8: Direct solutions for $\theta = 0.55, 1.15$

Navier-Stokes equations(excluding body forces) is considered:

$$\begin{cases} \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \\ \frac{\partial u}{\partial t} + u \cdot \frac{\partial u}{\partial x} + v \cdot \frac{\partial u}{\partial y} = -\frac{1}{\rho_0} \cdot \frac{\partial p}{\partial x} + \nu \Delta u \\ \frac{\partial v}{\partial t} + u \cdot \frac{\partial v}{\partial x} + v \cdot \frac{\partial v}{\partial y} = -\frac{1}{\rho_0} \cdot \frac{\partial p}{\partial y} + \nu \Delta v \end{cases} \tag{42}$$



(a)t=0.04s (b)t=0.2s (c)t=2s (d)t=5s

Fig. 9: Error distributions $\Delta u(x,t) = |u^{\theta}(x,t) - \hat{u}^{\theta}(x,t)|$ for $\theta = 0.55, 1.15$

where ρ_0 is a constant. Unlike diffusion problems, there exists three state variables: x-velocity(u), y-velocity(v), and pressure(p). Reynolds number equals to 100, thus determining the viscosity ν.

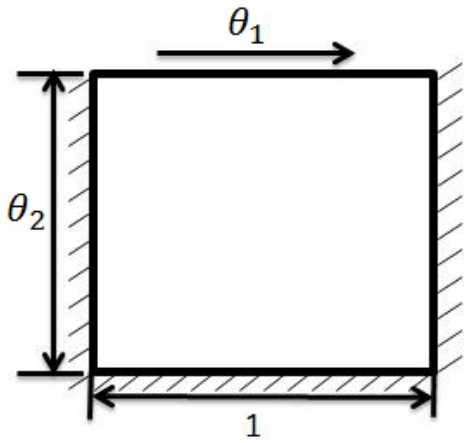


Fig. 10: Schematic diagram of computational domain

The computational domain is shown in Figure 10. θ_1 denotes the velocity of the top boundary, and velocities are 0 on the other three boundaries. θ_2 refers to the height of the cavity such that

the computational domain is $[0,1] \times [0,\theta_2]$. It is assumed that $\theta_1, \theta_2 \in [1,2]$. The initial velocities are 0 everywhere at the boundaries except at the top boundary. In numerical simulation, a second-order finite difference solver with a uniform spatial grid made of 41×41 points is used. The total simulation time T equals to 5.0s and time-step dt is set as 0.01s. In addition, a coarser spatial mesh of 11×11 points is generated to extract temporal snapshots and a uniform $N_t^* = 51$ point set in [0,T] is used to generate spatial snapshots.

As the height of the domain is chosen as one design parameter, the domain differs when θ_2 varies. To ensure the approximation accuracy, the length and width of each grid cell remains the same. However, the numbers of gird points are different, which make it difficult to generate snapshots. In this paper, when generating snapshot using the solution data, another grid with 41×41 uniformly spread in the domain even when the height of domain varies. As a result, the snapshots provided are the solutions at such grids, which can be easily interpolated through the solver data.

As stated in the heat diffusion problem, the NIROM with uniform samples (6×6 sample points in this case) are firstly used to adjust the parameters in GNIROM. It is worth noting that as the flow field characteristics and accuracy requirements for state variables are different, constants in the approach are given independently for each state variable: ε_1 in (11) to compute temporal and spatial modes are set $\{10^{-3}, 10^{-3}, 10^{-3}\}$ for u, v, p; ε_2 for the spatial modes in Section II (C) are given by $\{10^{-3}, 10^{-3}, 10^{-4}\}$, while ε_2 for the temporal modes are set $\{10^{-2}, 10^{-2}, 10^{-3}\}$ respectively; and in RBF σ in (19) is set as $\{35.4, 35.4, 0.707\}$ for the 3 variables separately.

1. Adaptive sampling

The sampling is initiated from four corner points in the parameter space $(\theta_1, \theta_2) = \{(1,1), (1,2), (2,1), (2,2)\}$. GNIROM terminates at 28th iterations and the selected samples are located at round spots in Figure 11.

11×11 points are selected as test points uniformly distributed in the parameter space $[1,2] \times [1,2]$. Figure 12 shows the trend of RMS for the test points' errors defined in (39)-(41) between the reduced order solution and by the numerical solver. With the iteration increasing, errors' RMS of the three state variables all gradually decrease and remain almost invariably at around 28th step. At this

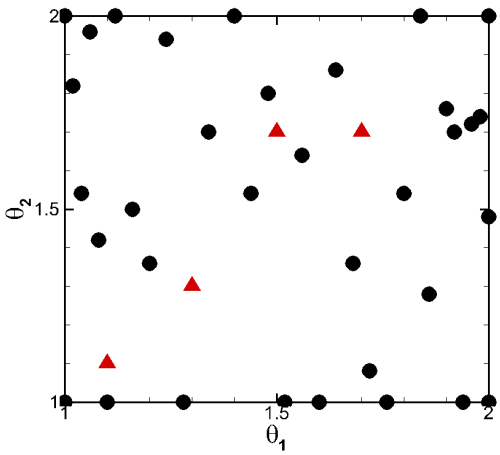
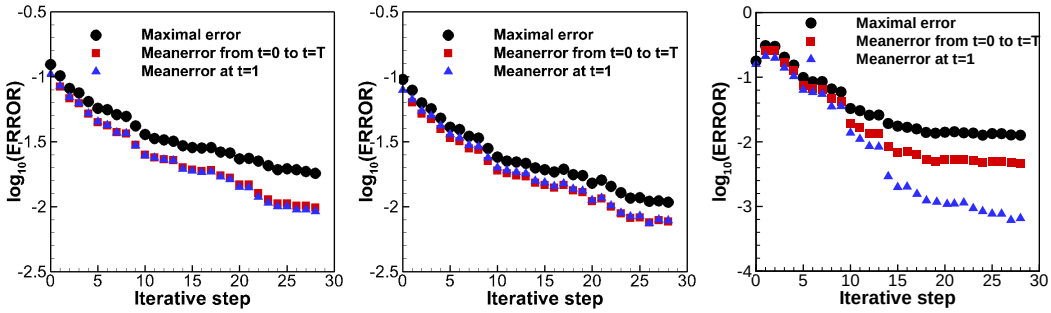


Fig. 11: Final parameter points after 28 iterations



(a) u (b) v (c) p

Fig. 12: Trend of RMS for test points' errors(Eq. (39-41)) in the Greedy iteration process.

time, mean errors for the velocity components are less than 10^{-2}, while those for the pressure are less than 10^{-3}.

Table 7: Errors' comparison of GNIROM and uniform sampling ROM in log10 scale

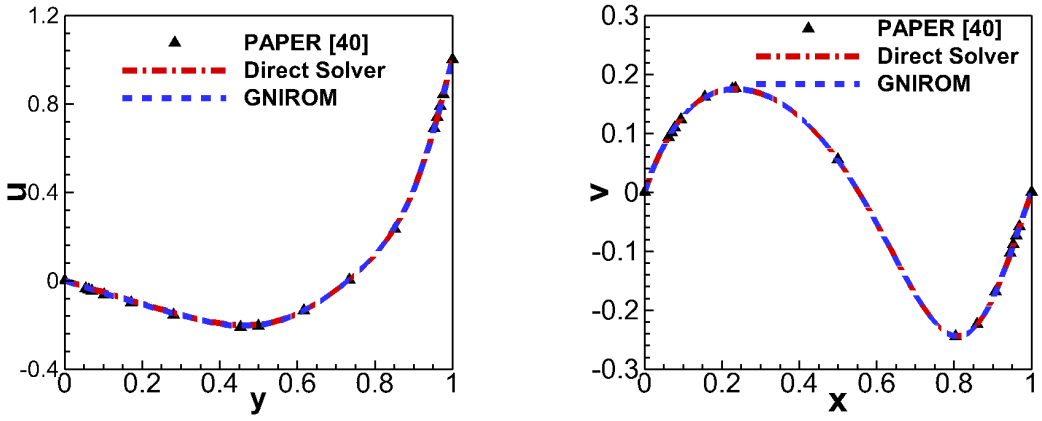
Approach	Maximal error			Mean error at $T \geq t \geq 0$			Mean error at t=1s		
	u	v	p	u	v	p	u	v	p
GNIROM	-1.744	-1.966	-1.892	-2.012	-2.121	-2.323	-2.044	-2.113	-3.108
ROM with uniform candidates	-1.231	-1.355	-1.825	-1.215	-1.368	-1.936	-1.216	-1.332	-2.423

Considering the ROM in Section II with 32 candidates spread uniformly in the parameter space, the errors' RMS at test points for the two different approaches are compared and listed in Table (7). For the same variable, the values of GNIROM system are substantially smaller than those computed

by the ROM with 32 candidates, showing that in this case the candidates determined by GNIROM have a notable advantage over the direct uniform sampling approach.

2. Fluid field approximation

After the sampling process, the scheme provides 7, 6 and 3 temporal modes as well as 27, 28 and 5 spatial modes for u, v and p, respectively. For each test sample, GNIROM only copes with $7\times27+6\times28+3\times5=372$ unknowns. While in this test-case with 3 state variables, the full model deals with a mesh of 1681 cells and 501 time intervals, meaning that it has to solve 2.52 million unknowns for each sample. Thus the DOFs of problem solving have been greatly reduced. Figure 13 compares velocity profiles of GNIROM with those by the direct solver when $\theta_1=1,\theta_2=1$, showing that the approximated profiles coincide very well with those of the solver as well as solutions in [42].



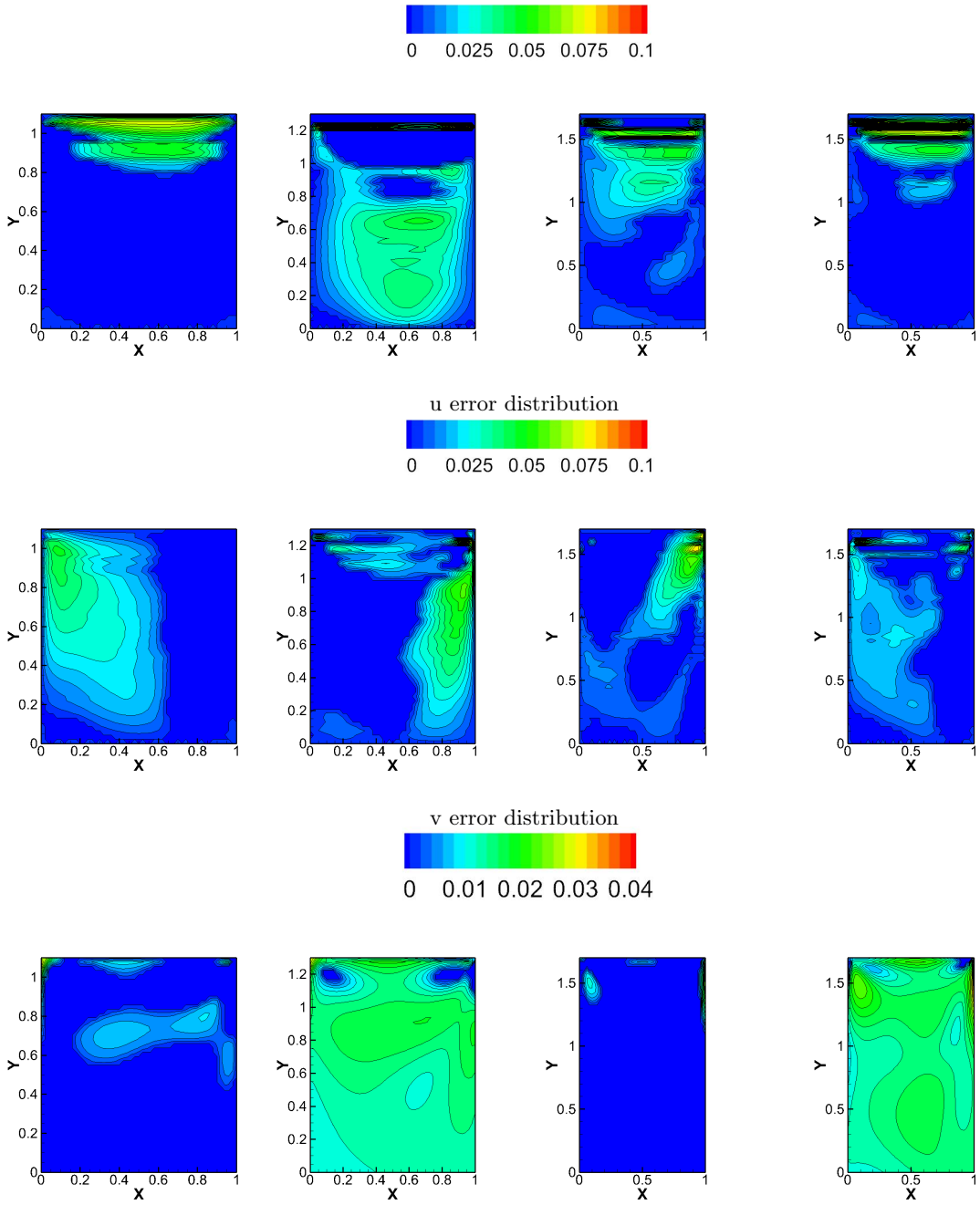
(a) u-velocity distribution along vertical line through the geometric center of the cavity

(b) v-velocity distribution along horizontal line through the geometric center of the cavity

Fig. 13: Velocity profile when $(\theta_1,\theta_2)=(1,1)$

To further validate the accuracy of the GNIROM, flow fields are compared at four test points marked by delta spots when t=5.0s in Figure 11. The error distributions of the three state variables are presented at those test points in Figure 14 specifically. Similar to the diffusion problem, there always exist bigger errors at the regions with a dramatic change in flow fields. As a whole, the errors are small in the whole field, demonstrating that GNIROM reconstructs the flow fields very well.

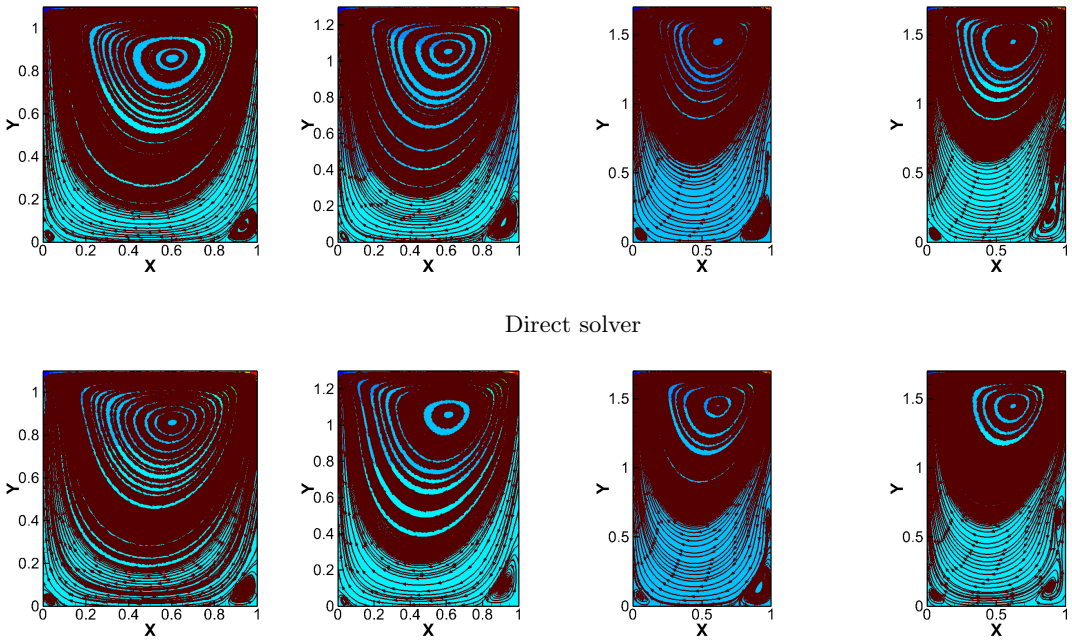
Moreover, streamlines and vorticity contours at t=5.0s between GNIROM's approximation and



(a)(1.1, 1.1) (b)(1.3, 1.3) (c)(1.5, 1.7) (d)(1.7, 1.7)

Fig. 14: Error distributions at the four points when t=5.0s

the Direct solver's result are compared specifically in Figure 15 and 16. From Figure 15, there exists a minor difference between the two rows of streamlines, which leads to a difference in visual discrimination, especially at the bottom corners. The reason is that, at the bottom corners the velocities' values are close to 0, leading to the fact that minor variations in velocity contours may



(a)$(1.1, 1.1)$ (b)$(1.3, 1.3)$ (c)$(1.5, 1.7)$ (d)$(1.7, 1.7)$

Fig. 15: Streamlines at the four test points when t=5.0s

lead to differences in the streamlines in such regions. However, GNIROM is competent in simulating the trend of streamlines as well as capturing the corner vortices even when the height of cavity varies a lot. And a very good agreement is presented in vorticity contours in Figure 16. Considering that vorticity is a differentiated quality, Figure 16 further predicts that a high approximation accuracy is achieved in this case.

3. Computational cost evaluation

Computational efficiency is a key factor to judge a ROM's performance and in this section computational cost of the proposed GNIROM is divided into two parts:

In the offline stage, the approach determines the samples iteratively. At each iterative step, the GNIROM includes four parts: extracting reduced basis, training RBF with samples' data, determining the new candidate and generating the new snapshot with the numerical solver. With the iterative step growing, the CPU timings of the first three parts increase correspondingly, thus only time cost at the last cycle is presented in Table 8. As shown in the table, the greedy approach

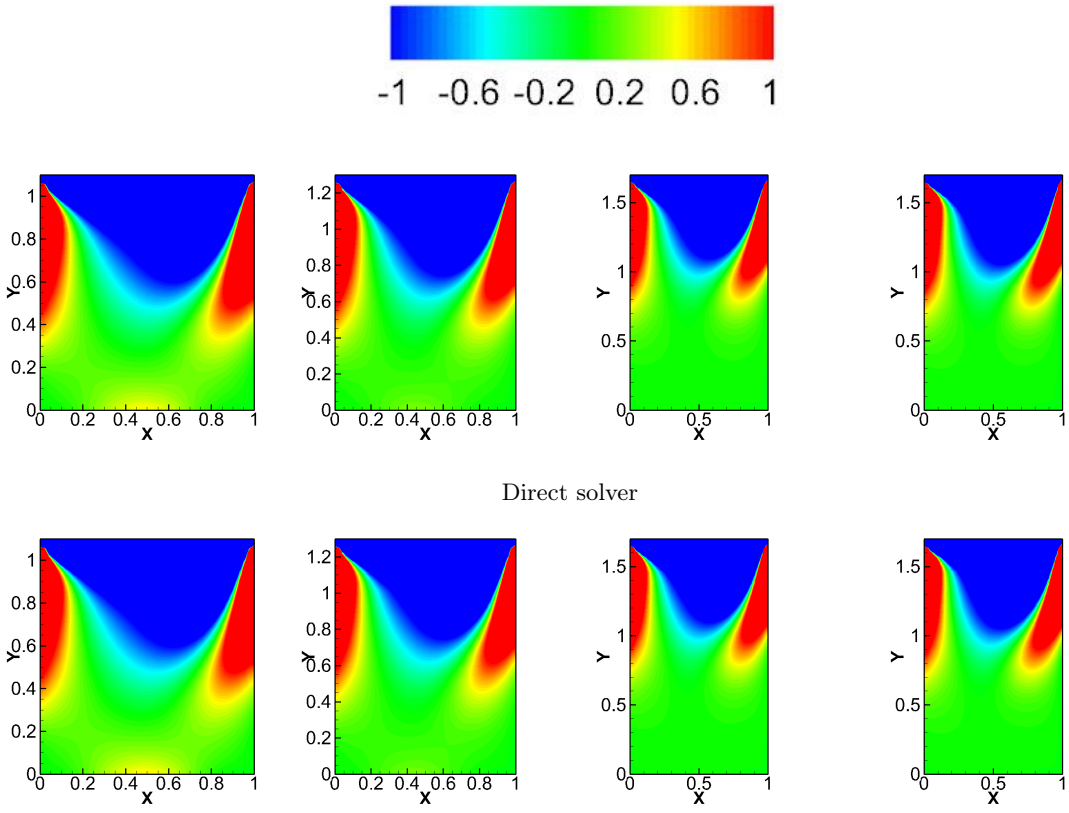


Fig. 16: Comparison of vorticity contours at the four test points when t=5.0s

is very cheap in computational cost.

In the online stage, solutions are approximated with the extracted basis functions. Table 9 lists the timing for one test sample approximated by the ROM as well as the direct solver, demonstrating that the CPU timing's speed-up is very significant.

Combining with the high approximation accuracy, it can be concluded that the proposed approach performs excellently and efficiently in this case.

V. Concluding remarks

This paper has presented the development of a non-intrusive ROM based on a greedy approach. To enable a robust approximation, this paper has introduced the use of RBF-QR. The efficacy and efficiency of the proposed method has been demonstrated by two cases, including heat conduction

Table 8: Time cost(in seconds) required by Greedy sampling at 28th cycle.

Phase	Time cost(in seconds)
Sampling:Extracting reduced basis	26.53
Sampling:Computing $\gamma_i^{km}(i = 1, 2, \cdots N)$	2.45
Sampling:Determining the new candidate	4.25
Numerical Solver	163.4
Total	196.63

Table 9: Time cost(in seconds) required by the solver and ROM (online stage).

Phase	Time cost(in seconds)
ROM: approximation for one sample (online)	0.0468
Numerical Solver(for one sample)	163.4

and a driven cavity flow. Although this paper focus on fluid dynamics, the approach is capable for approximating general parameterized time-accurate problems.

While the results are encouraging and the approach has been demonstrated on non-trivial examples, future work should seek to improve the effectiveness of the GNIROM algorithm, possibly through the use of a local reduced basis. It is also expected to adopt this approach to problems of a more complex nature in aerodynamics and aeroelasticity including those with larger dimension of parameter space and 3-dimensional problems.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under grant no. 11502211.

References

[1] Sirovich L. Turbulence and the dynamics of coherent structures. *Quarterly of applied mathematics*, 45(3):561–590, 1987.

[2] Hall KC. Eigenanalysis of unsteady flows about airfoils, cascades and wings. *AIAA Journal*, 32(12):2426–2432, 1994.

[3] Lumley JL. The structure of inhomogeneous turbulence. In *Proceedings of the International Colloquium on the Fine Scale Structure of the Atmosphere and its Influence on Radio Wave Propagation*, pages 166–178. Moscow:Nauka, 1967.

[4] Hall KC, Thomas JP, and Clark WS. Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. *AIAA journal*, 40(5):879–886, 2002.

[5] Silva WA. *Discrete-time linear and nonlinear aerodynamic impulse responses for efficient CFD analyses*. 1997.

[6] Deaner AE and Kevrekidis GE. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Physics of Fluids A: Fluid Dynamics*, 3(10):2337–2354, 1991.

[7] Xiao D, Fang F, Buchan AG, Pain CC, et al. Non-intrusive reduced order modelling of the navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 293:522–541, 2015.

[8] Casenave F, Ern A, and Lelièvre T. A nonintrusive reduced basis method applied to aeroacoustic simulations. *Advances in Computational Mathematics*, 41(5):961–986, 2015.

[9] Barrault MY, Maday Y, Nguyen NC, et al. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[10] Zimmermann R and Gortz S. Improved extrapolation of steady turbulent aerodynamics using a non-linear pod-based reduced order model. *The Aeronautical Journal*, 116(1184):079–1100, 2012.

[11] Walton S, Hassan O, and Morgan K. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20):8930–8945, 2013.

[12] Xiao D, Fang F, and Pain C. Non-intrusive reduced-order modelling of the navierÍcstokes equations based on rbf interpolation. *International Journal for Numerical Methods in Fluids*, 79(11):580–595, 2015.

[13] Audouze C, De Vuyst F, and Nair PB. Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations. *Numerical Methods for Partial Differential Equations*, 29(5):1587–1628, 2013.

[14] Fornberg B and Flyer N. Accuracy of radial basis function interpolation and derivative approximation on 1-d infinite grids. *Advances in Computational Mathematics*, 23(1-2):5–20, 2005.

[15] Nguyen NC and Peraire J. An efficient reduced-order modeling approach for non-linear parametrized

partial differential equations. *International Journal for Numerical Methods in Engineering*, 76(1):27–55, 2008.

[16] Knezevic DJ, Nguyen NC, and Patera AT. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, 21(07):1415–1442, 2011.

[17] Noack BR, Afanasiev K, Morzynski M, and Tadmor G. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497(497):335–363, 2003.

[18] Qawasmeh BR and Wei M. Low-dimensional models for compressible temporally developing shear layers. *Journal of Fluid Mechanics*, 731(731):364–393, 2013.

[19] Ballarin F, Faggiano E, and Ippolito S. Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a pod–galerkin method and a vascular shape parametrization. *Journal of Computational Physics*, 315:609–628, 2016.

[20] Chaturantabut S and Sorensen DC. Discrete empirical interpolation for nonlinear model reduction. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 4316–4321. IEEE, 2009.

[21] Xiao D, Fang F, Pain C, and Navon IM. A parameterized non-intrusive reduced order model and error analysis for general time-dependent nonlinear partial differential equations and its applications. *Computer Methods in Applied Mechanics and Engineering*, 317:868–889, 2017.

[22] Sobol' IM. On the systematic search in a hypercube. *SIAM Journal on Numerical Analysis*, 16(5):790–793, 1979.

[23] Du Q, Faber V, and Gunzburger M. Centroidal voronoi tessellations: Applications and algorithms. *SIAM review*, 41(4):637–676, 1999.

[24] Romero VJ, Burkardt JV, Gunzburger MD, et al. Initial evaluation of pure and "latinized" centroidal voronoi tessellation for non-uniform statistical sampling. 2005.

[25] Ahmadpoor M, Notghi B, and Brigham JC. A generalized iterative approach to improve reduced-order model accuracy for inverse problem applications. *Journal of Engineering Mechanics*, 142(5):04016020, 2016.

[26] Haasdonk B and Ohlberger M. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.

[27] Haasdonk B. Convergence rates of the pod–greedy method. *ESAIM: Mathematical Modelling and Numerical Analysis*, 47(3):859–873, 2013.

[28] Buffa A, Maday Y, Patera AT, et al. A priori convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(3):595–603, 2012.

[29] Binev P, Cohen A, Dahmen W, et al. Convergence rates for greedy algorithms in reduced basis methods. *SIAM journal on mathematical analysis*, 43(3):1457–1472, 2011.

[30] Veroy K and Patera AT. Certified real-time solution of the parametrized steady incompressible navier–stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005.

[31] Veroy K, Prud'Homme C, Rovas D, et al. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference*, page 3847, 2003.

[32] Rozza G, Huynh DBP, and Patera AT. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229, 2008.

[33] Grepl MA. Certified reduced basis methods for nonaffine linear time-varying and nonlinear parabolic partial differential equations. *Mathematical Models and Methods in Applied Sciences*, 22(03):1150015, 2012.

[34] Sasena M, Parkinson M, Goovaerts P, et al. Adaptive experimental design applied to ergonomics testing procedure. In *ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 529–537. American Society of Mechanical Engineers, 2002.

[35] Bui-Thanh T, Willcox K, and Ghattas O. Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications. *AIAA journal*, 46(10):2520–2529, 2008.

[36] Fornberg B, Larsson E, and Flyer N. Stable computations with gaussian radial basis functions. *SIAM Journal on Scientific Computing*, 33(2):869–892, 2011.

[37] Aubry N, Guyonnet R, and Lima R. Spatiotemporal analysis of complex signals: theory and applications. *Journal of Statistical Physics*, 64(3):683–739, 1991.

[38] Karhunen K. Zur spektraltheorie stochastischer prozesse. 1946.

[39] Kaufman L. Maximum likelihood, least squares, and penalized least squares for pet. *IEEE Transactions on Medical Imaging*, 12(2):200–214, 1993.

[40] Nowlan SJ. Maximum likelihood competitive learning. In *NIPS*, pages 574–582, 1989.

[41] Thacher Jr and Henry C. Conversion of a power to a series of chebyshev polynomials. *Communications of the ACM*, 7(3):181–182, 1964.

[42] Ghia U, Ghia KN, and Shin CT. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, 48(3):387–411, 1982.