

1 A Reduced-order Model for Gradient-based
2 Aerodynamic Shape Optimisation[☆]

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

5 ^a*Faculty of computing, Engineering and Media, De Montfort University, Leicester,*
6 *LE1,9BH, U.K.*

7 ^b*Department of Mechanical Engineering Sciences, The University of Surrey, Guildford,*
8 *England, GU2 7XH U.K.*

9 ^c*School of Mechanical and Aerospace Engineering, Queen's University Belfast, Belfast,*
10 *Northern Ireland, BT9 5AH, U.K.*

11 ^d*School of Computing, Engineering and Intelligent Systems, Ulster University, Northern*
12 *Ireland, BT48 7JL, U.K.*

13 **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

Preprint submitted to *Journal of Computational Physics* (Simão Marques)

August 17, 2020

¹Senior Lecturer

²Reader

³Professor

⁴Lecturer

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.

¹⁴ *Keywords:* ROM, CFD, Aerodynamics, Shape Optimisation,

¹⁵ Gradient-based Optimisation, Trust-Region, Multi-fidelity

16 **Nomenclature**

Latin Symbols

C_l	=	lift coefficient
C_d	=	drag coefficient
$C_{\mathcal{E}}$	=	equality constraint
$C_{\mathcal{I}}$	=	inequality constraint
C_m	=	pitching moment coefficient
D	=	diagonal matrix containing singular values
\mathcal{F}	=	objective function
J	=	fluid Jacobian matrix
lb	=	design parameter lower bound values
M_{∞}	=	free-stream Mach number
$MUSCL$	=	Monotonic Upwind Scheme for Conservation Laws
¹⁷ N	=	number of degrees of freedom
n_p	=	number of design parameters
n_r	=	number of basis
p	=	static pressure
\hat{p}	=	non-dimensional static pressure, $\hat{p} = \frac{p}{q_{\infty}}$
p	=	line search vector for Newton method
q	=	dynamic pressure
R	=	vector of fluid equations residuals
S	=	surface mesh points
s	=	optimisation step
t	=	time
¹⁸ 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
¹⁹ 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
²⁰ Ψ	=	reduced Jacobian matrix

21 1. Introduction

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

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

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

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

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

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

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

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

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

112 **2. Shape Optimization Using a Full-Order Model**

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

120

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

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

125 where \mathcal{F} is the objective function, \mathbf{w} and μ are fluid state and design pa-
126 rameters, respectively, $D \subset \mathbb{R}^{n_p}$ represents the parameter space, \mathbf{R} is the
127 residual of the Euler equations and $\mathcal{C}_{\mathcal{E}}$, $\mathcal{C}_{\mathcal{I}}$ are equality and inequality con-
128 straints, respectively. To solve for the fluid state variables, the fluid residual
129 $\mathbf{R}(\mathbf{w}(\mu), \mu)$ is driven to zero.

130

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

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

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

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

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

136 following linear system

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

137 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
 138 degrees of freedom of the fluid state. Substituting Eq. 4 into Eq. 2, results
 139 in

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

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

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

148 where Λ^T is the adjoint vector, obtained by solving the following linear (ad-
 149 joint) system,

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

150 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 ob-
 151 tained through auto-differentiation of the discretized Euler equations using
 152 the Tapenade library [14]. The Jacobian is typically a large, sparse matrix
 153 where only the non-zeros entries are computed. For small problems, matrix

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

157 The adjoint system scales with the number of functions in Eq. 1, there-
158 fore, is the preferred choice for aerodynamic shape optimisation problems,
159 where the number of parameters, n_p , usually outnumber the number of ob-
160 jective and constraint functions. In this work, the optimisation results based
161 solely on the FOM are used as reference and are solved using the Sequential
162 Quadratic Programming (SQP) algorithm from MatLab’s *fmincon* function
163 [26], with the gradient built using the solution of the adjoint system repre-
164 sented in Eq. 7.

165 **3. Shape Optimization Using a Reduced-Order Model**

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

175 *3.1. FOM Reduction*

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

178 i.e.:

$$\mathbf{w} \approx \bar{\mathbf{w}} + \Phi \mathbf{w}_r, \quad (8)$$

179 where $\bar{\mathbf{w}}$ is the operation reference state and \mathbf{w}_r is the reduced state vector.

180 By Substituting Eq. 8 into the fluid residual \mathbf{R} , the FOM is converted to the

181 following optimization problem,

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

182 This type of ROM is equivalent to the minimum residual approach, meaning

183 that if the ROB is enriched, the solution error in the $\mathbf{J}^T \mathbf{J}$ - *norm* is non-

184 increasing [8]. The first order optimality condition gives

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

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:

$$\Phi^T \mathbf{J}_k^T \mathbf{J}_k \Phi \mathbf{p}_k = -\Phi^T \mathbf{J}_k^T \mathbf{R}_k \quad (11)$$

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

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

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

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

193 3.2. Construction and Enrichment of Reduced Order Basis

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

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

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

$$\frac{\partial \mathbf{R}}{\partial \mu} = \frac{\partial \mathbf{R}}{\partial \mathbf{S}} \frac{\partial \mathbf{S}}{\partial \mu} \quad (15)$$

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

203 or third party CAD systems [2]; in this work the mesh is deformed following
 204 the transfinite interpolation method and the term $\frac{\partial \mathbf{R}}{\partial \mathbf{S}}$ which represents the
 205 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 :

$$\mathbf{X}^T \mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{V}^T, \quad (16)$$

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

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

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

217 3.3. Gradient Evaluation

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

$$\frac{d\mathbf{w}}{d\mu} \approx -\Phi [\mathbf{J}\Phi]^+ \frac{d\mathbf{R}}{d\mu}, \quad (18)$$

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

$$\frac{d\mathcal{F}}{d\mu} \approx \frac{\partial\mathcal{F}}{\partial\mu} - \frac{\partial\mathcal{F}}{\partial\mathbf{w}}\Phi[\mathbf{J}\Phi]^+ \frac{d\mathbf{R}}{d\mu}. \quad (19)$$

222 Eq. 19 can be expressed as the following by introducing the design velocity
 223 $V_n = \frac{d\mathbf{S}}{d\mu}$, which is the surface mesh derivative with respect to the design
 224 parameter μ [2],

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

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

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

234 4. Trust-Region Framework

As the name indicates, the trust-region method aims to establish a sub-domain 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 :

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

235 where the “ $\hat{}$ ” symbol indicates quantities computed using the ROM (using
236 Eq. 8 for the functionals and Eq. 20 for the gradients); s is the optimisation
237 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.:

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

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

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

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

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

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

245 respect to s for all μ and continuous in μ for all s ; b) the set of problem
 246 parameters is closed and bounded; c) the sufficient decrease condition re-
 247 quires the step to satisfy the fraction of Cauchy decrease (FCD). In addition
 248 the second derivatives of the ROM at μ_k remain bounded within the trust
 249 region domain for all k [1]. The FCD condition was derived for the clas-
 250 sical trust-region method, where the FOM is approximated by a quadratic
 251 Taylor’s series expansion. With POD based approaches, this is no longer
 252 feasible, instead the low-fidelity function step is determined by solving Eq.
 253 21 using MatLab’s *fmincon* SQP algorithm, which maximizes the decrease in
 254 the objective function of the trust-region subproblem.

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

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

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

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

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

272 5. Results

273 5.1. Subsonic Flow Inverse Design

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

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

276 where p is the pressure distribution. Equation 26 aims to modify the aerofoil
 277 shape to match the target pressure distribution $p_{RAE2822}$ produced by an
 278 inviscid analysis of the *RAE2822* aerofoil, hence the optimum shape should
 279 correspond to the *RAE2822* aerofoil. The flow is described by the fluid
 280 Euler equations, the free-stream conditions are $(M_{\infty}, \alpha) = (0.5, 0^{\circ})$. The
 281 *NACA0012* aerofoil is used as the initial geometry. The CST method using
 282 20 weights ($n_p = 20$), is used to parameterize the aerofoil by superimposing
 283 Bernstein polynomials, shown in Fig. 1, on the initial geometry definition.
 284 The CST formulation also provides the analytical definition of V_n . Figure
 285 2-(a) shows the aerodynamic grid around the *NACA0012* aerofoil and Fig.
 286 2-(b) the respective pressure field. The O-type grid contains 121×41 points
 287 in the circumferential and radial directions respectively, resulting in the fluid
 288 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}

```
1: Initialize:
2:   compute snapshot using Eq. 14, compute  $\Phi$  with Eq. 17
3: while  $\Delta_k > \Delta_{min}$  do
4:   solve optimisation subproblem, Eq. 21
5:   if Fluid residual  $L_2 > \varepsilon$  or  $\|\nabla_{ROM}\| < \varepsilon_{\nabla}$  then
6:     compute  $\rho$ 
7:     if  $\rho < \eta_1$  then
8:       reject step, update  $\Delta_k(\eta_1)$ 
9:     else
10:      accept step, update  $\Delta_k(\eta_1, \eta_2)$ 
11:       $\mu_{k+1} \leftarrow \mu_k + s$ 
12:      compute new snapshot using Eq. 14, enrich basis, update  $\Phi$ 
13:      compute  $\nabla \hat{\mathcal{F}}(\mu_{k+1})$  if corrections are required.
14:    end if
15:  else
16:    accept step, increase  $\Delta_k$ 
17:     $\mu_{k+1} \leftarrow \mu_k + s$ 
18:  end if
19:  if  $|\mu_{k+1} - \mu_k| < \varepsilon_{\mu}$  then
20:    stop
21:  end if
22: end while
```

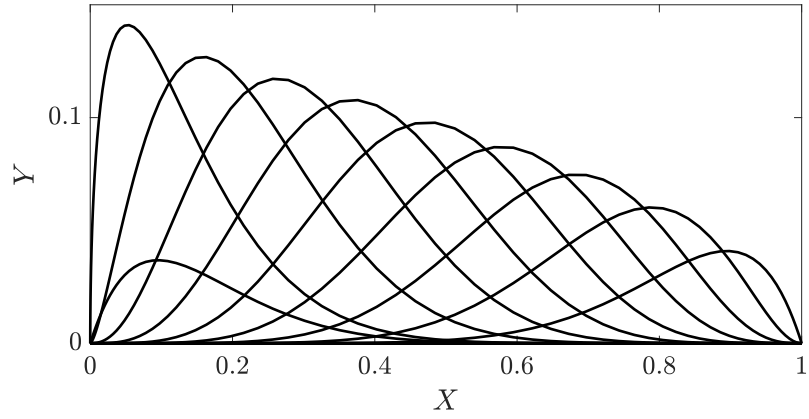


Figure 1: Bernstein polynomials used to parameterise the geometry.

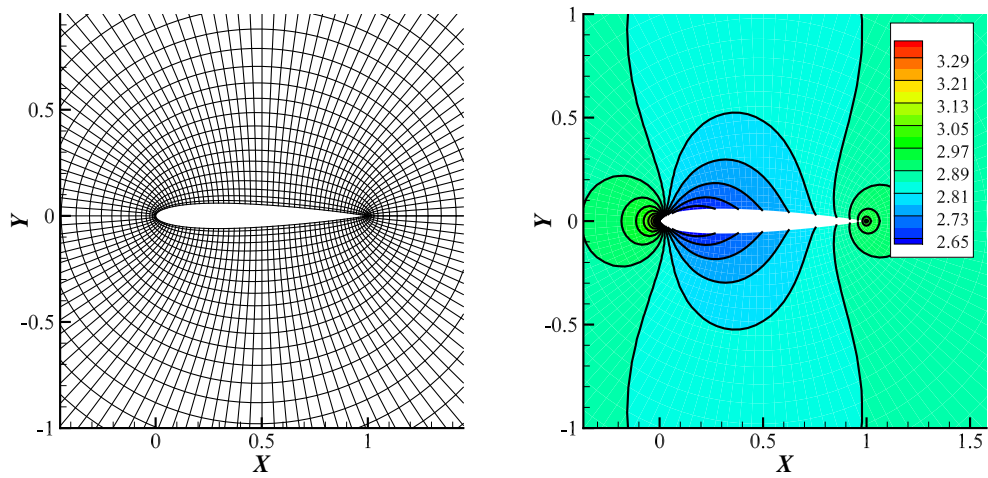


Figure 2: (a) aerodynamic grid of *NACA0012* aerofoil; (b) non-dimensional pressure, \hat{p} , flow field at $(M_\infty, \alpha) = (0.5, 0)$.

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

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

$$\approx -\frac{\partial p}{\partial \mathbf{w}} \Phi [\mathbf{J}\Phi]^+ \frac{\partial \mathbf{R}}{\partial \mathbf{S}} V_n. \quad (28)$$

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

297 Figure 3 shows the gradient calculated for the initial conditions using
 298 finite differencing (FD) and the adjoint FOM (Eq. 6), together the with
 299 the ROM prediction obtained from Eq. 28. All methods are in excellent
 300 agreement with each other, indicating the Carter condition is respected at
 301 the centre of the trust region. The trust-region setup for this problem avoids
 302 the use of corrections and assumes the gradient approximation satisfies the
 303 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

304 for all iterations.

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

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

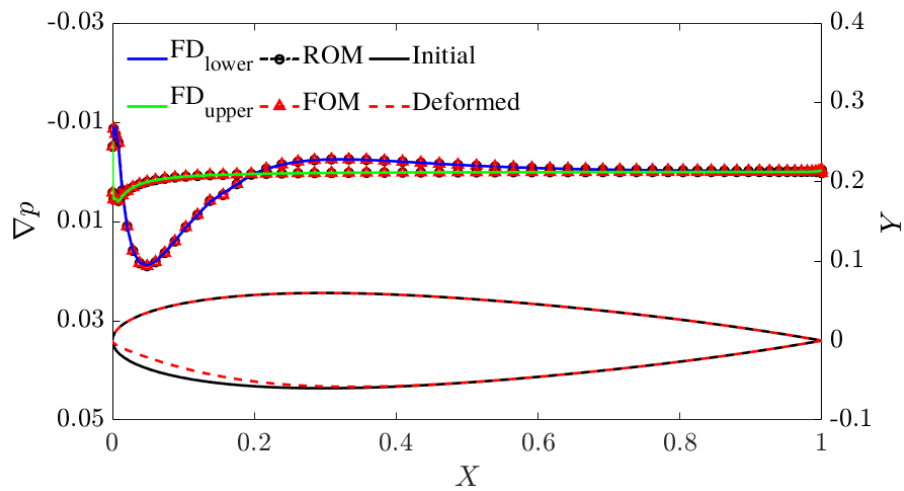


Figure 3: Pressure gradient with respect to the first weight comparison. The Finite-Differencing (FD) result is computed by perturbing the first weight with a step-size $\delta = 0.01$. The deformed shape is enlarged by 10 times for better visualization. The ROM used 20 basis.

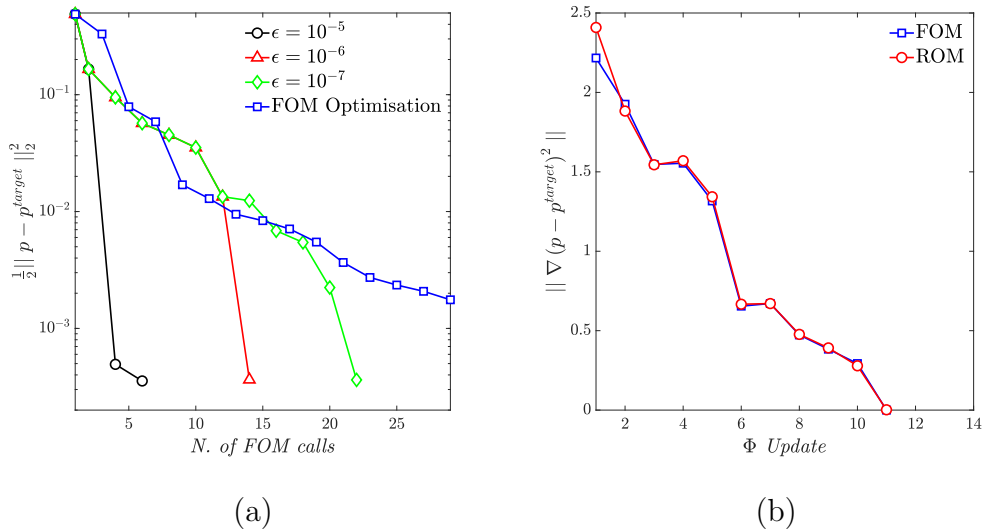


Figure 4: (a) Convergence of the objective function for different residual thresholds, ε ; (b) objective function gradient norm evaluated at the centre of each trust-region.

315 5 using the residual threshold of $\varepsilon = 10^{-7}$. Results show the ROM converging
316 to the FOM optimum, with both methods requiring just over 100 iterations to
317 reach the optimum. As shown in table. 3, the ROM requires 11 steady state
318 evaluation and linear snapshots to be collected to match the target pressure,
319 which compares favorably with respect to the FOM optimisation. The cost
320 of using the ROM to evaluate the objective function and gradient is not
321 trivial, the ROM iterations require the evaluation of $\mathbf{J}\Phi$, which dominates the
322 computational effort of each iteration (about 95%), the remainder operations
323 to compute \mathbf{w}_r are two orders of magnitude faster using QR decomposition
324 MatLab function *qr*[25]; in the end, a reduction to approximately 70% of
325 the total time required by the FOM based optimisation was achieved, when
326 using $\varepsilon = 10^{-7}$. The pressure fields obtained from both the final geometries
327 are compared in Fig. 6, which further underlines the ROM accuracy. The

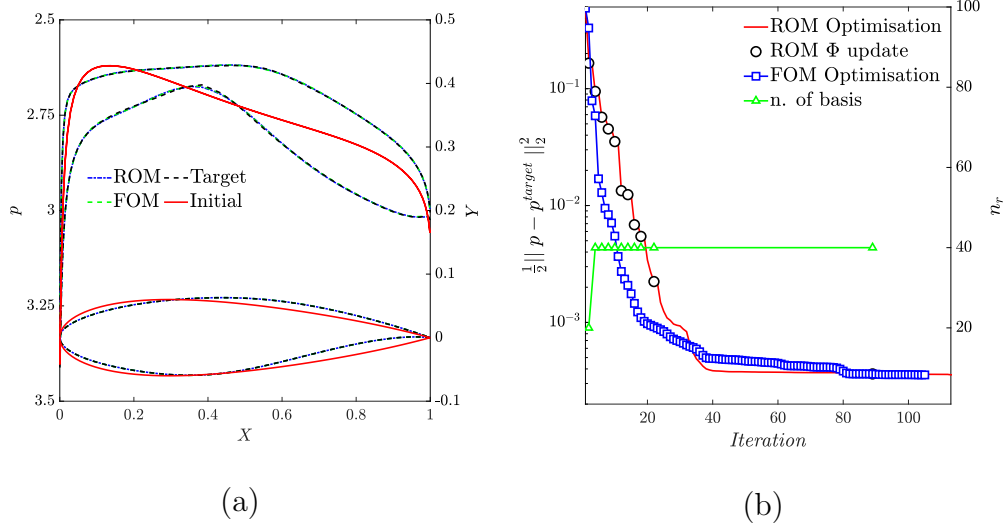


Figure 5: (a) Optimized shape and the pressure distribution compared with target *RAE2822* and associated pressure. (b) The objective function convergence history and retained number of ROBs.

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

331 5.2. Transonic Flow Drag minimization

332 The *RAE2822* aerofoil constrained drag minimization at $(M_\infty, \alpha) = (0.73, 2^\circ)$
 333 problem is adopted to evaluate the proposed ROM for aerodynamic shape
 334 optimization in the transonic regime. The constrained optimisation problem
 335 is defined as follows:

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

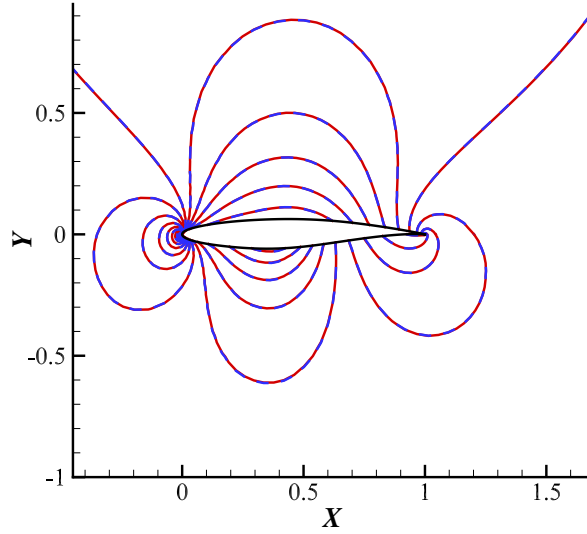


Figure 6: Pressure field comparison between FOM and ROM ($\varepsilon = 10^{-7}$) final aerofoils. The solid and dashed lines represent ROM and FOM, respectively. The ROM result is produced by the FOM using with the aerofoil shape from the ROM optimization.

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}

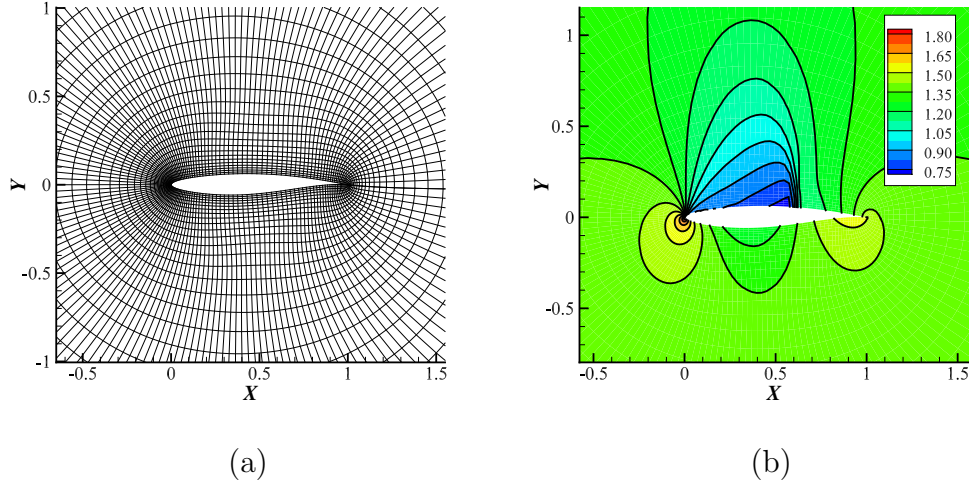


Figure 7: (a) aerodynamic grid of *RAE2822* aerofoil and (b) pressure flow field at $(M_\infty, \alpha) = (0.73, 2^\circ)$.

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

342 The gradients of the objective function, lift and pitching moment con-
 343 straints are compared in Fig. 8, showing excellent agreement between the
 344 different methods available to compute the gradients. For the constrained op-
 345 timisation problem, it was advantageous to include the corrections described
 346 by Eq. 22. Although numerical experiments show the problem reaches con-
 347 sistent converged solutions without corrections, the addition of these sped
 348 up convergence and yielded further reductions of the objective function.

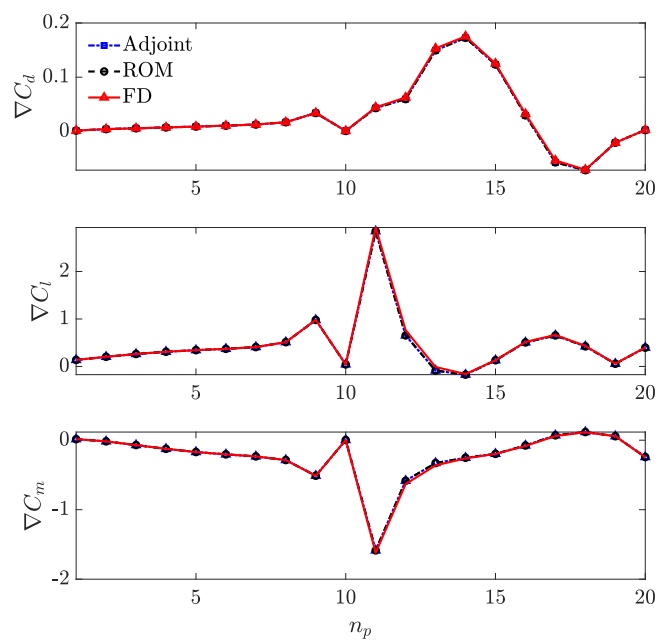


Figure 8: Constrained optimisation gradient comparison - the finite-difference result is computed by perturbing the design parameters with a step-size $\delta = 0.001$. The ROM used 20 basis.

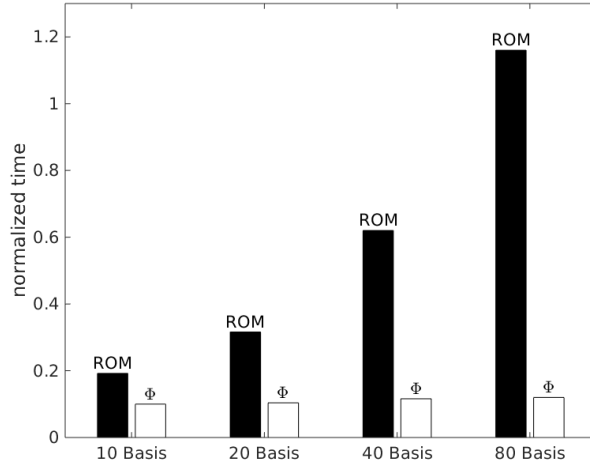
349 The presence of a strong nonlinearity in this problem required the use
350 of a smaller residual threshold, $\varepsilon = 10^{-8}$, to produce meaningful solutions;
351 however, the total number of Φ updates remains similar to the previous
352 problem. The convergence of the ROM based optimisation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

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

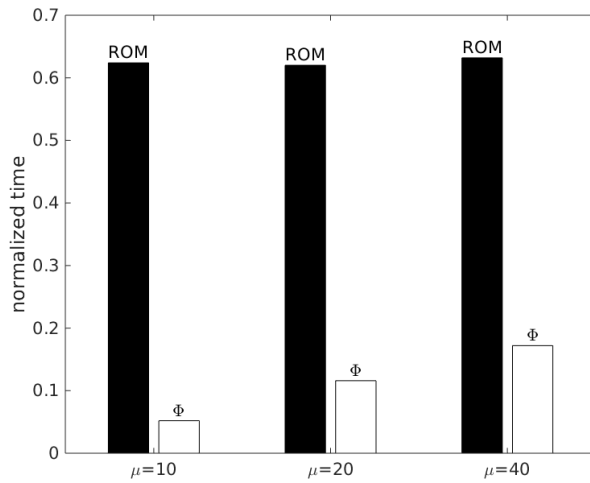
374 design parameters, increasing number of ROBs has a dramatic effect on the
375 effort required by the ROM - Fig. 9-(a), this stems mainly from the need to
376 evaluate the reduced Jacobian $\mathbf{J}\phi$, which is an operation $\propto (N \times n_r)$; on the
377 other hand, the cost of evaluating the ROM is independent of the number
378 of design parameters, the cost of updating Φ , for a fixed number of ROBs,
379 increases linearly with the number of design parameters, as shown in Fig.
380 9-(b). It is worth pointing out that the number of basis required to solve a
381 problem adequately is dependent on the number of design parameters and
382 as shown by the results, increasing the number of basis beyond 40 will not
383 result in significant improvements over the FOM approach. As noted in [39],
384 this excessive cost in evaluating functionals using the ROM can be mitigated
385 by employing hyper-reduction techniques. Figure 11 shows the solutions cor-
386 responding to the minima found by the FOM and ROM optimisations. The
387 results show that even for problems exhibiting nonlinearities such as shocks,
388 the ROM based strategy is able to produce similar results to those obtained
389 with conventional FOM optimisation strategies.

390 6. Conclusion

391 A ROM is developed for gradient based aerodynamic shape optimization
392 with reduced Newton's iterations. Auto-Differentiation is used to evaluate
393 the reduced Jacobian without forming the full fluid Jacobian explicitly dur-
394 ing the reduced Newton's iterations. A sampling procedure based on the
395 solution of linear system of equations, is adopted to construct the subspace.
396 The procedure only requires solving linear systems with the number of de-
397 sign 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.

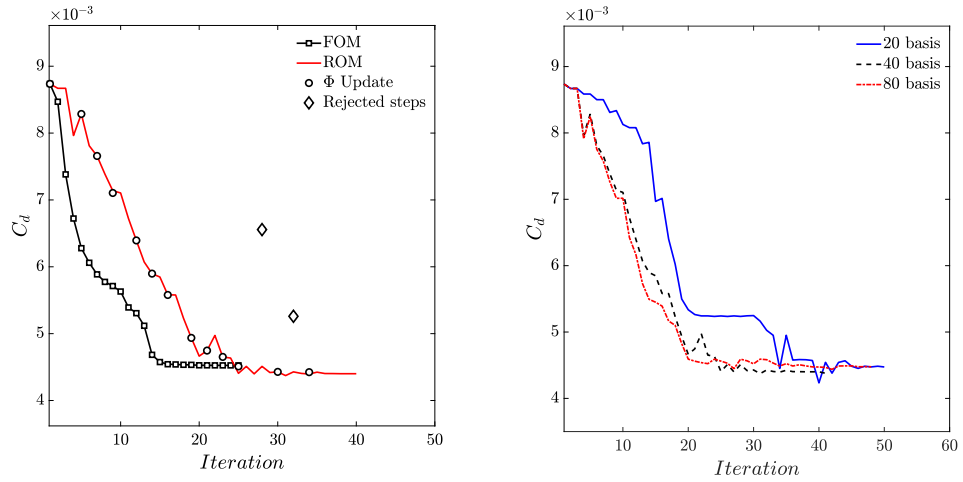


Figure 10: (a) ROM and FOM based optimisation convergence history; (b) impact of number of ROBs on optimisation convergence.

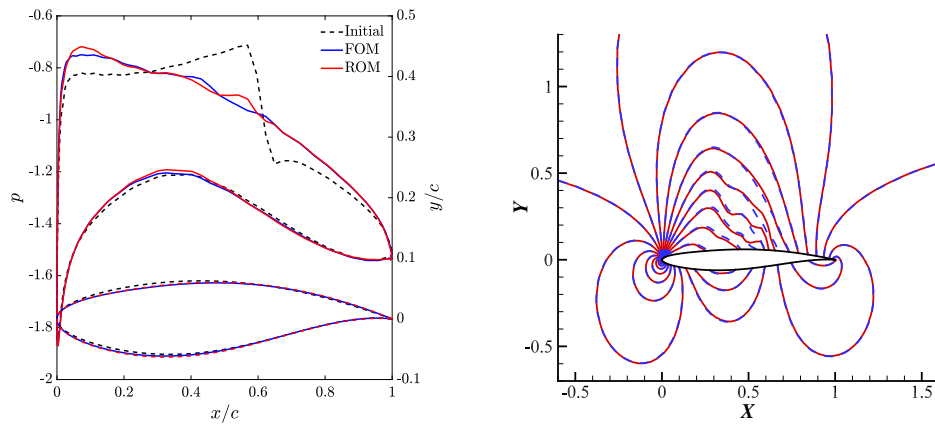


Figure 11: Pressure field comparison between FOM and ROM ($\varepsilon = 10^{-8}$). The ROM result is obtained by FOM steady state solution with the aerofoil shape from the ROM optimization. The solid and dashed lines represent ROM and FOM, respectively.

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%	

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

406 **Acknowledgments**

407 The funding provided for this research from the Engineering and Physical
408 Sciences Research Council (Grant No. EP/P025692/1) is gratefully acknowl-
409 edged.

- 410 [1] Agarwal, A., Biegler, L.T., 2013. A trust-region framework for con-
411 strained optimization using reduced order modeling. *Optimization and*
412 *Engineering* 14, 3–35. doi:10.1007/s11081-011-9164-0.
- 413 [2] Agarwal, D., Robinson, T.T., Armstrong, C.G., Marques, S., Vasilopou-
414 los, I., Meyer, M., 2018. Parametric design velocity computation for
415 cad-based design optimization using adjoint methods. *Engineering with*
416 *Computers* 34, 225–239. doi:10.1007/s00366-017-0534-x.
- 417 [3] Alexandrov, N.M., Lewis, R.M., 2003. First-order approximation and
418 model management in optimization, in: Biegler, L.T., Heinkenschloss,
419 M., Ghattas, O., van Bloemen Waanders, B. (Eds.), *Large-Scale PDE-*

- 420 Constrained Optimization, Springer Berlin Heidelberg, Berlin, Heidel-
421 berg. pp. 63–79.
- 422 [4] Ammar, A., Mokdad, B., Chinesta, F., Keunings, R., 2006. A new
423 family of solvers for some classes of multidimensional partial differen-
424 tial equations encountered in kinetic theory modeling of complex fluids.
425 Journal of Non-Newtonian Fluid Mechanics 139, 153 – 176.
- 426 [5] Amsallem, D., Farhat, C., Haasdonk, B., . Special issue on model re-
427 duction. International Journal for Numerical Methods in Engineering
428 102, 931–932. doi:10.1002/nme.4889.
- 429 [6] Benner, P., Gugercin, S., Willcox, K., 2015. A survey of projection-based
430 model reduction methods for parametric dynamical systems. SIAM Re-
431 view 57, 483–531.
- 432 [7] Bergmann, M., Cordier, L., 2008. Optimal control of the cylinder
433 wake in the laminar regime by trust-region methods and pod reduced-
434 order models. Journal of Computational Physics 227, 7813 – 7840.
435 doi:10.1016/j.jcp.2008.04.034.
- 436 [8] Bui-Thanh, T., Willcox, K., Ghattas, O., 2008. Model reduction for
437 large-scale systems with high-dimensional parametric input. SIAM Jour-
438 nal on Scientific Computing 30, 3270–3288. doi:10.1137/070694855.
- 439 [9] Carter, R.G., 1991. On the global convergence of trust region algo-
440 rithms using inexact gradient information. SIAM Journal on Numerical
441 Analysis 28, 251–265.

- 442 [10] Conn, A., Gould, N., Toint, P., 2000. Trust Region Methods. MPS-
443 SIAM Series on Optimization, Society for Industrial and Applied Math-
444 ematics.
- 445 [11] Conn, A.R., Gould, N., Sartenaer, A., Toint, P.L., 1993. Global conver-
446 gence of a class of trust region algorithms for optimization using inexact
447 projections on convex constraints. *SIAM Journal on Optimization* 3,
448 164–221.
- 449 [12] Elliott, J., Peraire, J., 1997. Practical three-dimensional aerodynamic
450 design and optimization using unstructured meshes. *AIAA journal* 35,
451 1479–1485.
- 452 [13] Giles, M.B., Pierce, N.A., 2000. An introduction to the adjoint approach
453 to design. *Flow, Turbulence and Combustion* 65, 393–415.
- 454 [14] Hascoët, L., Pascual, V., 2013. The Tapenade Automatic Differentia-
455 tion tool: Principles, Model, and Specification. *ACM Transactions On*
456 *Mathematical Software* 39.
- 457 [15] Iuliano, E., Quagliarella, D., 2013. Proper orthogonal decomposition,
458 surrogate modelling and evolutionary optimization in aerodynamic de-
459 sign. *Computers & Fluids* 84, 327–350.
- 460 [16] Jameson, A., 1988. Aerodynamic design via control theory. *Journal of*
461 *scientific computing* 3, 233–260.
- 462 [17] Jameson, A., Martinelli, L., Pierce, N., 1998. Optimum aerodynamic
463 design using the navier–stokes equations. *Theoretical and computational*
464 *fluid dynamics* 10, 213–237.

- 465 [18] Kevin, C., Charbel, B., Charbel, F., 2011. Efficient non-linear model
466 reduction via a least-squares petrov-galerkin projection and compressive
467 tensor approximations. *International Journal for Numerical Methods in*
468 *Engineering* 86, 155–181.
- 469 [19] Kulfan, B.M., 2008. Universal parametric geometry representation
470 method. *Journal of Aircraft* 45, 142 – 158.
- 471 [20] Lassila, T., Manzoni, A., Quarteroni, A., Rozza, G., 2014. Model order
472 reduction in fluid dynamics: challenges and perspectives, in: *Reduced*
473 *Order Methods for modeling and computational reduction*. Springer, pp.
474 235–273.
- 475 [21] LeGresley, P., Alonso, J., 2000. Airfoil design optimization using reduced
476 order models based on proper orthogonal decomposition, in: *Fluids 2000*
477 *conference and exhibit*, p. 2545.
- 478 [22] Lucia, D.J., Beran, P.S., Silva, W.A., 2004. Reduced-order modeling:
479 new approaches for computational physics. *Progress in Aerospace Sci-*
480 *ences* 40, 51 – 117. doi:10.1016/j.paerosci.2003.12.001.
- 481 [23] Manzoni, A., Quarteroni, A., Rozza, G., 2012. Shape optimization for
482 viscous flows by reduced basis methods and free–form deformation. *In-*
483 *ternational Journal for Numerical Methods in Fluids* 70, 646–670.
- 484 [24] Martins, J.R.R.A., Hwang, J.T., 2013. Review and unification of meth-
485 ods for computing derivatives of multidisciplinary computational mod-
486 els. *AIAA Journal* 51, 2582–2599.

- 487 [25] MATLAB, 2018. Matlab, version 2018b. The MathWorks, Natick, MA,
488 USA.
- 489 [26] MATLAB Optimization Toolbox, 2018. Matlab, version 2018b. The
490 MathWorks, Natick, MA, USA.
- 491 [27] Negri, F., Manzoni, A., Rozza, G., 2015. Reduced basis approximation
492 of parametrized optimal flow control problems for the stokes equations.
493 Computers and Mathematics with Applications 69, 319 – 336.
- 494 [28] Nielsen, E.J., Anderson, W.K., 1999. Aerodynamic design optimization
495 on unstructured meshes using the navier-stokes equations. AIAA journal
496 37, 1411–1419.
- 497 [29] Noack, B.R., Morzynski, M., Tadmor, G., 2011. Reduced-order mod-
498 elling for flow control. volume 528. Springer Science & Business Media.
- 499 [30] Pironneau, O., 1984. Optimal Shape Design for Elliptic Systems.
500 Springer series in computational physics, Springer-Verlag. URL:
501 <https://books.google.co.uk/books?id=6JqyAAAAIAAJ>.
- 502 [31] Qian, E., Grepl, M., Veroy, K., Willcox, K., 2017. A certified trust
503 region reduced basis approach to pde-constrained optimization. SIAM
504 Journal on Scientific Computing 39, S434–S460.
- 505 [32] Rowley, C.W., 2005. Model Reduction for Fluids, using Balanced Proper
506 Orthogonal Decomposition. International Journal of Bifurcation and
507 Chaos 15, 997–1013.

- 508 [33] Sirovich, L., 1987. Turbulence and the dynamics of coherent structures.
509 i: Coherent structures. ii: Symmetries and transformations. iii: Dynam-
510 ics and scaling. *Quarterly of Applied Mathematics* 45, 583–590.
- 511 [34] Skinner, S., Zare-Behtash, H., 2018. State-of-the-art in aerodynamic
512 shape optimisation methods. *Applied Soft Computing* 62, 933 – 962.
- 513 [35] Willcox, K., Peraire, J., 2002. Balanced model reduction via the proper
514 orthogonal decomposition. *AIAA journal* 40, 2323–2330.
- 515 [36] Yao, W., Marques, S., 2015. Prediction of transonic limit-cycle oscilla-
516 tions using an aeroelastic harmonic balance method. *AIAA Journal* 53,
517 2040–2051. doi:10.2514/1.J053565.
- 518 [37] Yondo, R., Andrés, E., Valero, E., 2018. A review on design of ex-
519 periments and surrogate models in aircraft real-time and many-query
520 aerodynamic analyses. *Progress in Aerospace Sciences* 96, 23–61.
- 521 [38] Yue, Y., Meerbergen, K., 2013. Accelerating optimization of parametric
522 linear systems by model order reduction. *SIAM Journal on Optimization*
523 23, 1344–1370.
- 524 [39] Zahr, M., Charbel, F., 2015. Progressive construction of a parametric
525 reduced–order model for PDE–constrained optimization. *International*
526 *Journal for Numerical Methods in Engineering* 102, 1111–1135.