Construction of Parametrically-Robust CFD-Based Reduced-Order Models for PDE-Constrained Optimization

Matthew J. Zahr *, David Amsallem †, and Charbel Farhat‡
Stanford University, Stanford, CA 94305, U.S.A.

A method for simultaneously constructing a reduced-order model and using it as a surrogate model to solve a PDE-constrained optimization problem is introduced. A reduced-order model is built for the parameters corresponding to the initial guess of the optimization problem. Since the resulting reduced-order model can be expected to be accurate only in the vicinity of this point in the parameter space, emphasis is placed on constructing this model by searching for regions of high error. These are determined by solving a small, nonlinear program with the objective defined as a linear combination of a residual error indicator and the objective function of the original PDE-constrained optimization problem. The reduced-order model is updated with information from the high-dimensional model in the regions of large error, and the process is repeated with more emphasis placed on solving the PDE-constrained optimization problem. The iteration terminates when the optimality conditions of the surrogate PDE-constrained optimization problem are satisfied. Application to a standard, nonlinear CFD shape optimization problem shows that the proposed method effectively solves a PDE-constrained optimization problem with few full CFD simulation queries.

I. Introduction

Optimization problems constrained by nonlinear Partial Differential Equations (PDE) arise in many engineering fields and contexts including inverse modeling, control, and shape optimization. Among the most computationally expensive PDE-constrained optimization problems is shape optimization constrained by fluid flow equations, such as the Euler or Navier-Stokes equations. The solution of such an optimization problem is found iteratively after a computational fluid dynamics (CFD) method has been employed to discretize the flow equations. The high-dimensional nature of CFD simulations makes aerodynamic shape optimization a very large-scale nonlinear program with constraints defined by the discretized flow equations, which may correspond to tens of millions of nonlinear equations for practical simulations. The result is a large-scale non-convex, nonlinear program that is, in general, very difficult and computationally prohibitive to solve.

Despite the high-dimensional nature of many CFD models, the solution trajectory is typically confined to a low-dimensional affine subspace. This low-dimensional feature of CFD models can be leveraged via Model Order Reduction (MOR). MOR relies on the construction of a Reduced-Order Model (ROM) which has many fewer degrees of freedom than the original High-Dimensional Model (HDM), typically several orders

*Graduate Student, Institute for Computational and Mathematical Engineering, William F. Durand Building, Room 028A, Stanford University, Stanford, CA 94035-3035, USA. E-mail: mzahr@stanford.edu. AIAA Student Member.

†Engineering Research Associate, Department of Aeronautics and Astronautics, William F. Durand Building, Room 028A, Stanford University, Stanford, CA 94035-3035, USA. E-mail: amsallem@stanford.edu. AIAA Member.

‡Vivian Church Hoff Professor of Aircraft Structures, Department of Aeronautics and Astronautics, William F. Durand Building, Room 257, Stanford University, Stanford, CA 94305-3035; AIAA Fellow.

of magnitude fewer, without significant loss of fidelity [1–9]. The solution of a ROM is constrained to be a linear combination of columns of a pre-computed Reduced-Order Basis (ROB).

In this paper, nonlinear, projection-based MOR methods are considered. For problems with polynomial nonlinearity *in the state vector and parameter*, all quantities scaling with the dimension of the HDM can be pre-computed, resulting in a small, fast ROM [10]. However, in the general case, the implementation of MOR methods requires at least one operation per nonlinear iteration involving quantities that have dimensions of the original HDM [11]. To address this issue, various *hyperreduction* techniques have been proposed as an additional level of approximation built on the ROM [5–8, 11–16]. The purpose of hyperreduction is to eliminate operations that scale with the HDM via pre-computations, enabling the hyperreduced ROM to realize significant speedups. Extending the proposed method to incorporate hyperreduction is the subject of ongoing work.

To address the computational cost of shape optimization, the HDM constraint is replaced with the ROM. The result is a search for an optimal shape by considering flows that lie in the subspace defined by the ROB, which can be formulated as a shape optimization problem with the shape parameters and the reduced coordinates of the flow solution as the optimization variables. This significantly decreases the number of optimization variables and nonlinear constraints. However, in general, the flow around the optimal shape will not lie in the span of the ROB. Thus, the solution of the ROM-constrained nonlinear program is an approximation to the optimal shape.

Traditionally, model order reduction has been viewed in an offline-online framework. In the offline phase, generally considered non-time-critical, the expensive operations involved in constructing a ROM are performed. The online phase, consisting of all time-critical computations, involves exploiting the ROM, possibly many times. This offline-online breakdown is most appropriate for real-time analysis situations, where the goal is to construct a ROM that can deliver predictions as fast and accurately as possible on a "deployable" device, such as a smartphone or tablet [17,18]. In the context of optimization, the offline-online breakdown is not appropriate since the only quantity of interest is the elapsed time from the beginning of the design process to the time the optimal solution is obtained. This suggests that ROM construction is time-critical in the optimization context. Furthermore, the task of training a ROM such that it maintains accuracy in the entire parameter space is unrealistic for problems with a large-dimensional parameter space.

Experience with ROMs has brought to light their lack of robustness with respect to parameter variations [19–23]. This is particularly crippling for the use of ROMs in optimization since all optimization methods amount to a search in parameter space. An approach to progressively build a reduced-order model during the search for an optimal solution was introduced in a trust region framework in [24, 25], the TRPOD method. TRPOD uses a trust-region framework with non-quadratic model problem defined by the PDE-constrained optimization objective function with the reduced-order model as a surrogate for the original discretized PDE to yield a convergent optimization procedure. In this work, the parametric non-robustness of reduced-order models is addressed by defining a procedure for simultaneously constructing a reduced-order model and solving the PDE-constrained optimization problem of interest. Early iterations use a residual indicator to emphasize ROM construction, while later iterations shift focus to the objective function of the PDE-constrained optimization problem of interest.

II. Model Reduction

II.A. Nonlinear High-Dimensional Model

Consider a set of nonlinear, time-dependent Partial Differential Equations (PDE) that have been spatially discretized to yield the following system of Ordinary Differential Equations (ODE):

$$\begin{aligned} \frac{d\mathbf{w}(t)}{dt} &= \mathbf{f}(\mathbf{w}(t), t, \boldsymbol{\mu}) \\ \mathbf{w}(0) &= \mathbf{w}_0, \end{aligned} \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^N$ is the state vector, $t \in \mathbb{R}_+$ represents time, $\boldsymbol{\mu} \in \mathcal{D} \subseteq \mathbb{R}^p$ is a vector of parameters, and $\mathbf{f} : \mathbb{R}^N \times \mathbb{R}_+ \times \mathcal{D} \to \mathbb{R}^N$ is a nonlinear, vector-valued function defining the nonlinearity of the discretized

PDE. As mentioned in Section I, the techniques developed in this paper are foundational steps toward ROM-based shape optimization. Accordingly, (1) represents a discretization of the Navier-Stokes or Euler equations around an aircraft system or subsystem and the vector of parameters, $\boldsymbol{\mu} \in \mathcal{D}$, defines the shape of the aircraft.

In this document, model reduction is applied at the fully discrete level [6,8]. Implicit temporal discretization of (1) yields the following sequence of nonlinear systems of equations:

$$\mathbf{r}(\mathbf{w}^{(i)}, t_i, \boldsymbol{\mu}) = 0 \qquad i = 1, 2, \ldots, N_t, \tag{2}$$

where $\mathbf{w}^{(i)} = \mathbf{w}(t_i)$. For brevity, define $\mathbf{r}^{(i)}(\mathbf{w}^{(i)}, \boldsymbol{\mu}) = \mathbf{r}(\mathbf{w}^{(i)}, t_i, \boldsymbol{\mu})$. Nonlinear systems of equations such as (2) are solved using iterative methods, such as the Newton-Raphson method. It will be necessary, at times, to distinguish between nonlinear iterates; define $\mathbf{w}^{(i,l)}$ as the l-th nonlinear iteration arising during the search for $\mathbf{w}(t_i)$. Using this notation and assuming convergence of the nonlinear iteration, $\lim_{l\to\infty} \mathbf{w}^{(i,l)} = \mathbf{w}^{(i)}$. Similarly, define $\mathbf{r}^{(i,l)} = \mathbf{r}^{(i)}(\mathbf{w}^{(i,l)}, \boldsymbol{\mu})$. From the terminology established in Section I, (2) is the fully-discrete governing equation of the HDM.

II.B. Model Order Reduction

The central assumption of model reduction is that the state vector belongs to a low-dimensional affine subspace:

$$\mathbf{w}^{(i)} = \mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i)}, \tag{3}$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{N\times k^{\mathbf{w}}}$ is the reduced-order basis, and $\mathbf{w}_r^{(i)} \in \mathbb{R}^{k^{\mathbf{w}}}$ are the reduced coordinates of $\mathbf{w}^{(i)}$. In general, $k^{\mathbf{w}} \ll N$.

By substituting the MOR assumption (3) into the HDM (2), we have the underdetermined nonlinear system of equations:

$$\mathbf{r}^{(i)}(\mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i)}, \boldsymbol{\mu}) = 0. \tag{4}$$

Since (4) does not, in general, have a solution, additional constraints are introduced by requiring that the nonlinear residual be orthogonal to the columns of $\boldsymbol{\Psi} \in \mathbb{R}^{N\times k^{\mathbf{w}}}$, the left basis. Enforcing this orthogonality constraint leads to the (square) nonlinear system of equations:

$$\boldsymbol{\Psi}^T\mathbf{r}^{(i)}(\mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i)}, \boldsymbol{\mu}) = 0. \tag{5}$$

The simplest option for the left basis is $\boldsymbol{\Psi} = \boldsymbol{\Phi}$ leading to a Galerkin projection of the form

$$\boldsymbol{\Phi}^T\mathbf{r}^{(i)}(\mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i)}, \boldsymbol{\mu}) = 0. \tag{6}$$

It can be shown that this choice of left basis leads to an "optimal" Newton search direction for problems with symmetric, positive-definite (SPD) Jacobians [6].

Another option for the left basis is $\boldsymbol{\Psi} = \frac{\partial \mathbf{r}}{\partial \mathbf{w}}(\mathbf{w}, t, \boldsymbol{\mu})\,\boldsymbol{\Phi}$ leading to the Least-Squares Petrov-Galerkin (LSPG) [6,8,26] projection of the form

$$\boldsymbol{\Phi}^T\mathbf{J}^{(i,l)T}\mathbf{r}^{(i)}(\mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i,l)}, \boldsymbol{\mu}) = 0. \tag{7}$$

where $\mathbf{J}^{(i,l)} = \frac{\partial \mathbf{r}}{\partial \mathbf{w}}\left(\mathbf{w}^{(0)} + \boldsymbol{\Phi}\mathbf{w}_r^{(i,l)}, t_i, \boldsymbol{\mu}\right)$ and $\mathbf{w}_r^{(i,l)}$ is the l-th nonlinear iteration arising in the search for $\mathbf{w}_r^{(i)}$. It can be shown that this choice of left basis leads to an "optimal" Newton search direction for problems with non-SPD Jacobians [6,26].

II.C. Reduced-Order Basis Construction

In this section, a Proper Orthogonal Decomposition (POD) [27] algorithm based on the method of snapshots for constructing ROMs is introduced. POD is a popular method for data compression that is often used in model reduction to compute the reduced basis $\boldsymbol{\Phi}$. Standard techniques for building a reduced basis consist

Algorithm 1 ROM Construction

Input: ROB size, $k^{\mathbf{w}}$
Output: ROB, $\mathbf{\Phi}$

1: Select k training points: $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k\} \subset \mathcal{D}$
2: **for** $j = 1, 2, \ldots, k$ **do**
3: For each $i \in \{1, 2, \ldots, N_t\}$, solve $\mathbf{r}^{(i)}(\mathbf{w}, \boldsymbol{\mu}_j) = 0 \rightarrow \hat{\mathbf{w}}_j^i$ (Collection)
4: Define $\mathbf{X}_j = \begin{bmatrix} \hat{\mathbf{w}}_j^1 & \hat{\mathbf{w}}_j^2 & \cdots & \hat{\mathbf{w}}_j^{N_t} \end{bmatrix} \in \mathbb{R}^{N \times N_t}$
5: **end for**
6: $\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_k \end{bmatrix} \in \mathbb{R}^{N \times k \cdot N_t}$
7: Compute the thin SVD of $\mathbf{X}$: $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{Z}^T$, where $\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_{k \cdot N_t} \end{bmatrix}$
8: $\mathbf{\Phi} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_{k^{\mathbf{w}}} \end{bmatrix}$ (Compression)

of two main steps: 1) snapshot collection and 2) snapshot compression. The collection phase consists of selecting a set of parameters $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\} \subset \mathcal{D}$, solving the HDM at each of these configurations, and saving state vector snapshots. This procedure is summarized in Algorithm 1.

The training points used to construct the ROM determine its robustness with respect to parameter variations. It is important to note that every training point selected requires a solution of the HDM. Accordingly, the goal of ROB construction is to chose k parameters $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\}$ with which to train the ROM such that k is small (i.e. $k = \mathcal{O}(10)$) and the resulting ROM is accurate in as much of the parameter space as possible.

III. PDE-Constrained Optimization

With the ROM framework established, the focus is returned to the central goal of using ROMs as surrogates in PDE-constrained optimization. For the remainder of this document, the scope will be narrowed to the case of a steady PDE. With this simplification, the $^{(i)}$ superscripts can be dropped without introducing ambiguity. Consider the following shape optimization problem:

$$\begin{aligned} \underset{\boldsymbol{\mu} \in \mathbb{R}^p}{\text{minimize}} \quad & f\left(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}\right) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \\ & \mathbf{c}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0 \end{aligned} \tag{8}$$

where the objective function $f : \mathbb{R}^N \times \mathcal{D} \rightarrow \mathbb{R}$ may represent drag, negative lift, weight, or any other feature to be optimized. The equality and inequality constraints, $\mathbf{g} : \mathbb{R}^N \times \mathcal{D} \rightarrow \mathbb{R}^{n_e}$ and $\mathbf{c} : \mathbb{R}^N \times \mathcal{D} \rightarrow \mathbb{R}^{n_i}$, could represent constraints on admissible shapes, requirements on flow variables or integrated quantities, such as subsonic flow or positive lift. In (8), the state vector $\mathbf{w}$ is implicitly defined as a function of the shape parameters $\boldsymbol{\mu}$ via the HDM equation $\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = 0$ and the Implicit Function Theorem, under standard assumptions. This leads to the Nested Analysis and Design (NAND) framework for PDE-constrained optimization.

The global solution of (8) will yield the parameter $\boldsymbol{\mu}^*$, the location in parameter space where the objective function achieves a minimum value over all feasible flows, $\mathbf{w}$, and admissible shapes, $\boldsymbol{\mu}$. Any nonlinear programming solver can be applied to (8) to find a local optimum, such as penalty methods, sequential quadratic programming (SQP), or interior-point methods [28].

III.A. ROM-Constrained Optimization

Given the iterative nature of optimization problems, a (local) solution of (8) will be computationally expensive since every operation involves high-dimensional quantities, i.e. scales with N. In an attempt to rapidly find

a good approximation to the solution of (8), the high-dimensional constraint is replaced with the ROM:

$$\begin{aligned} \underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f\left(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}\right) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \\ & \mathbf{c}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0, \end{aligned} \tag{9}$$

where $\mathbf{w}_r$ is implicitly defined as a function of $\boldsymbol{\mu}$ through the equation $\boldsymbol{\Psi}^T\mathbf{r}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r, \boldsymbol{\mu}) = 0$ and the Implicit Function Theorem, under usual assumptions. The size of the nonlinear program that must be solved is unchanged, but the dimension of the implicit function defining the state vector as a function of the parameter has been significantly reduced using a ROM as a surrogate. As previously mentioned, for significant speed-ups to be realized, the ROM needs to be replaced by a hyperreduced ROM [5–8, 11–16].

III.B. Simultaneous ROM Construction and ROM-Constrained Optimization

As alluded to in Section I, the offline-online framework for reduced-order modeling is not natural in the optimization context because there is no distinction between time-critical and non-time-critical phases. The only important considerations are the total computing time and resources required to determine the optimal shape. Accordingly, this section presents a strategy for progressively constructing a reduced-order model during the search for the optimal solution of the ROM-constrained optimization problem.

Consider the real-valued function $\mathcal{L}(\mathbf{w}_r, \boldsymbol{\mu}) = \frac{1}{2}||\mathbf{r}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r, \boldsymbol{\mu})||_2^2$. If $\mathbf{w}_r$ satisfies $\boldsymbol{\Psi}^T\mathbf{r}\left(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r, \boldsymbol{\mu}\right) = 0$, this function is the norm of residual of the HDM evaluated at the solution of the ROM. Since residual norms can be used as error indicators for well-posed problems, $\mathcal{L}(\mathbf{w}_r, \boldsymbol{\mu})$ provides some sense of the accuracy of the ROM at the cost of solving the governing equation of the ROM, without requiring the solution of the HDM. A linear combination of the residual error indicator $\mathcal{L}$ and the objective function of the problem of interest (8), f, is used to define the objective function of a new problem:

$$\begin{aligned} \underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) - \gamma\mathcal{L}(\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \\ & \mathbf{c}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0, \end{aligned} \tag{10}$$

where $\boldsymbol{\Psi}^T\mathbf{r}\left(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r, \boldsymbol{\mu}\right) = 0$ and γ is a positive constant.

If $\gamma \to \infty$, the solution of (10) is the point in the parameter space where the error indicator is largest. If $\mathcal{L}(\mathbf{w}_r(\boldsymbol{\mu}^*), \boldsymbol{\mu}^*) > \epsilon$ for some fixed $\epsilon > 0$, the ROM needs additional training in the neighborhood about $\boldsymbol{\mu}^*$. When $\gamma = 0$, the solution of (10) coincides the solution of the ROM-constrained optimization problem (9). The proposed strategy for simultaneously constructing a ROB and solving the ROM-constrained optimization starts from an initial, given basis $\boldsymbol{\Phi}$, built from a very coarse sampling of the parameter space, possibly only the initial point for the optimization algorithm. The resulting ROM will lack robustness in a large portion of the parameter space due to its sparse training. Accordingly, early in the algorithm, emphasis is given to maximizing $\mathcal{L}$ by setting γ to some positive value. Once the solution of (10) is found, the HDM is simulated for the shape defined by this solution and the ROB $\boldsymbol{\Phi}$ is updated with the new snapshots. Every HDM sample and subsequent basis update improves the parametric robustness of the ROM. Accordingly, as the algorithm progresses, weight is shifted from maximizing $\mathcal{L}$ to minimizing f. Transition between maximization of $\mathcal{L}$ and minimization of f is performed progressively by treating γ as a penalty-type parameter by progressively dropping to it zero. The optimality conditions of (9) are used to terminate this process.

Remark. For many CFD applications, the steady fluid flow equations are solved by pseudo-time-stepping [29, 30]; see Section IV for additional details. Accordingly, the HDM solve in Line 5 of Algorithm 2 is a pseudo-unsteady solve and will generate snapshot information at each "time" step; similar to the procedure outlined in Algorithm 1.

Algorithm 2 Simultaneous ROM construction and ROM-constrained optimization

Input: Initial ROB, $\boldsymbol{\Phi}$; initial value of weight factor, $\gamma_0 \in \mathbb{R}_+$; amplification factor, $0 < \rho < 1$
Output: Solution to (9), $\boldsymbol{\mu}^*$ (approximation to solution of (8))

$\gamma = \gamma_0$
while optimality conditions of (9) not satisfied **do**
  Solve (10) $\rightarrow \boldsymbol{\mu}^*$ and $\mathbf{w}_r^* = \mathbf{w}_r(\boldsymbol{\mu}^*)$
  if $||\mathbf{r}(\mathbf{w}_0 + \boldsymbol{\Phi}\mathbf{w}_r^*, \boldsymbol{\mu}^*)|| > \epsilon$ **then**
    Solve $\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}^*) = 0 \rightarrow \mathbf{w}^*$
    Update $\boldsymbol{\Phi}$ with new snapshots $\mathbf{w}^*$
  end if
  $\gamma = \rho\gamma$
end while

IV. Application: Nozzle Shape Optimization

In this section, the proposed method in Algorithm 2 is applied to shape optimization of a 1D nozzle problem with flow governed by the Euler equations. The Quasi-1D Euler equations are

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{1}{A}\frac{\partial (A\mathbf{F})}{\partial x} = \mathbf{Q} \qquad x \in [0, L], \tag{11}$$

where

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \qquad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e+p)u \end{bmatrix}, \qquad \mathbf{Q} = \begin{bmatrix} 0 \\ \frac{p}{A}\frac{\partial A}{\partial x} \\ 0 \end{bmatrix}.$$

The finite volume method with Roe Fluxes and entropy corrections are used to semi-discretize (11). Temporal dependence is included in (11) for the purposes of pseudo-time-stepping to a steady state since it is well-known that applying Newton's method to solve the spatial-only part of (11) will be plagued by local minima.

For the numerical experiment, the spatial discretization uses 100 volumes, resulting in 300 degrees of freedom in the HDM. Pseudo-time-stepping is performed using Backward Euler with a monotonically increasing CFL law to quickly converge to a steady-state. All ROMs in the numerical experiment have dimension $k^{\mathbf{w}} = 25$.

The nozzle area, $A(x)$, is parametrized using cubic splines with 13 control points. The position of these 13 spline control points are the parameters, $\boldsymbol{\mu}$, for the optimization problem to follow. The shape optimization problem under consideration is parameter estimation, whereby the optimal parameter $\boldsymbol{\mu}^*$ is chosen a-priori (uniquely defining $A^*(x)$) and the solution of the discrete form of (11) is computed, $\mathbf{w}^*$. Then, the parameter estimation problem is

$$\begin{aligned} \underset{\boldsymbol{\mu} \in \mathbb{R}^p}{\text{minimize}} \quad & \frac{1}{2}||\mathbf{w}(\boldsymbol{\mu}) - \mathbf{w}^*||_2^2 \\ \text{subject to} \quad & \mathbf{c}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0, \end{aligned} \tag{12}$$

where $\mathbf{c}$ represents constraints on admissible nozzle shapes and/or flows and $\mathbf{w}$ is an implicit function of $\boldsymbol{\mu}$ through equation (11). The constraints used in (12) are

$$\mathbf{c}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = \begin{bmatrix} -A''(x_i) \\ A(x_i) - A_u(x_i) \\ A_l(x_i) - A(x_i) \\ A'(0) \\ -A'(L) \end{bmatrix} \qquad \text{for each } i.$$

The constraints enforce practical requirements on the nozzle shape: convexity of the nozzle at every point in the domain, area bounds on the nozzle at every point in the domain, and negative inlet and positive

outlet slope. Figure 1 contains the relevant nozzle shapes defining (12): the initial guess for the optimization algorithm ($A_0(x)$), the optimal solution ($A^*(x)$), the area bounds ($A_l(x)$, $A_u(x)$ for the lower and upper bounds, respectively), and the location of the spline control points. Figure 2 contains the Mach distribution resulting from solving the HDM at the initial guess and optimal solution.

Numerical Experiment

Three methods are applied to solve (12),

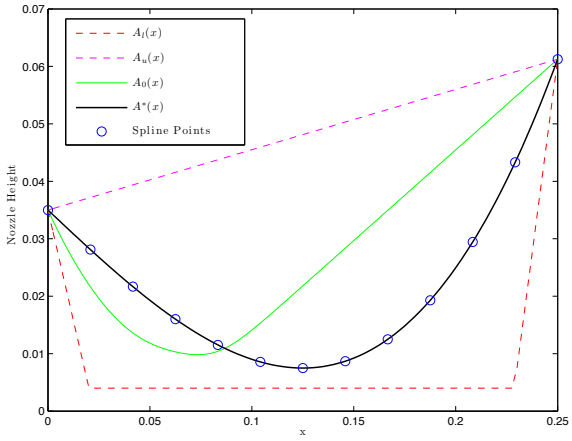
- HDMOpt, whereby (12) is solved with standard PDE-constrained optimization technology (NAND framework) without introducing a reduced-order model,
- LHSOpt, whereby a ROM is built using Algorithm 1 with parameter samples chosen based on a Latin Hypercube Sampling (LHS) of the feasible region; the HDM is replaced with the resulting ROM in (12),
- MergeOpt, whereby Algorithm 2 is applied to (12).

For HDMOpt and MergeOpt, the initial guess for the optimization algorithm is shown in Figure 1. For LHSOpt, the initial guess for optimization is the HDM sample that attains the lowest value of the objective function. SNOPT [31] is the nonlinear programming software used to generate all results in this document. Due to the randomness inherent in LHSOpt from the LHS used to define the HDM samples and build the ROM, LHSOpt is applied to (12) multiple times to gather statistics. Since hyperreduction was not incorporated in this study, the large computational cost involved in solving (12) using LHSOpt limited our sample size to 100.

Table 1 contains the results of applying the aforementioned methods to (12). HDMOpt achieves very small errors in both the parameter being estimated ($\boldsymbol{\mu}^*$) and state being matched ($\mathbf{w}^*$) at the expense of 202 HDM evaluations. The LHSOpt algorithm was applied to solve (12) using 12 nozzle samples to generate snapshots and construct a reduced-order model of dimension of 25 (Algorithm 1). LHSOpt was able to approximately solve (12) with only 12 queries to the HDM and 168 queries to the ROM (on average); however, the error in both $\boldsymbol{\mu}^*$ and $\mathbf{w}^*$ is, on average, unacceptably large. The nozzle samples chosen by the LHS in LHSOpt, as well as the corresponding Mach number distribution along the domain, for a trial that resulted in small errors, are shown in Figures 3 and 4, respectively. Finally, MergeOpt generated approximations to $\boldsymbol{\mu}^*$ and $\mathbf{w}^*$ with acceptable errors of about 3% and 1%, respectively; requiring 9 queries to the HDM and 770 queries to the ROM. Figures 5 and 6 show the nozzle samples chosen by MergeOpt and the corresponding Mach distribution, respectively.

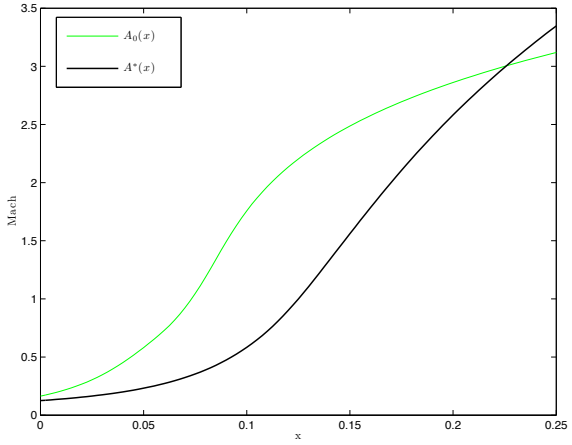
Recall, each nozzle sample requires a solution of the expensive, high-dimensional model to generate snapshots with which to update the reduced-order basis. As discussed in Section II.C, the goal when using ROMs as a surrogate in PDE-constrained optimization is to find an accurate approximation to the optimal solution while keeping the number of queries to the HDM low. Table 1 shows that MergeOpt was able to achieve this goal *at the cost of a large number of ROM simulations*. When hyperreduction is introduced in the next phase of this project, this will be an acceptable trade-off because optimized implementations of hyperreduction have yielded speedups of over 400 on large-scale CFD problems [6,7]. Comparing Figures 3 and 4 with Figures 5 and 6, it is clear that MergeOpt progressively takes samples closer to the optimal solution while LHSOpt samples the entire parameter space since it is restricted to the confines of the offline-online framework. Focusing the HDM samples near the optimal solution allows smaller, more accurate ROMs to be constructed *for the regions of interest*; the result is a better approximation of the optimal solution with fewer queries to the HDM.

Figure 1: Nozzle Configurations



Nozzle Configurations: Lower/Upper Bounds (A_l, A_u), Initial Guess (A_0), Optimal Solution (A^*)

Figure 2: Mach Distribution at Nozzle Configurations



Mach Distribution at Nozzle Configurations: Initial Guess (A_0) and Optimal Solution (A^*)

Table 1: Accuracy and performance comparison for various PDE-constrained optimization methods for the shape optimization problem

	MergeOpt	**LHSOpt (mean)**	**HDMOpt**
L2 Error in $\boldsymbol{\mu}^*$ (%)	2.88	16.4	1.04×10^{-7}
L2 Error in $\mathbf{w}^*$ (%)	0.67	12.3	6.79×10^{-9}
# HDM Simulations	9	12	202
# ROM Simulations	770	168	-

Figure 3: LHSOpt HDM Nozzle Configuration Samples

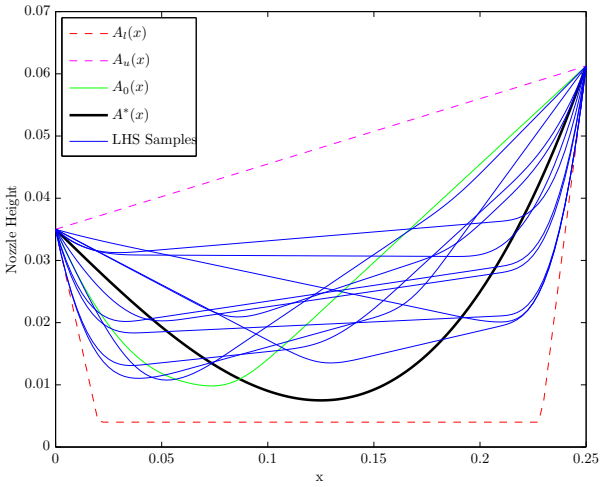


Figure 4: Mach Distribution at LHSOpt HDM Nozzle Configuration Samples

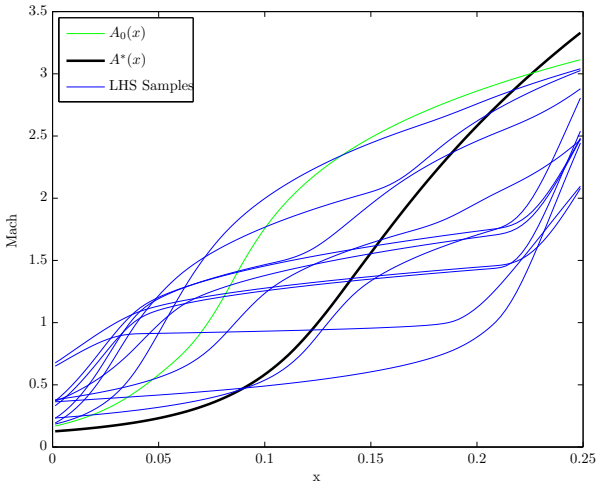


Figure 5: MergeOpt HDM Nozzle Configuration Samples

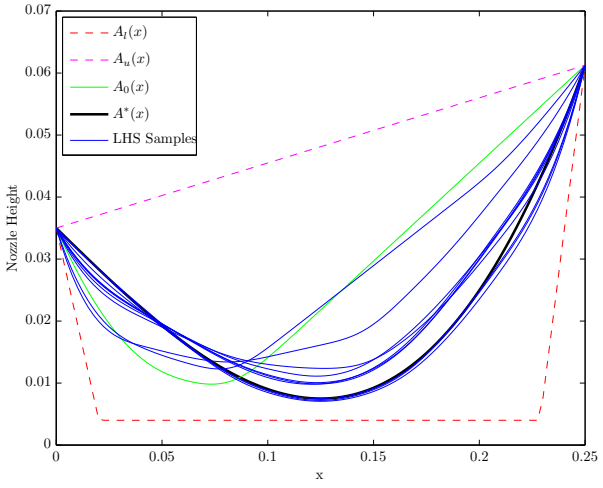
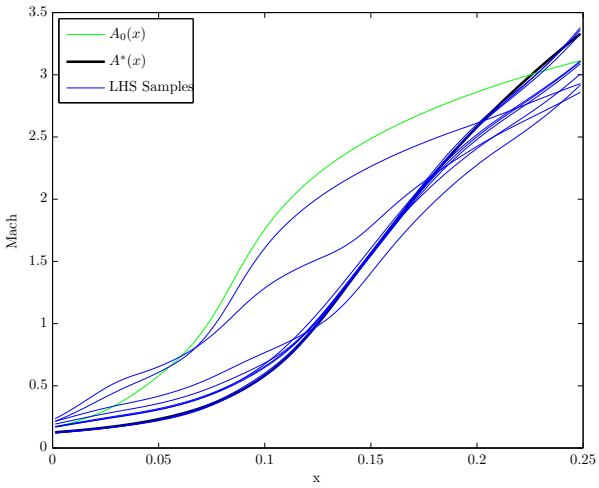


Figure 6: Mach Distribution at MergeOpt HDM Nozzle Configuration Samples



V. Conclusion

A technique for simultaneously constructing a reduced-order model and using it as a surrogate for PDE-constrained optimization is introduced. With optimization as the central focus, the offline-online framework of model order reduction is no longer appropriate. The reasons for this are two-fold: (1) the only quantity of interest in design optimization is the total wall time from the beginning of the design process to the time an optimal solution is obtained and (2) building an accurate, parametrically-robust reduced-order model for a large-scale CFD simulation with a high-dimensional parameter space is an unrealistic goal since the assumption that the state trajectory is confined to a low-dimensional subspace will likely breakdown . It was shown that breaking the offline-online structure allows a reduced-order model to be progressively specialized for regions of the parameter space near the optimal solution. The result is small, more accurate ROMs in the region of interest. The proposed algorithm was applied to the shape optimization of a rocket nozzle with flow governed by the Euler equations. The proposed algorithm was able to find a more accurate approximation to the optimal nozzle shape than a ROM constructed from a Latin Hypercube sampling of the feasible region.

Acknowledgements

The first author acknowledges the support of a Department of Energy Computational Science Graduate Fellowship. The authors also acknowledge partial support by the Army Research Laboratory through the Army High Performance Computing Research Center under Cooperative Agreement W911NF-07-2-0027, partial support by the Office of Naval Research under Grant N00014-11-1-0707, partial support by The Boeing Company under Contract Sponsor Ref. 45047, and partial support by a research grant from King Abdulaziz City for Science and Technology (KACST). The content of this publication does not necessarily reflect the position or policy of any of these supporters, and no official endorsement should be inferred.

The first author would also like to thank Professor Walter Murray of Stanford University for helpful and productive discussions on this topic.

References

[1]Rewienski, M. and White, J., "Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations." *Linear Algebra and its Applications*, Vol. 415, No. 2-3, 2006, pp. 426–454.

[2]Lieu, T., Farhat, C., and Lesoinne, M., "Reduced-order fluid/structure modeling of a complete aircraft configuration," *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, No. 41-43, August 2006, pp. 5730–5742.

[3]Bui-Thanh, T., Willcox, K., and Ghattas, O., "Parametric Reduced-Order Models for Probabilistic Analysis of Unsteady Aerodynamic Applications," *AIAA Journal*, Vol. 46, No. 10, 2008, pp. 2520–2529.

[4]Amsallem, D., Cortial, J., and Farhat, C., "Toward Real-Time Computational-Fluid-Dynamics-Based Aeroelastic Computations Using a Databaseof Reduced-Order Information." *AIAA Journal*, Vol. 48, No. 9, 2010, pp. 2029–2037.

[5]Chaturantabut, S. and Sorensen, D., "Nonlinear Model Reduction via Discrete Empirical Interpolation," *SIAM Journal on Scientific Computing*, Vol. 32, 2010, pp. 2737.

[6]Carlberg, K., Farhat, C., and Bou-Mosleh, C., "Efficient Nonlinear Model Reduction via a Least-Squares Petrov-Galerkin Projection and Compressive Tensor Approximations," *International Journal for Numerical Methods in Engineering*, Vol. 86, No. 2, 2011, pp. 155–181.

[7]Carlberg, K., Farhat, C., Cortial, J., and Amsallem, D., "The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows," *Journal of Computational Physics*, 2013.

[8]Carlberg, K., Cortial, J., Amsallem, D., Zahr, M., and Farhat, C., "The GNAT nonlinear model reduction method and its application to fluid dynamics problems," *6th AIAA Theoretical Fluid Mechanics Conference*, AIAA Paper, July 2011, p. 3112.

[9]Amsallem, D. and Farhat, C., "Stabilization of projection-based reduced-order models," *International Journal for Numerical Methods in Engineering*, 2012.

[10]Barbic, J. and Adviser-James, D. L., "Real-time reduced large-deformation models and distributed contact for computer graphics and haptics," 2007.

[11]Rathinam, M. and Petzold, L., "A new look at proper orthogonal decomposition," *SIAM Journal on Numerical Analysis*, 2004, pp. 1893–1925.

[12]Everson, R. and Sirovich, L., "Karhunen-Loeve procedure for gappy data," *Journal of the Optical Society of America*, Vol. 12, No. 8, 1995, pp. 1657–1664.

[13]Barrault, M., Maday, Y., Nguyen, N., and Patera, A., "An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations," *Computes Rendus de l'Academie des Sciences Paris*, Vol. 339, 2004, pp. 667–672.

[14]Ryckelynck, D., "A priori hyperreduction method: An adaptive approach," *Journal of Computational Physics*, Vol. 202, No. 1, 2005, pp. 346–366.

[15]Astrid, P., Weiland, S., and Willcox, K., "Missing point estimation in models described by proper orthogonal decomposition," *Automatic Control*, 2008.

[16]Willcox, K., "Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition," *Computers & Fluids*, Vol. 35, No. 208-226, 2006.

[17]Amsallem, D., Zahr, M., and Farhat, C., "On-Line Framework for Real-Time Linearized Parametric CFD Computations Based on Model Reduction," *Manuscript in preparation*.

[18]Amsallem, D., Deolalikar, S., Fazzel, G., and Farhat, C., "Model Predictive Control under Coupled Fluid-Structure Constraints a Database of Reduced-Order Models on a Tablet," *submitted to the AIAA Fluid Dynamics and Co-located Conferences and Exhibit, 24 - 27 June 2013*.

[19]Epureanu, B., "A parametric analysis of reduced order models of viscous flows in turbomachinery," *Journal of fluids and structures*, Vol. 17, 2003, pp. 971–982.

[20]Lieu, T. and Farhat, C., "Adaptation of aeroelastic reduced-order models and application to an F-16 configuration," *AIAA Journal*, Vol. 45, No. 6, 2007, pp. 1244–1257.

[21]Amsallem, D. and Farhat, C., "Interpolation method for adapting reduced-order models and application to aeroelasticity," *AIAA Journal*, Vol. 46, No. 7, 2008, pp. 1803–1813.

[22]Amsallem, D., Cortial, J., Carlberg, K., and Farhat, C., "A method for interpolating on manifolds structural dynamics reduced-order models," *International Journal for Numerical Methods in Engineering*, Vol. 80, No. 9, November 2009, pp. 1241–1258.

[23]Amsallem, D. and Farhat, C., "An Online Method for Interpolating Linear Parametric Reduced-Order Models," *SIAM Journal on Scientific Computing*, Vol. 33, No. 5, 2011, pp. 2169–2198.

[24]Arian, E., Fahl, M., and Sachs, E. W., "Trust-region proper orthogonal decomposition for flow control," Tech. rep., DTIC Document, 2000.

[25]Fahl, M., *Trust-region methods for flow control based on reduced order modelling*, Ph.D. thesis, Universitätsbibliothek, 2001.

[26]Bui-Thanh, T., Willcox, K., and Ghattas, O., "Model Reduction For Large-Scale Systems With High-Dimensional Parametric Input Space," *SIAM Journal on Scientific Computing*, Vol. 30, No. 6, 2008, pp. 3270–3288.

[27]Sirovich, L., "Turbulence and the dynamics of coherent structures. Part I: Coherent structures," *Quarterly of applied mathematics*, Vol. 45, No. 3, 1987, pp. 561–571.

[28]Nocedal, J. and Wright, S. J., *Numerical Optimization*, Springer, 2nd ed., 2006.

[29]Kelley, C. and Keyes, D. E., "Convergence analysis of pseudo-transient continuation," *SIAM Journal on Numerical Analysis*, Vol. 35, No. 2, 1998, pp. 508–523.

[30]Kelley, C., Liao, L.-Z., Qi, L., Chu, M. T., Reese, J., and Winton, C., "Projected pseudo-transient continuation," 2007.

[31]Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM journal on optimization*, Vol. 12, No. 4, 2002, pp. 979–1006.