A Reduced-order Model for Gradient-based Aerodynamic Shape Optimisation☆

Weigang Yao[a,1], Simão Marques[b,1,*], Trevor Robinson[c,2], Cecil Armstrong[c,3], Liang Sun[d,4]

[a]*Faculty of computing, Engineering and Media, De Montfort University, Leicester, LE1,9BH, U.K.*
[b]*Department of Mechanical Engineering Sciences, The University of Surrey, Guildford, England, GU2 7XH U.K.*
[c]*School of Mechanical and Aerospace Engineering, Queen's University Belfast, Belfast, Northern Ireland, BT9 5AH, U.K.*
[d]*School of Computing, Engineering and Intelligent Systems, Ulster University, Northern Ireland, BT48 7JL, U.K.*

Abstract

This work presents a reduced order model for gradient based aerodynamic shape optimization. The solution of the fluid Euler equations is converted to reduced Newton iterations by using the Least Squares Petrov-Galerkin projection. The reduced order basis are extracted by Proper Orthogonal Decomposition from snapshots based on the fluid state. The formulation distinguishes itself by obtaining the snapshots for all design parameters by solving a linear system of equations. Similarly, the reduced gradient formulation is derived by projecting the full-order model state onto the subspace spanned by the reduced basis. Auto-differentiation is used to evaluate the reduced Jacobian without forming the full fluid Jacobian explicitly during the reduced Newton iterations. Throughout the optimisation trajectory, the residual of the reduced Newton iterations is used as an indicator to update the snapshots and enrich the reduced order basis. The resulting multi-fidelity

*Corresponding Author
Email address: s.marques@surrey.ac.uk (Simão Marques)

Preprint submitted to Aerospace Science & Technology *August 17, 2020*

[1]Senior Lecturer
[2]Reader
[3]Professor
[4]Lecturer

optimisation problem is managed by a trust-region algorithm. The ROM is demonstrated for a subsonic inverse design problem and for an aerofoil drag minimization problem in the transonic regime. The results suggest that the proposed algorithm is capable of aerodynamic shape optimization while reducing the number of full-order model queries and time to solution with respect to an adjoint gradient based optimisation framework.

Keywords: ROM, CFD, Aerodynamics, Shape Optimisation, Gradient-based Optimisation, Trust-Region, Multi-fidelity

Nomenclature

Latin Symbols

C_l	=	lift coefficient
C_d	=	drag coefficient
$\mathcal{C}_{\mathcal{E}}$	=	equality constraint
$\mathcal{C}_{\mathcal{I}}$	=	inequality constraint
C_m	=	pitching moment coefficient
$\mathbf{D}$	=	diagonal matrix containing singular values
$\mathcal{F}$	=	objective function
$\mathbf{J}$	=	fluid Jacobian matrix
lb	=	design parameter lower bound values
M_∞	=	free-stream Mach number
$MUSCL$	=	Monotonic Upwind Scheme for Conservation Laws
N	=	number of degrees of freedom
n_p	=	number of deseign parameters
n_r	=	number of basis
p	=	static pressure
$\hat{p}$	=	non-dimensional static pressure, $\hat{p} = \dfrac{p}{q_\infty}$
$\mathbf{p}$	=	line search vector for Newton method
q	=	dynamic pressure
$\mathbf{R}$	=	vector of fluid equations residuals
$\mathbf{S}$	=	surface mesh points
$\mathbf{s}$	=	optimisation step
t	=	time
$\mathbf{U}$	=	left singular vectors

ub	=	design parameter upper bound values
$\mathbf{V}$	=	right singular vectors
V_n	=	design velocity
$\mathbf{w}$	=	vector of fluid and structural unknowns
$\mathbf{w}_r$	=	reduced vector of fluid unknowns
$\mathbf{X}$	=	matrix of snapshots

Greek Symbols

α	=	step length for Newtown method, angle of attack
Δ	=	trust region radius
δ	=	small perturbation
ε	=	residual threshold
ε_μ	=	trust-region termination threshold
ε_∇	=	trust-region gradient threshold
ζ	=	constant for Carter condition
Λ	=	vector of adjoint unknowns
λ_i	=	singular value
η_1, η_2	=	trust-region effectiveness thresholds
ρ	=	trust-region effectiveness
Φ	=	vector subspace
μ	=	design parameters
Ψ	=	reduced Jacobian matrix

1. Introduction

Numerical shape optimization usually requires the solution of parametric Partial Differential Equations (PDEs). This is a challenging process due to the high computational cost associated with having to interrogate a large and complex model multiple times. The number of evaluations of the full-order model (FOM) depends on the number of parameters, the number of constraints and objectives of the problem. For shape optimization problems this usually means several more parameters than functional constraints. Even for problems with a moderate number of parameters, the need for multiple model evaluations quickly makes it less attractive to deploy evolutionary methods or alternative global optimization algorithms [34]. Gradient-based optimization approaches, on the other hand, require the calculation of parametric sensitivity and respective function gradients. The cost of evaluating gradients, i.e. the scaling of the gradient evaluation with the number of parameters and associated number of model evaluations, can be mitigated by employing adjoint methods [30, 16, 17, 12, 28, 13]. This, however, introduces another PDE system with the same number of degrees of freedom as the primal PDE problem, equally or more challenging to solve.

Reduced-order modelling remains a popular topic in many engineering disciplines as a means to accelerate otherwise impractical or intractable simulations. Broadly speaking, the term reduced-order model (ROM) is a relative term that presupposes the existence of a FOM, of which some output is to be replicated at a reduced expense. More pertinent to this work, are the class of ROMs that maintain a close link to the physics described by the FOM but are computationally more efficient. This typically involves exam-

ining the governing equations and performing some type of model reduction, suitable for the physics of the problem. A particular model reduction technique assumes that the parametric behaviour of the FOM can be captured by a small number of modes or basis, typically obtained by methods such as Proper Orthogonal Decomposition (POD) [33], Balance POD [35, 32], Proper Generalized Decomposition [4]; excellent overviews of the range of applications and model reduction methods can be found in [22, 29, 20, 6, 5, 37]. In this work the resultant set of modes or basis is referred to as reduced order basis (ROB).

Within the context of model reduction, shape optimization involving nonlinear fluid equations poses significant challenges to model reduction due to the large number of parameters, more or less exacerbated by the degree of nonlinearity exhibited by the flow problem. Therefore, it is difficult to build a static and global basis that effectively covers the design space. Relevant efforts to this study include the approach for inverse aerofoil design of LeGresley and Alonso [21], where a POD ROM based on a Hicks-Henne surface parameterization was constructed to approximate the gradient by finite-differencing. Manzoni et al. successfully solve the Stokes equations for shape optimization of coronary arteries parameterized using a mapped Free-Form Deformation (FFD) technique and a reduced basis method [23]; the reduced basis are built from samples of the design space, obtained during an offline phase that incurs the majority of the computational cost. This technique was later expanded and applied to minimize vorticity by solving the same equations [27]. The elliptical nature of the PDE system enabled the authors to compute error bounds for the ROM. The availability of error

bounds for the surrogate model allowed Yue and Meerbergen [38] to prove that a trust-region method converges to the FOM optimum relying only on surrogates. This idea is further developed in [31], that proposed a trust region approach using reduced basis to build a ROM with a posteriori error bounds for elliptical and parabolic PDEs. Here, instead of building a global set of basis, the trust region method triggers updates to the basis when the error deteriorates, minimizing calls to the FOM.

For problems exhibiting stronger nonlinearities, building a global ROB, deducing error bounds for the ROM becomes increasingly challenging or is not feasible. A possible solution to this problem is to implement a zonal approach where the FOM is used to solve for the region of the flow subject to strong nonlinearities and a ROM reduces the overall cost by solving the remainder of the domain [15]. In the absence of strict error bounds, Zahr and Farhat assume a monotonic relationship between residual norm of the ROM and its error and proposed a nonlinear trust-region optimisation method that updates the ROB along the optimization trajectory, when the residual fails to reach a required threshold [39]. The authors exploit a Least-Squares Petrov-Galerkin projection [18] to reduce the state equations and respective sensitivities, hence, each snapshot involved concatenating samples of the fluid state variables and sensitivities with respect to the design variables.

To further reduce the number of snapshots required to build accurate ROMs, this work proposes a new type of ROM for gradient based aerodynamic shape optimization problems, centred on ROB built with samples from solving a linear system. As in reference [39], the residual is used to trigger the enrichment of the ROB within a trust-region framework, however, in this

work only sensitivity information is collected as part of each snapshot used to build both the fluid state and compute the gradient of the design variables. Further efficiency is gained by employing an auto-differentiation procedure to compute the reduced Jacobian. A subsonic inverse design and a transonic flow drag minimization problems are adopted to demonstrate the capability of the ROM based optimization methodology for problems exhibiting weak and strong nonlinearities. The efficiency of the proposed method is benchmarked against the FOM adjoint based optimisation strategy.

The paper is organized as follows: section 2 introduces the reference optimisation framework using the FOM for analysis and respective adjoint based gradient computation; section 3 derives the ROM, describes the ROBs updating procedure and the ROM gradient evaluation required for the optimisation; section 4 introduces the trust-region formalism used to solve the optimisation problems employing the ROM; this is followed by two sets of results and respective analysis; finally, the paper finishes with a conclusion section.

2. Shape Optimization Using a Full-Order Model

The FOM used to obtain the flow solutions in this work solves the compressible fluid Euler equations. The equations are discretized using a cell-centred finite-volume scheme over block-structured conforming meshes, employing a Roe flux function, together with $MUSCL$ interpolation, and the van Albada limiter is used to obtain second-order accuracy. The nonlinear system of algebraic equations is marched forward in time by an explicit four-stage Runge-Kutta method [36].

The flow solver provides the optimisation objective and constraints as a function of a given geometry. The geometry itself, is parameterised in terms of design parameters that allow defining the shape of interest. Hence, the optmization problem can be formulated as:

$$\begin{aligned} \underset{\mu \in \mathscr{D}}{\text{minimize}} \quad & \mathcal{F}(\mathbf{w}(\mu), \mu) \\ \text{subject to} \quad & \mathbf{R}(\mathbf{w}(\mu), \mu) = 0, \\ & \mathcal{C}_{\mathcal{E}}(\mathbf{w}(\mu), \mu) = 0, \\ & \mathcal{C}_{\mathcal{I}}(\mathbf{w}(\mu), \mu) \leq 0, \end{aligned} \tag{1}$$

where $\mathcal{F}$ is the objective function, $\mathbf{w}$ and μ are fluid state and design parameters, respectively, $\mathscr{D} \subset \mathbb{R}^{n_p}$ represents the parameter space, $\mathbf{R}$ is the residual of the Euler equations and $\mathcal{C}_{\mathcal{E}}$, $\mathcal{C}_{\mathcal{I}}$ are equality and inequality constraints, respectively. To solve for the fluid state variables, the fluid residual $\mathbf{R}(\mathbf{w}(\mu), \mu)$ is driven to zero.

For gradient-based optimisation, the total derivative can be used to express how the objective function changes with respect μ:

$$\frac{d\mathcal{F}}{d\mu} = \frac{\partial \mathcal{F}}{\partial \mu} + \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \frac{d\mathbf{w}}{d\mu}. \tag{2}$$

Note that while computing Eq. 2, $\mathbf{R}(\mathbf{w}(\mu), \mu) = 0$ must still be satisfied, therefore we can use

$$\frac{d\mathbf{R}}{d\mu} = \frac{\partial \mathbf{R}}{\partial \mu} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \frac{d\mathbf{w}}{d\mu} = 0. \tag{3}$$

The cost of evaluating $\frac{d\mathbf{w}}{d\mu}$ is not trivial as it requires the solution of the

following linear system

$$\frac{d\mathbf{w}}{d\mu} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \mu}, \tag{4}$$

where the term $\frac{\partial \mathbf{R}}{\partial \mathbf{w}} \in \mathbb{R}^{N \times N}$, is the fluid Jacobian $\mathbf{J}$, with N representing the degrees of freedom of the fluid state. Substituting Eq. 4 into Eq. 2, results in

$$\frac{d\mathcal{F}}{d\mu} = \frac{\partial \mathcal{F}}{\partial \mu} - \frac{\partial \mathcal{F}}{\partial \mathbf{w}}\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \mu}. \tag{5}$$

The evaluation of the derivatives in Eq. 5 can be performed by finite-differencing, however this quickly becomes impractical even for a small number of design variables. Alternatively, it is possible to employ the so called *direct* or *adjoint* methods. This requires the solution of an appropriate linear system, see for example Hwang and Martins for further details [24]. The direct method requires the solution of Eq. 4, the cost of solving this linear system is proportional to the number of design variables. The adjoint method involves re-writing Eq. 5 as

$$\frac{d\mathcal{F}}{d\mu} = \frac{\partial \mathcal{F}}{\partial \mu} + \Lambda^T \frac{\partial \mathbf{R}}{\partial \mu}, \tag{6}$$

where Λ^T is the adjoint vector, obtained by solving the following linear (adjoint) system,

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^T \Lambda = -\left[\frac{\partial \mathcal{F}}{\partial \mathbf{w}}\right]^T. \tag{7}$$

The Jacobian matrix $\frac{\partial \mathbf{R}}{\partial \mathbf{w}}$ and right hand side vector $\frac{\partial \mathcal{F}}{\partial \mathbf{w}}$ in Eq. 7 are obtained through auto-differentiation of the discretized Euler equations using the Tapenade library [14]. The Jacobian is typically a large, sparse matrix where only the non-zeros entries are computed. For small problems, matrix

factorization can be used to solve Eq. 7, which is the case in this work; alternatively, iterative solvers such as GMRES can be used for larger meshes, whereby only matrix-vector products are necessary to solve the linear system.

The adjoint system scales with the number of functions in Eq. 1, therefore, is the preferred choice for aerodynamic shape optimisation problems, where the number of parameters, n_p, usually outnumbers the number of objective and constraint functions. In this work, the optimisation results based solely on the FOM are used as reference and are solved using the Sequential Quadratic Programming (SQP) algorithm from MatLab's *fmincon* function [26], with the gradient built using the solution of the adjoint system represented in Eq. 7.

3. Shape Optimization Using a Reduced-Order Model

The aim of this work is to reduce the cost of solving Eq. 1, by employing an approximate model, i.e. the ROM, in lieu of the FOM, at minimum loss of fidelity. To minimize cost, the ROM is built as the optimisation progresses, i.e. the ROM is only developed along the optimisation trajectory. This work aims to minimize or mitigate the dependence of the number of model queries on the number of design variables required to either build the ROM or compute gradients. The result is a multi-fidelity optimisation formulation, employing a trust-region strategy to manage the high and low fidelity model solves.

3.1. FOM Reduction

The central idea of projection-based ROMs is to project the full state vector $\mathbf{w}$ onto a subspace $\Phi \in \mathbb{R}^{N \times n_r}$ spanned by the ROB, where $N \gg n_r$,

i.e.:

$$\mathbf{w} \approx \bar{\mathbf{w}} + \Phi \mathbf{w}_r, \tag{8}$$

where $\bar{\mathbf{w}}$ is the operation reference state and $\mathbf{w}_r$ is the reduced state vector. By Substituting Eq. 8 into the fluid residual $\mathbf{R}$, the FOM is converted to the following optimization problem,

$$\underset{\mathbf{w}_r \in \mathbb{R}^{n_r}}{\text{minimize}} \quad L_2 = \frac{1}{2} \|\mathbf{R}(\bar{\mathbf{w}} + \Phi \mathbf{w}_r, \mu)\|_2^2. \tag{9}$$

This type of ROM is equivalent to the minimum residual approach, meaning that if the ROB is enriched, the solution error in the $\mathbf{J}^T\mathbf{J} - norm$ is non-increasing [8]. The first order optimality condition gives

$$\frac{dL_2}{d\mathbf{w}_r} = \Psi^T \mathbf{R}(\bar{\mathbf{w}} + \Phi \mathbf{w}_r, \mu) = 0, \tag{10}$$

where $\Psi = \mathbf{J}\Phi \in \mathbb{R}^{N \times n_r}$ is the reduced Jacobian matrix. Following [18], the Newton method is employed to solve Eq. 10, referred to as the reduced Newton iterations:

$$\Phi^T \mathbf{J}_k^T \mathbf{J}_k \Phi \mathbf{p}_k = -\Phi^T \mathbf{J}_k^T \mathbf{R}_k \tag{11}$$

$$\mathbf{w}_r^{k+1} = \mathbf{w}_r^k + \alpha_k \mathbf{p}_k. \tag{12}$$

In the above reduced Newton method, also known as Least-Squares Petrov-Galerkin projection formulation, α_k is the step length and $\mathbf{p}_k$ is the line search direction vector. The reduced Jacobian $\mathbf{J}\Phi$ needs to be evaluated at each iteration, which is the major computational cost and requires access to the solver functions. As mentioned above, the Jacobian matrix is typically a large and sparse matrix, scalable with $\mathbf{w}$ degrees of freedom. Evaluating and storing $\mathbf{J}$ explicitly quickly becomes impractical for large scale problems, By

realizing that

$$\mathbf{J}\Phi = \delta\mathbf{R}(\bar{\mathbf{w}})_\Phi \approx \frac{1}{\delta}\left[\mathbf{R}(\bar{\mathbf{w}} + \delta\Phi, \mu) - \mathbf{R}(\bar{\mathbf{w}}, \mu)\right] \tag{13}$$

it is apparent that $\mathbf{J}\Phi$ can be approximated by finite-differences, however this would require the user to set the perturbation parameter, δ. This can be avoided by using auto-differentiation. Recall that $\Phi \in \mathbb{R}^{N\times n_r}$, therefore forming $\mathbf{J}\Phi$ requires $N \times n_r$ flux function calls. The value $(N \times n_r)$ is, in general, much smaller than the number of non-zero entries in $\mathbf{J}$, i.e. the number of flux function evaluations required to explicitly compute $\mathbf{J}$, which suggests auto-differentiation methods become more advantageous for larger problems, therefore it is the preferred approach in this work.

3.2. Construction and Enrichment of Reduced Order Basis

It is common practice to extract ROB from a matrix of snapshots or database $\mathbf{X}$ by POD [33]. The key is to build sufficient data in the range of interest. In the present work, only snapshots of $\frac{\partial \mathbf{w}}{\partial \mu}$ samples are considered, assuming the nonlinearity is weak or the state does not deviate far from the operation state $\bar{\mathbf{w}}$. The snapshots are computed by solving the linear system of equations given by Eq. 4, that is,

$$\mathbf{X} = \frac{d\mathbf{w}}{d\mu} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1}\left[\frac{\partial \mathbf{R}}{\partial \mu}\right]. \tag{14}$$

In Eq. 14, $\frac{\partial \mathbf{R}}{\partial \mathbf{w}}$ is computed from the fluid system and

$$\frac{\partial \mathbf{R}}{\partial \mu} = \frac{\partial \mathbf{R}}{\partial \mathbf{S}}\frac{\partial \mathbf{S}}{\partial \mu} \tag{15}$$

where $\mathbf{S}$ represents the surface mesh and $\frac{\partial \mathbf{S}}{\partial \mu}$ is obtained from the parameterization of the problem, e.g. Class-Shape Transformation (CST) [19], FFD [24]

or third party CAD systems [2]; in this work the mesh is deformed following the transfinite interpolation method and the term $\frac{\partial \mathbf{R}}{\partial \mathbf{S}}$ which represents the flow residual sensitivity to mesh changes is obtained by auto-differentiation.

Once the database is obtained, the following procedure is adopted to build the subspace Φ:

$$\mathbf{X}^T\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T, \tag{16}$$

$$\Phi = \mathbf{X}\mathbf{V}\mathbf{D}^{-\frac{1}{2}}. \tag{17}$$

Where $\mathbf{D} = diag[\lambda_1, \lambda_2, ..., \lambda_{np}]$ corresponds to the singular values, and $\mathbf{U}$, $\mathbf{V}$ are the left and right singular vectors, respectively. The cost of applying singular value decomposition (SVD) to $\mathbf{X}^T\mathbf{X}$, which is an $n_r \times n_r$ matrix, is trivial.

It is difficult, if possible at all, to construct a static global subspace Φ for a nonlinear system, therefore, it is necessary to update Φ along the optimization trajectory. The L_2 norm of the fluid residual in the Newton's iteration is used as an indicator to update Φ. If the L_2 norm remains larger than the user defined threshold ε, new snapshots are generated by Eq. 14 at the current state and appended to the previous snapshot matrix, and used to enrich Φ.

3.3. Gradient Evaluation

By substituting Eq. 8 into the FOM gradient Eq. 4 yields the reduced gradient approximation:

$$\frac{d\mathbf{w}}{d\mu} \approx -\Phi \left[\mathbf{J}\Phi\right]^+ \frac{d\mathbf{R}}{d\mu}, \tag{18}$$

Where "+" denotes Moore-Penrose pseudoinverse. Substituting Eq. 18 into Eq. 2 and obtain

$$\frac{d\mathcal{F}}{d\mu} \approx \frac{\partial \mathcal{F}}{\partial \mu} - \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \Phi \left[\mathbf{J}\Phi\right]^{+} \frac{d\mathbf{R}}{d\mu}. \tag{19}$$

Eq. 19 can be expressed as the following by introducing the design velocity $V_n = \frac{d\mathbf{S}}{d\mu}$, which is the surface mesh derivative with respect to the design parameter μ [2],

$$\frac{d\mathcal{F}}{d\mu} \approx \left[\frac{\partial \mathcal{F}}{\partial \mathbf{S}} - \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \Phi \left[\mathbf{J}\Phi\right]^{+} \frac{\partial \mathbf{R}}{\partial \mathbf{S}}\right] V_n. \tag{20}$$

The objective function sensitivity to the mesh changes, $\frac{\partial \mathcal{F}}{\partial \mathbf{S}}$, is obtained from the auto-differentiation of the solver.

Compared with the full order adjoint system in Eq. 7, which requires solving an $N \times N$ linear system, the reduced system only needs to solve an $N \times n_r$ linear system, which provides the speed up in the gradient computation. However, it is necessary to point out that the Φ updates scale with the number of design variables due to the need to solve Eq. 14, hence this method remains competitive if the number of Φ updates remains significantly lower than the FOM calls required by the adjoint based method.

4. Trust-Region Framework

As the name indicates, the trust-region method aims to establish a subdomain where the low-fidelity model provides an adequate representation of the FOM [10]. At each major iteration k, an optimisation subproblem is

defined on the trust-region centred at μ_k and radius Δ_k:

$$\begin{aligned} \underset{s\in\mathfrak{B}_k}{\text{minimize}} \quad & \hat{\mathcal{F}}(\mathbf{w}(\mu_k),\mu) \\ \text{subject to} \quad & \hat{\mathcal{C}}_{\mathcal{E}}(\mathbf{w}(\mu_k),\mu_k+s)=0, \\ & \hat{\mathcal{C}}_{\mathcal{I}}(\mathbf{w}(\mu_k),\mu_k+s)\leq 0, \\ & lb\leq(\mu_k+s)\leq ub, \\ & ||s||_\infty\leq\Delta_k, \end{aligned} \tag{21}$$

where the " ˆ " symbol indicates quantities computed using the ROM (using Eq. 8 for the functionals and Eq. 20 for the gradients); s is the optimisation step size and $\mathfrak{B}_k=\{\mu\in\mathbb{R}^{n_p}:||\mu-\mu_k||\leq\Delta_k\}$.

Trust region methods are provably convergent for constrained optimisation problems to an optimum of the FOM, provided both models satisfy a number of conditions, including that the low-fidelity model is corrected to be at least first-order consistent with the FOM [3, 7, 1], i.e.:

$$\mathcal{F}(\mu_k)=\hat{\mathcal{F}}(\mu_k),\mathcal{C}_{\mathcal{E}}(\mu_k)=\hat{\mathcal{C}}_{\mathcal{E}}(\mu_k),\mathcal{C}_{\mathcal{I}}(\mu_k)=\hat{\mathcal{C}}_{\mathcal{I}}(\mu_k) \tag{22}$$

$$\nabla\mathcal{F}_k=\nabla\hat{\mathcal{F}}_k,\nabla\mathcal{C}_{\mathcal{E}}(\mu_k)=\nabla\hat{\mathcal{C}}_{\mathcal{E}}(\mu_k),\nabla\mathcal{C}_{\mathcal{I}}(\mu_k)=\nabla\hat{\mathcal{C}}_{\mathcal{I}}(\mu_k). \tag{23}$$

For unconstrained optimisation, the first-order consistency requirement can be relaxed and a suitable approximation of the gradient at the centre of the trust region will suffice. This approximation is usually based on the Carter condition,

$$\frac{||\nabla\mathcal{F}_k-\nabla\hat{\mathcal{F}}_k||}{||\nabla\hat{\mathcal{F}}_k||}\leq\zeta,\ \forall k, \tag{24}$$

with the constant $\zeta<1$, [9, 11].

To guarantee convergence to a stationary point of the FOM, the following are also required [10]: a) $\hat{\mathcal{F}}$ is locally Lipschitz continuous and regular with

respect to s for all μ and continuous in μ for all s; b) the set of problem parameters is closed and bounded; c) the sufficient decrease condition requires the step to satisfy the fraction of Cauchy decrease (FCD). In addition the second derivatives of the ROM at μ_k remain bounded within the trust region domain for all k [1]. The FCD condition was derived for the classical trust-region method, where the FOM is approximated by a quadratic Taylor's series expansion. With POD based approaches, this is no longer feasible, instead the low-fidelity function step is determined by solving Eq. 21 using MatLab's *fmincon* SQP algorithm, which maximizes the decrease in the objective function of the trust-region subproblem.

The effectiveness of the trust-region step is evaluated by the ratio of the actual improvement over the improvement predicted by the ROM, ρ:

$$\rho_k = \frac{\mathcal{F}(\mu_k) - \mathcal{F}(\mu_k + s)}{\hat{\mathcal{F}}(\mu_k) - \hat{\mathcal{F}}(\mu_k + s)}. \tag{25}$$

For values of $\rho < \eta_1$ the step is rejected and the trust-region radius is reduced; if $\eta_1 < \rho < \eta_2$, the trust-region size is maintained and increased when $\rho > \eta_2$. In this work, the trust-region is set up with $\eta_1 = 0.5$ and $\eta_2 = 0.9$; the trust-region size, Δ_{k+1}, is then reduced by a factor of 0.5, maintained or increased by a factor 1.25.

The optimisation terminates when the change in the design variables is less than the termination threshold, ε_μ or the trust-region size drops below a minimum $\Delta_k < \Delta_{min}$. This is complemented by enforcing the assessment of the ratio ρ when the ROM gradient norm is below the threshold ε_∇. Hence, if the ROM is not accurate when the ROM optimiser is in effect converged, the trust-region radius is reduced, which means that eventually the linear approximation will be valid and the FOM gradient norm will also drop below

the ε_∇. This is particularly relevant for highly nonlinear problems, where the ROM accuracy is more volatile. Algorithm 1 summarises the complete process.

5. Results

5.1. Subsonic Flow Inverse Design

A subsonic inverse design test case used in [39] was chosen to assess the current ROM, the problem is formulated as:

$$\begin{aligned} \underset{\mu \in \mathscr{D}}{\text{minimize}} \quad & \frac{1}{2}\|p(w,\mu) - p_{RAE2822}\|_2^2 \\ \text{subject to} \quad & \mathbf{R}(\mathbf{w}(\mu),\mu) = 0 \end{aligned} \tag{26}$$

where p is the pressure distribution. Equation 26 aims to modify the aerofoil shape to match the target pressure distribution $p_{RAE2822}$ produced by an inviscid analysis of the $RAE2822$ aerofoil, hence the optimum shape should correspond to the $RAE2822$ aerofoil. The flow is described by the fluid Euler equations, the free-stream conditions are $(M_\infty, \alpha) = (0.5, 0^\circ)$. The $NACA0012$ aerofoil is used as the initial geometry. The CST method using 20 weights ($n_p = 20$), is used to parameterize the aerofoil by superimposing Bernstein polynomials, shown in Fig. 1, on the initial geometry definition. The CST formulation also provides the analytical definition of V_n. Figure 2-(a) shows the aerodynamic grid around the $NACA0012$ aerofoil and Fig. 2-(b) the respective pressure field. The O-type grid contains 121×41 points in the circumferential and radial directions respectively, resulting in the fluid state $\mathbf{w}$ of dimension $N = 19844$.

Algorithm 1 ROM for Shape Optimization

Input: Initial geometry, initial parameters values, ROM update threshold ε, ROM maximum order, Δ_0

Output: Optimum shape, fluid state $\mathbf{w}$

Initialize:
  compute snapshot using Eq. 14, compute Φ with Eq. 17
while $\Delta_k > \Delta_{min}$ **do**
  solve optimisation subproblem, Eq. 21
  if Fluid residual $L_2 > \varepsilon$ **or** $||\nabla_{ROM}|| < \varepsilon_\nabla$ **then**
    compute ρ
    if $\rho < \eta_1$ **then**
      reject step, update $\Delta_k(\eta_1)$
    else
      accept step, update $\Delta_k(\eta_1, \eta_2)$
      $\mu_{k+1} \leftarrow \mu_k + s$
      compute new snapshot using Eq. 14, enrich basis, update Φ
      compute $\nabla\hat{\mathcal{F}}(\mu_{k+1})$ if corrections are required.
    end if
  else
    accept step, increase Δ_k
    $\mu_{k+1} \leftarrow \mu_k + s$
  end if
  if $|\mu_{k+1} - \mu_k| < \varepsilon_\mu$ **then**
    stop
  end if
end while

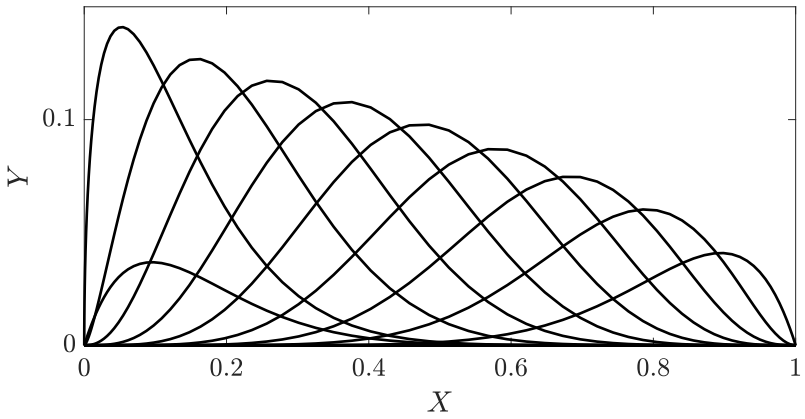


Figure 1: Bernstein polynomials used to parameterise the geometry.

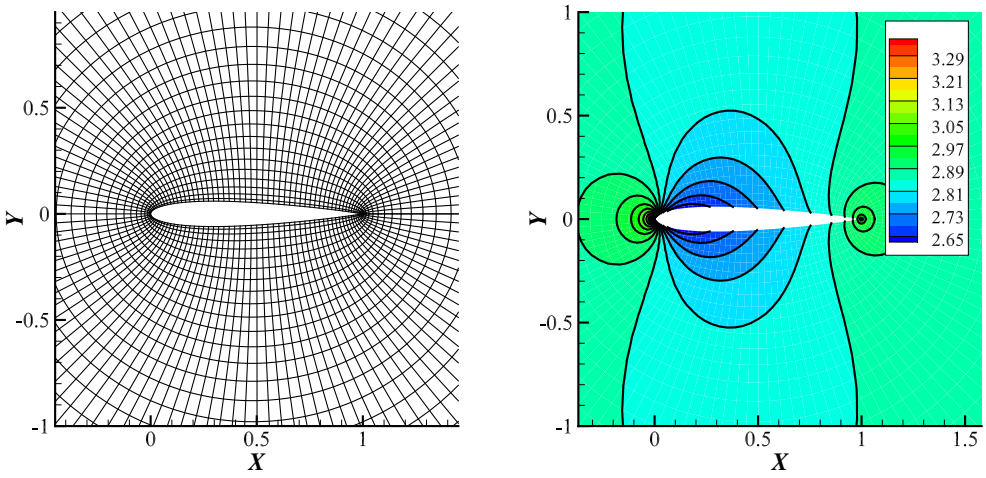


Figure 2: (a) aerodynamic grid of $NACA0012$ aerofoil; (b) non-dimensional pressure, $\hat{p}$, flow field at $(M_\infty, \alpha) = (0.5, 0)$.

Table 1: Relative cost to compute the ROM components, normalized by the wall-clock time required to solve the steady state once.

$\bar{\mathbf{w}} + \Phi$	$\mathbf{J}\Phi$	$(\frac{\partial p}{\partial \mathbf{w}}, \frac{\partial \mathbf{R}}{\partial \mathbf{S}})$	$\mathbf{w_r}$
1.75	0.05	0.12	0.25–0.4

The grid size is normalized by the aerofoil chord length c. Following Eq. 19, the pressure gradient with respect to the design parameters is given by

$$\frac{dp}{d\mu} = -\frac{\partial p}{\partial \mathbf{w}} \mathbf{J}^{-1} \frac{\partial \mathbf{R}}{\partial \mathbf{S}} V_n, \tag{27}$$

$$\approx -\frac{\partial p}{\partial \mathbf{w}} \Phi \left[\mathbf{J}\Phi\right]^{+} \frac{\partial \mathbf{R}}{\partial \mathbf{S}} V_n. \tag{28}$$

The subspace Φ is constructed using POD on the snapshot matrix defined by Eq. 14. The process starts with 20 ROBs being retained for the first optimization iteration, and is limited to 40 ROBs for the remainder iterations. Table 1 shows the wall clock time required to build the ROM and reconstruct the fluid state. It is worth noting that the cost of building Φ requires computing the steady state, i.e. the reference state $\bar{\mathbf{w}}$, extracting the fluid Jacobian and solving the n_p linear system of Eq. 4. The cost of evaluating $\mathbf{w_r}$ increases with the addition of basis to the ROM.

Figure 3 shows the gradient calculated for the initial conditions using finite differencing (FD) and the adjoint FOM (Eq. 6), together the with the ROM prediction obtained from Eq. 28. All methods are in excellent agreement with each other, indicating the Carter condition is respected at the centre of the trust region. The trust-region setup for this problem avoids the use of corrections and assumes the gradient approximation satisfies the Carter condition, results in Table 2 show the Carter condition is indeed met

Table 2: Carter condition parameter, ζ, and trust-region effectiveness, ρ.

ROM update	ζ	ρ
1	0.1317	–
2	0.0648	1.0570
3	0.0686	0.7916
4	0.0262	1.0312
5	0.0307	1.0226
6	0.0386	0.9949
7	0.0304	0.9635
8	0.0447	0.9702
9	0.0280	1.0221
10	0.3216	0.9697
11	0.2381	0.9462

for all iterations.

The threshold ε determines when Φ is updated, a smaller value of ε results in more frequent updates, hence more FOM calls, as shown by Fig. 4-(a). A ROM with $\varepsilon = 10^{-5}$ requires three updates. However, it is worth noting the threshold is case dependent and must be defined a priori by the user. In this case, the a value of $\varepsilon = 10^{-7}$ is used to complete the assessment of the ROM. The gradient norm predicted by the ROM at the centre of each trust-region is compared to the FOM gradient in Fig. 4-(b), indicating the level of agreement shown in Fig. 3 is maintained throughout the optimisation trajectory.

The optimisation trajectory and final geometry obtained are shown in Fig.

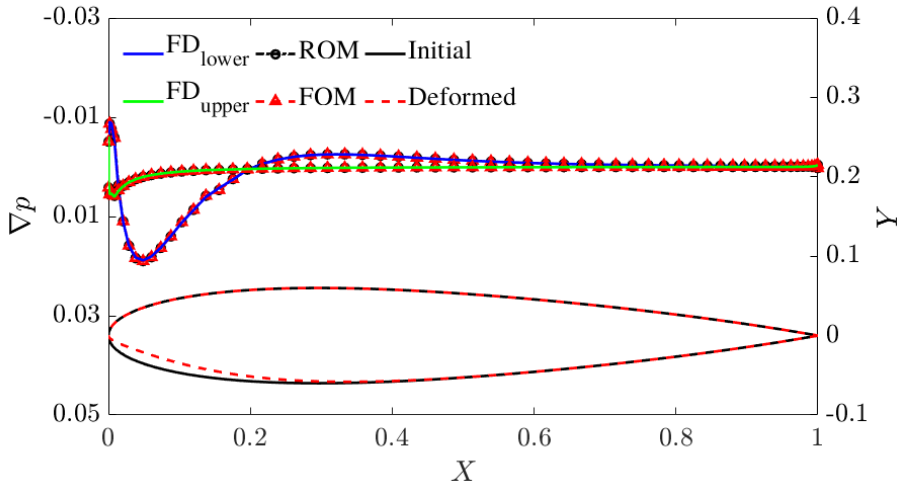


Figure 3: Pressure gradient with respect to the first weight comparison. The Finite-Differencing (FD) result is computed by perturbing the first weight with a step-size $\delta = 0.01$. The deformed shape is enlarged by 10 times for better visualization. The ROM used 20 basis.

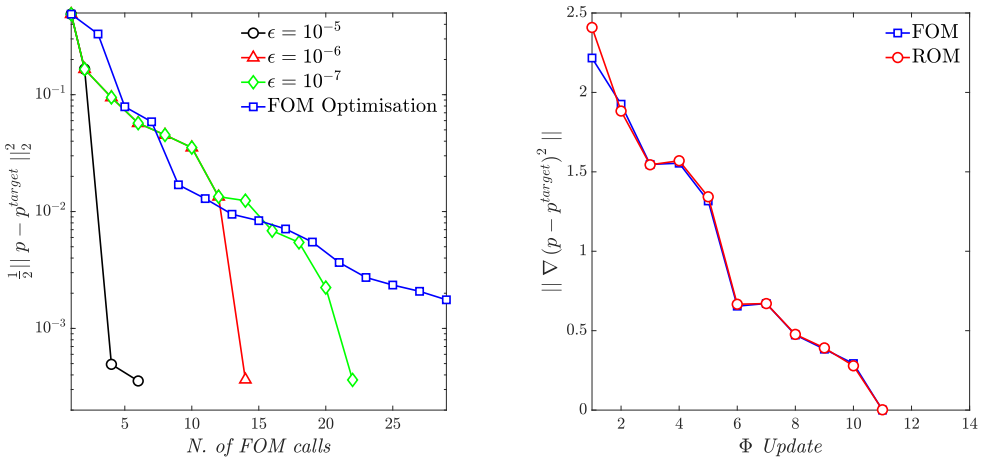
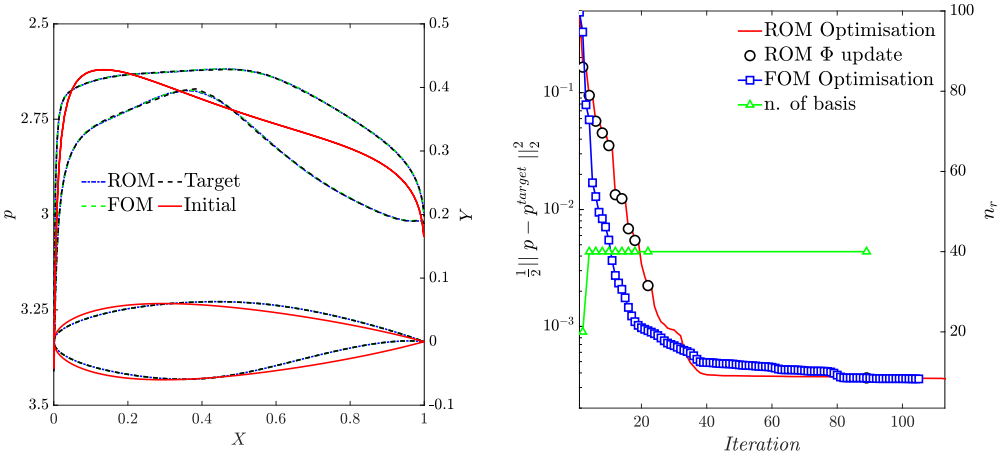


Figure 4: (a) Convergence of the objective function for different residual thresholds, ε; (b) objective function gradient norm evaluated at the centre of each trust-region.

5 using the residual threshold of $\varepsilon = 10^{-7}$. Results show the ROM converging to the FOM optimum, with both methods requiring just over 100 iterations to reach the optimum. As shown in table. 3, the ROM requires 11 steady state evaluation and linear snapshots to be collected to match the target pressure, which compares favorably with respect to the FOM optimization. The cost of using the ROM to evaluate the objective function and gradient is not trivial, the ROM iterations require the evaluation of $\mathbf{J\Phi}$, which dominates the computational effort of each iteration (about 95%), the remainder operations to compute $\mathbf{w_r}$ are two orders of magnitude faster using QR decomposition MatLab function *qr*[25]; in the end, a reduction to approximately 70% of the total time required by the FOM based optimisation was achieved, when using $\varepsilon = 10^{-7}$. The pressure fields obtained from both the final geometries are compared in Fig. 6, which further underlines the ROM accuracy. The



(a) (b)

Figure 5: (a) Optimized shape and the pressure distribution compared with target *RAE*2822 and associated pressure. (b) The objective function convergence history and retained number of ROBs.

fluid system is basically linear or rather weakly nonlinear, this resulted in an optimisation trajectory without any rejected steps, despite the ever increasing trust-region radius.

5.2. Transonic Flow Drag minimization

The *RAE*2822 aerofoil constrained drag minimization at $(M_\infty, \alpha) = (0.73, 2^\circ)$ problem is adopted to evaluate the proposed ROM for aerodynamic shape optimization in the transonic regime. The constrained optimisation problem is defined as follows:

$$\begin{aligned} &\underset{\mu \in \mathscr{D}}{\text{minimize}} \quad C_d \\ &\text{subject to} \quad \mathbf{R}(\mathbf{w}(\mu), \mu) = 0 \\ &\qquad\qquad (C_l, C_m) = (C_l, C_m)_{RAE2822} \end{aligned} \tag{29}$$

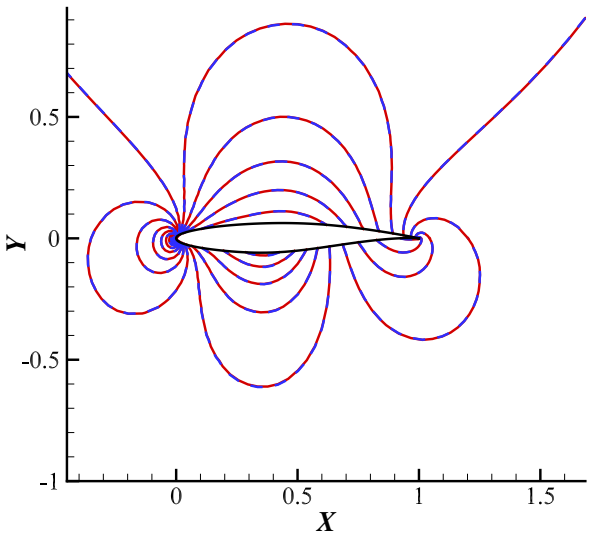
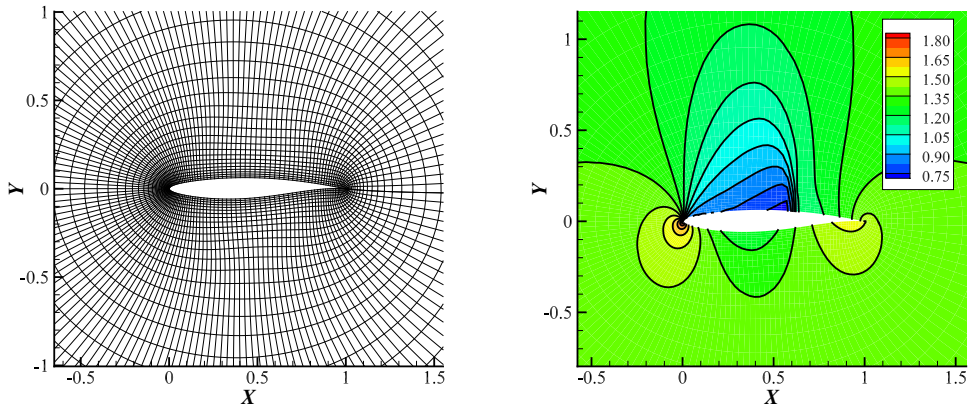


Figure 6: Pressure field comparison between FOM and ROM ($\varepsilon = 10^{-7}$) final aerofoils. The solid and dashed lines represent ROM and FOM, respectively. The ROM result is produced by the FOM using with the aerofoil shape from the ROM optimization.

Table 3: Performance and resource usage - comparison of FOM and ROM based inverse design problem.

	FOM Optimisation		ROM Optimisation	
	n. evals.	wall clock [s]	n. evals.	wall clock [s]
FOM steady state	105	1039	12	119
FOM adjoint state	105	1040	-	-
Update Φ:	-	-	11	77
ROM	-	-	242	762
Total Wall Clock:	2079		958	
$\frac{1}{2}\|p(w,\mu) - p_{RAE2822}\|_2^2$:	3.56×10^{-4}		3.53×10^{-4}	



(a) (b)

Figure 7: (a) aerodynamic grid of $RAE2822$ aerofoil and (b) pressure flow field at $(M_\infty, \alpha) = (0.73, 2^\circ)$.

As in the previous case, a CST parameterization with 20 weights ($n_p = 20$) is used to control the shape. As shown in Fig. 7-(a), the solutions are obtained on an O-type grid with 161×41 points in the circumferential and radial directions, respectively. The resultant fluid state $\mathbf{w}$ has a dimension of $N = 26404$. The initial pressure flow field is shown in Fig. 7-(b), which exhibits a shock just aft of the mid chord on the upper surface.

The gradients of the objective function, lift and pitching moment constraints are compared in Fig. 8, showing excellent agreement between the different methods available to compute the gradients. For the constrained optimisation problem, it was advantageous to include the corrections described by Eq. 22. Although numerical experiments show the problem reaches consistent converged solutions without corrections, the addition of these sped up convergence and yielded further reductions of the objective function.

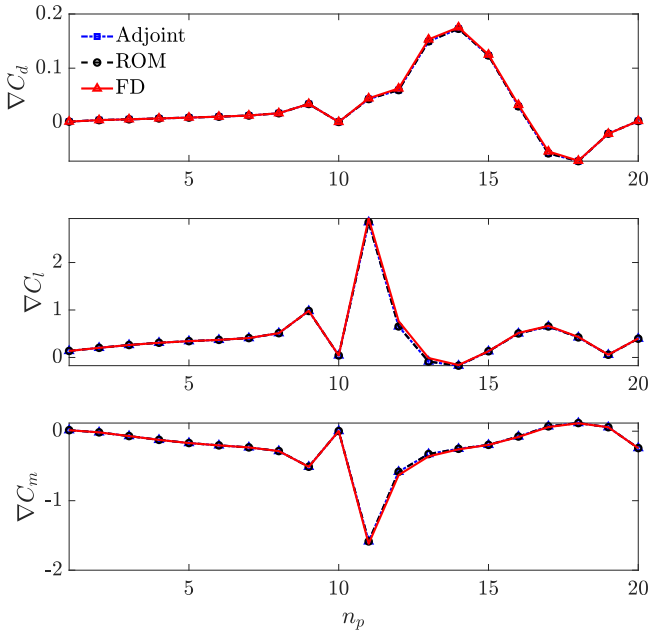


Figure 8: Constrained optimisation gradient comparison - the finite-difference result is computed by perturbing the design parameters with a step-size $\delta = 0.001$. The ROM used 20 basis.

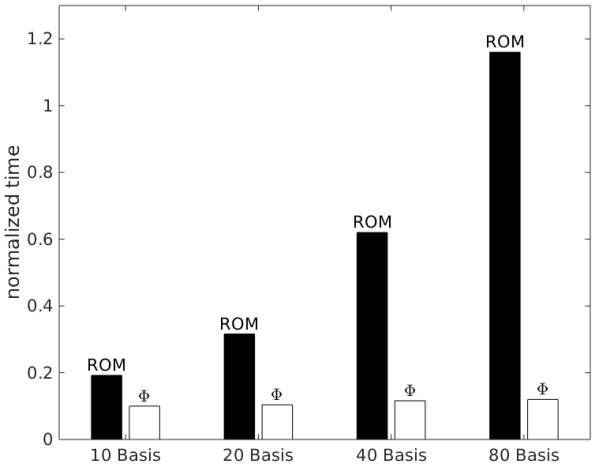
The presence of a strong nonlinearity in this problem required the use of a smaller residual threshold, $\varepsilon = 10^{-8}$, to produce meaningful solutions; however, the total number of Φ updates remains similar to the previous problem. The convergence of the ROM based optimisation is compared to the FOM using the adjoint method to compute the gradient in Fig. 10-(a). As in the previous test, the ROM is able to find a minimum close to the FOM based optimisation solution. Reducing the number of ROBs used to build the ROM deteriorates the convergence rate as illustrated in Fig. 10-(b). Nevertheless, the final solution is identical to those retaining 40 or 80 ROBs. Increasing the number of ROBs beyond 40 appears to have a limited impact on the solution.

The number of FOM calls for both optimisation strategies plotted in Fig. 10-(a) are shown in Table. 4, together with wall clock times. Using the matrix factorisation method to solve the adjoint systems greatly reduces the solution time required to obtain the gradients for the FOM optimisation with respect to the flow solution. For the ROM based optimisation, and as before, the number of FOM calls is reduced by almost a quarter. Despite the number of FOM calls reducing significantly for the ROM problem, the cost of evaluating the ROM using 40 basis leads only to a marginal reduction in wall-clock times for both strategies. However, without impacting the quality of final solution, it is possible to solver this problem retaining 20 ROBs, which yields more significant savings. To understand the impact of the number of ROB and design variables on the performance of this ROM based optimisation, the computational time to evaluate a functional with the ROM and updating Φ is compared in Fig. 9. For a given number of

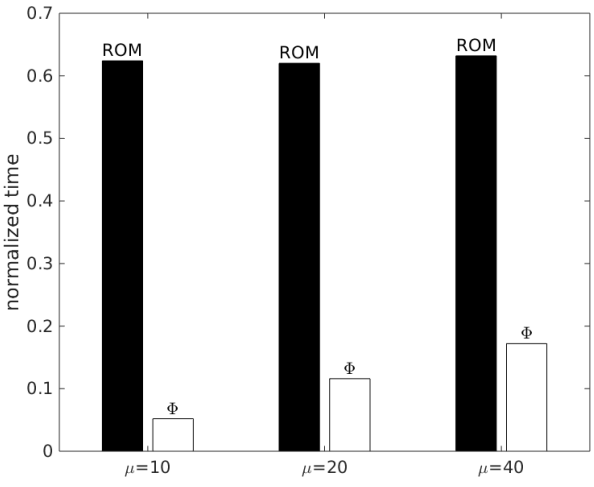
design parameters, increasing number of ROBs has a dramatic effect on the effort required by the ROM - Fig. 9-(a),this stems mainly from the need to evaluate the reduced Jacobian $\mathbf{J}\phi$, which is an operation $\propto (N \times n_r)$; on the other hand, the cost of evaluating the ROM is independent of the number of design parameters, the cost of updating Φ, for a fixed number of ROBs, increases linearly with the number of design parameters, as shown in Fig. 9-(b). It is worth pointing out that the number of basis required to solve a problem adequately is dependent on the number of design parameters and as shown by the results, increasing the number of basis beyond 40 will not result in significant improvements over the FOM approach. As noted in [39], this excessive cost in evaluating functionals using the ROM can be mitigated by employing hyper-reduction techniques. Figure 11 shows the solutions corresponding to the minima found by the FOM and ROM optimisations. The results show that even for problems exhibiting nonlinearities such as shocks, the ROM based strategy is able to produce similar results to those obtained with conventional FOM optimisation strategies.

6. Conclusion

A ROM is developed for gradient based aerodynamic shape optimization with reduced Newton's iterations. Auto-Differentiation is used to evaluate the reduced Jacobian without forming the full fluid Jacobian explicitly during the reduced Newton's iterations. A sampling procedure based on the solution of linear system of equations, is adopted to construct the subspace. The procedure only requires solving linear systems with the number of design parameters without resorting to sample for new fluid states for each



(a)



(b)

Figure 9: computational effort for the ROM steady state solution and updating the Φ: (a) as a function of the number of ROBs retained, for 20 design parameters; (b) as a function of the number of design parameters, whilst retaining 40 ROBs. Time is normalized with respect to cost one steady state solution.

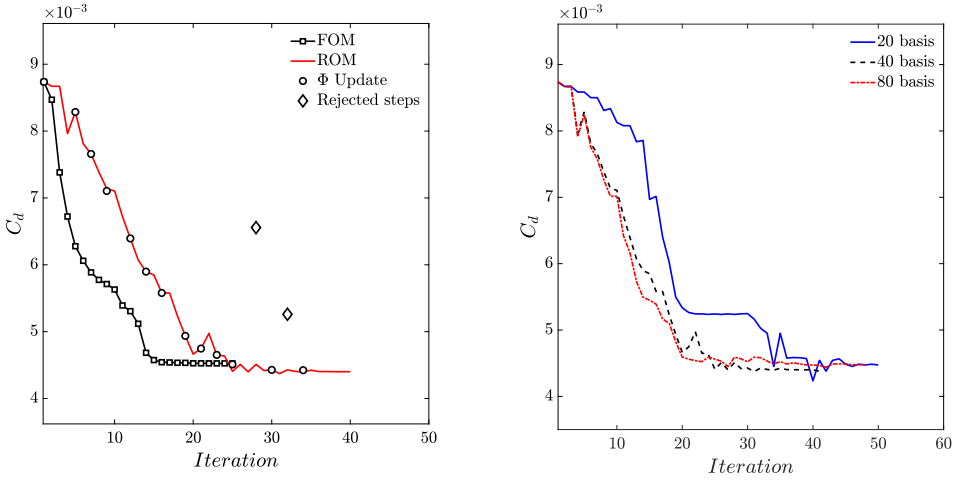


Figure 10: (a) ROM and FOM based optimisation convergence history; (b) impact of number of ROBs on optimisation convergence.

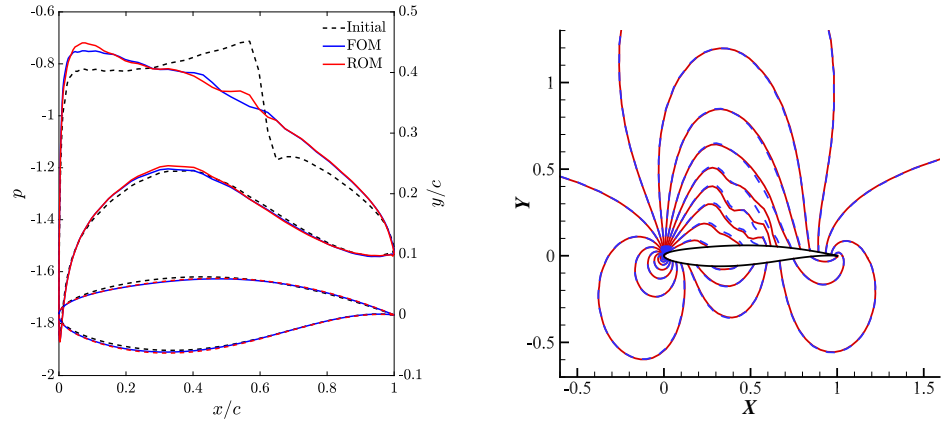


Figure 11: Pressure field comparison between FOM and ROM ($\varepsilon = 10^{-8}$). The ROM result is obtained by FOM steady state solution with the aerofoil shape from the ROM optimization. The solid and dashed lines represent ROM and FOM, respectively.

Table 4: Performance and resource usage - comparison of FOM and ROM based transonic drag minimization.

	FOM		ROM 20 ROB		ROM 40 ROB	
	n. evals.	time[s]	n. evals.	time[s]	n. evals.	time[s]
FOM steady state	74	1952	14	364	15	416
FOM adjoint state	18	242	-	-	-	-
Update Φ:	-	-	8	21	13	42
ROM	-	-	142	1169	89	1379
Total wall clock time:	2194		1554		1825	
C_d reduction:	48.10%		48.17%		48.33%	

design variable, which quickly becomes prohibitive for large number of design parameters. It was demonstrated that the proposed ROM is suitable for subsonic inverse design and transonic flow drag minimization problems. The inverse design optimization problem can be solved with fewer basis updates than the transonic case, as the latter represents a strongly nonlinear system. Even for the more demanding test case, it was possible to reduce the number of FOM analysis by a factor of four, however this was not translated into significant time savings.

Acknowledgments

The funding provided for this research from the Engineering and Physical Sciences Research Council (Grant No. EP/P025692/1) is gratefully acknowledged.

[1] Agarwal, A., Biegler, L.T., 2013. A trust-region framework for constrained optimization using reduced order modeling. Optimization and Engineering 14, 3–35. doi:10.1007/s11081-011-9164-0.

[2] Agarwal, D., Robinson, T.T., Armstrong, C.G., Marques, S., Vasilopoulos, I., Meyer, M., 2018. Parametric design velocity computation for cad-based design optimization using adjoint methods. Engineering with Computers 34, 225–239. doi:10.1007/s00366-017-0534-x.

[3] Alexandrov, N.M., Lewis, R.M., 2003. First-order approximation and model management in optimization, in: Biegler, L.T., Heinkenschloss, M., Ghattas, O., van Bloemen Waanders, B. (Eds.), Large-Scale PDE-

Constrained Optimization, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 63–79.

[4] Ammar, A., Mokdad, B., Chinesta, F., Keunings, R., 2006. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Journal of Non-Newtonian Fluid Mechanics 139, 153 – 176.

[5] Amsallem, D., Farhat, C., Haasdonk, B., . Special issue on model reduction. International Journal for Numerical Methods in Engineering 102, 931–932. doi:10.1002/nme.4889.

[6] Benner, P., Gugercin, S., Willcox, K., 2015. A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Review 57, 483–531.

[7] Bergmann, M., Cordier, L., 2008. Optimal control of the cylinder wake in the laminar regime by trust-region methods and pod reduced-order models. Journal of Computational Physics 227, 7813 – 7840. doi:10.1016/j.jcp.2008.04.034.

[8] Bui-Thanh, T., Willcox, K., Ghattas, O., 2008. Model reduction for large-scale systems with high-dimensional parametric input. SIAM Journal on Scientific Computing 30, 3270–3288. doi:10.1137/070694855.

[9] Carter, R.G., 1991. On the global convergence of trust region algorithms using inexact gradient information. SIAM Journal on Numerical Analysis 28, 251–265.

[10] Conn, A., Gould, N., Toint, P., 2000. Trust Region Methods. MPS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics.

[11] Conn, A.R., Gould, N., Sartenaer, A., Toint, P.L., 1993. Global convergence of a class of trust region algorithms for optimization using inexact projections on convex constraints. SIAM Journal on Optimization 3, 164–221.

[12] Elliott, J., Peraire, J., 1997. Practical three-dimensional aerodynamic design and optimization using unstructured meshes. AIAA journal 35, 1479–1485.

[13] Giles, M.B., Pierce, N.A., 2000. An introduction to the adjoint approach to design. Flow, Turbulence and Combustion 65, 393–415.

[14] Hascoët, L., Pascual, V., 2013. The Tapenade Automatic Differentiation tool: Principles, Model, and Specification. ACM Transactions On Mathematical Software 39.

[15] Iuliano, E., Quagliarella, D., 2013. Proper orthogonal decomposition, surrogate modelling and evolutionary optimization in aerodynamic design. Computers & Fluids 84, 327–350.

[16] Jameson, A., 1988. Aerodynamic design via control theory. Journal of scientific computing 3, 233–260.

[17] Jameson, A., Martinelli, L., Pierce, N., 1998. Optimum aerodynamic design using the navier–stokes equations. Theoretical and computational fluid dynamics 10, 213–237.

[18] Kevin, C., Charbel, B., Charbel, F., 2011. Efficient non-linear model reduction via a least-squares petrov-galerkin projection and compressive tensor approximations. International Journal for Numerical Methods in Engineering 86, 155–181.

[19] Kulfan, B.M., 2008. Universal parametric geometry representation method. Journal of Aircraft 45, 142 – 158.

[20] Lassila, T., Manzoni, A., Quarteroni, A., Rozza, G., 2014. Model order reduction in fluid dynamics: challenges and perspectives, in: Reduced Order Methods for modeling and computational reduction. Springer, pp. 235–273.

[21] LeGresley, P., Alonso, J., 2000. Airfoil design optimization using reduced order models based on proper orthogonal decomposition, in: Fluids 2000 conference and exhibit, p. 2545.

[22] Lucia, D.J., Beran, P.S., Silva, W.A., 2004. Reduced-order modeling: new approaches for computational physics. Progress in Aerospace Sciences 40, 51 – 117. doi:10.1016/j.paerosci.2003.12.001.

[23] Manzoni, A., Quarteroni, A., Rozza, G., 2012. Shape optimization for viscous flows by reduced basis methods and free−form deformation. International Journal for Numerical Methods in Fluids 70, 646–670.

[24] Martins, J.R.R.A., Hwang, J.T., 2013. Review and unification of methods for computing derivatives of multidisciplinary computational models. AIAA Journal 51, 2582–2599.

[25] MATLAB, 2018. Matlab, version 2018b. The MathWorks, Natick, MA, USA.

[26] MATLAB Optimization Toolbox, 2018. Matlab, version 2018b. The MathWorks, Natick, MA, USA.

[27] Negri, F., Manzoni, A., Rozza, G., 2015. Reduced basis approximation of parametrized optimal flow control problems for the stokes equations. Computers and Mathematics with Applications 69, 319 – 336.

[28] Nielsen, E.J., Anderson, W.K., 1999. Aerodynamic design optimization on unstructured meshes using the navier-stokes equations. AIAA journal 37, 1411–1419.

[29] Noack, B.R., Morzynski, M., Tadmor, G., 2011. Reduced-order modelling for flow control. volume 528. Springer Science & Business Media.

[30] Pironneau, O., 1984. Optimal Shape Design for Elliptic Systems. Springer series in computational physics, Springer-Verlag. URL: https://books.google.co.uk/books?id=6JqyAAAAIAAJ.

[31] Qian, E., Grepl, M., Veroy, K., Willcox, K., 2017. A certified trust region reduced basis approach to pde-constrained optimization. SIAM Journal on Scientific Computing 39, S434–S460.

[32] Rowley, C.W., 2005. Model Reduction for Fluids, using Balanced Proper Orthogonal Decomposition. International Journal of Bifurcation and Chaos 15, 997–1013.

[33] Sirovich, L., 1987. Turbulence and the dynamics of coherent structures. i: Coherent structures. ii: Symmetries and transformations. iii: Dynamics and scaling. Quarterly of Applied Mathematics 45, 583–590.

[34] Skinner, S., Zare-Behtash, H., 2018. State-of-the-art in aerodynamic shape optimisation methods. Applied Soft Computing 62, 933 – 962.

[35] Willcox, K., Peraire, J., 2002. Balanced model reduction via the proper orthogonal decomposition. AIAA journal 40, 2323–2330.

[36] Yao, W., Marques, S., 2015. Prediction of transonic limit-cycle oscillations using an aeroelastic harmonic balance method. AIAA Journal 53, 2040–2051. doi:10.2514/1.J053565.

[37] Yondo, R., Andrés, E., Valero, E., 2018. A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses. Progress in Aerospace Sciences 96, 23–61.

[38] Yue, Y., Meerbergen, K., 2013. Accelerating optimization of parametric linear systems by model order reduction. SIAM Journal on Optimization 23, 1344–1370.

[39] Zahr, M., Charbel, F., 2015. Progressive construction of a parametric reduced−order model for PDE−constrained optimization. International Journal for Numerical Methods in Engineering 102, 1111–1135.