MODEL REDUCTION FOR LARGE-SCALE SYSTEMS WITH HIGH-DIMENSIONAL PARAMETRIC INPUT SPACE

ACDL TECHNICAL REPORT TR-07-2

AUGUST 2007

T. BUI-THANH[*], K. WILLCOX[†], AND O. GHATTAS[‡]

Abstract. A model-constrained adaptive sampling methodology is proposed for reduction of large-scale systems with high-dimensional parametric input spaces. Our model reduction method uses a reduced basis approach, which requires the computation of high-fidelity solutions at a number of sample points throughout the parametric input space. A key challenge that must be addressed in the optimization, control, and probabilistic settings is the need for the reduced models to capture variation over this parametric input space, which, for many applications, will be of high dimension. We pose the task of determining appropriate sample points as a PDE-constrained optimization problem, which is implemented using an efficient adaptive algorithm that scales well to systems with a large number of parameters. The methodology is demonstrated for examples with parametric input spaces of dimension 11 and 21, which describe thermal analysis and design of a heat conduction fin, and compared with statistically-based sampling methods. For this example, the model-constrained adaptive sampling leads to reduced models that, for a given basis size, have error several orders of magnitude smaller than that obtained using the other methods.

Key words. model reduction, optimization, sampling, heat conduction

AMS subject classifications. 37N40, 37M99, 65K10

1. Introduction.

Model reduction is a systematic approach to derive cost-efficient representations of large-scale systems that result, for example, from discretization of partial differential equations (PDEs). Reduced models are essential in applications such as optimal design, optimal control, probabilistic analysis, and inverse problems, for which the physical system must be simulated repeatedly and thus computations with the large-scale model become prohibitive. To be useful for these applications, the reduced model must provide an accurate representation of the large-scale model over a wide range of parameters. When the dimension of the input parameter space is large, which is the case for many engineering applications of interest, sampling the parametric space efficiently to create the reduced model becomes a significant challenge.

Common model reduction approaches for dynamical systems include proper orthogonal decomposition (POD) and Krylov-based methods. In these methods, the reduced basis is formed as the span of a set of state solutions, commonly referred to as snapshots. These snapshots are computed by solving the full system for selected values of the parameters and selected forcing inputs (and possibly selected frequencies if a Krylov-subspace method is used). The quality of the resulting reduced-order model is very dependent on the choice of parameters and inputs over which snapshots are computed. This is because the span of the snapshot determines the span of the re-

[*] Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 (tanbui@mit.edu)

[†] Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 (kwillcox@mit.edu)

[‡] Department of Mechanical Engineering, Jackson School of Geosciences, and Institute for Computational Engineering & Sciences, University of Texas at Austin, Austin, TX 78712 (omar@ices.texas.edu)

duced basis, which in turn determines the quality of the resulting reduced-order model. A key issue that needs to be addressed therefore is sampling; that is, how to choose the parameters and inputs over which to compute the basis. In particular, discretization produces high-dimensional parametric input spaces when the parameters represent continuous fields (such as initial conditions, boundary conditions, distributed source terms, and geometry). Model reduction for high-dimensional parametric input spaces remains a challenging problem. One can use knowledge of the application at hand to determine representative parametric inputs, as has been done to sample the parameter space for the quasi-convex optimization relaxation method [29], and to generate a POD or Krylov basis for problems in which the number of input parameters is small, for example aerodynamic applications [13], Rayleigh-Bénard convection [21], parametrized design of interconnect circuits [5, 10], and parameters describing inhomogeneous boundary conditions for parabolic PDEs [17]. For optimal control applications, online adaptive sampling has been employed as a systematic way to generate snapshot information [1, 2, 18, 20]. However, these methods have not been scaled to problems that contain more than a handful of parameters. The recently developed iterative rational Krylov algorithm [16] proposes a systematic method for selecting interpolation points for multipoint rational Krylov approximations based on an $\mathcal{H}_2$-norm optimality criterion. This method has been applied to reduction of large-scale linear time invariant (LTI) systems, although its extension to parameter dependent LTI systems remains an open question.

Standard sampling schemes such as uniform sampling (uniform gridding of the parameter space) or random sampling are another option for creating snapshots. However, if the dimension of the parameter space is large, uniform sampling will quickly become too computationally expensive due to the combinatorial explosion of samples needed to cover the parameter space. Random sampling, on the other hand, might fail to recognize important regions in the parameter space. One sampling strategy that provides a compromise between the uniformity and the size of the sample is the stratified sampling family, of which the popular Latin hypercube sampling (LHS) method is one example [23]. The LHS method is more efficient than uniform sampling and often more accurate than random sampling. Recently, the centroidal voronoi tessellation (CVT) sampling method [11] has been shown to outperform LHS, random sampling Monte Carlo methods, and Hammersley quasi Monte Carlo sequence methods for statistical sampling and function integration [25].

To address the challenge of sampling a high-dimensional parameter space to build a reduced basis, the greedy sampling method was introduced in [14,15,30,31]. The key premise of greedy sampling is to adaptively choose samples by finding the location at which the estimate of the error in the reduced model is maximum, over a predetermined discrete set of parameters. The greedy sampling method was applied to find reduced models for the parameterized steady incompressible Navier-Stokes equations [30]. It was also combined with *a posteriori* error estimators for parameterized parabolic PDEs, and applied to several optimal control and inverse problems [14, 15].

Here, we employ the concept of greedy sampling, and formulate the task of determining appropriate sample locations as a sequence of adaptive model-constrained optimization problems. The optimization formulation treats the parameter space over which greedy sampling is applied as continuous; that is, we do not require the *a priori* selection of a discrete parameter set. Further, the objective functional to be minimized employs the true computed error between full-order and reduced-order outputs; thus, our approach is applicable in cases for which error estimators are unavailable. Unlike other sampling methods, we demonstrate that the model-constrained optimization sampling approach scales well to systems with a large number of parameters.

This article is organized as follows. Section 2 presents an overview of projection-based model reduction and discusses computation of the test basis using a minimum-residual approach. Section 3 then describes computation of the trial basis using the proposed model-constrained optimization adaptive sampling approach. Theoretical analysis of the method is presented in Section 4, while Section 5 demonstrates the methodology through an example that considers thermal analysis and design of a heat conduction fin. Finally, Section 6 presents conclusions.

2. Model reduction via projection. In this section, we present a general projection framework for model reduction, and discuss computation of the test basis using a minimum-residual approach.

2.1. General projection framework. Most large-scale model reduction frameworks are based on a projection approach, which can be described in general terms as follows. Consider the general dynamical system

$$\mathbf{R}(\dot{\mathbf{x}}(t), \mathbf{x}(t), \mathbf{z}, \mathbf{u}(t)) = \mathbf{0}, \quad \mathbf{x}(0) = \mathbf{x}^0, \quad \mathbf{y} = \ell(\mathbf{x}(t), \mathbf{z}, \mathbf{u}(t)), \tag{2.1}$$

where $\mathbf{x} = \mathbf{x}(t) \in \mathbb{R}^n$ is the state vector of dimension n, $\mathbf{z} \in \mathbb{R}^d$ is the parameter vector containing d input parameters of interest, t denotes time, $\mathbf{x}^0$ is the specified initial state, $\mathbf{u}(t) \in \mathbb{R}^p$ is a vector of p forcing inputs, $\mathbf{y} \in \mathbb{R}^q$ is a vector containing q outputs of interest computed by some output operator ℓ, and $\mathbf{R}$ is some discrete operator (for example, a residual operator resulting from a spatial discretization of a set of PDEs). We are interested in large-scale systems of the form (2.1), typically with state order $n > 10^5$, which might arise from spatial discretization of a set of PDEs, or from a naturally discrete large-scale simulation, such as in molecular dynamics or circuit simulation.

In the context of design and optimization, in which the system needs to be simulated repeatedly for many different design parameters $\mathbf{z}$, and for real-time applications, the full problem (2.1) is too computationally expensive to solve. Instead, we seek an approximate solution,

$$\tilde{\mathbf{x}}(t) = \mathbf{\Phi}\mathbf{x}_r(t), \tag{2.2}$$

where $\mathbf{\Phi} = [\phi_1 \ \phi_2 \ \cdots \ \phi_m] \in \mathbb{R}^{n \times m}$ is the trial basis, which contains as columns the m basis vectors ϕ_i, and $\mathbf{x}_r(t) \in \mathbb{R}^m$ is the vector of the coordinates of the approximate solution $\tilde{\mathbf{x}}$ in the reduced basis $\mathbf{\Phi}$. We also introduce the test basis $\mathbf{\Psi} \in \mathbb{R}^{n \times m}$, giving the mth-order reduced model of (2.1) as

$$\mathbf{\Psi}^T\mathbf{R}(\mathbf{\Phi}\dot{\mathbf{x}}_r(t), \mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t)) = \mathbf{0}, \quad \mathbf{x}_r(0) = \mathbf{\Psi}^T\mathbf{x}^0, \quad \mathbf{y}_r = \ell(\mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t)), \tag{2.3}$$

where $\mathbf{y}_r \in \mathbb{R}^q$ is the approximate output. We have used the reduced transform (2.2) and the assumption $\mathbf{\Psi}^T\mathbf{\Phi} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix, to obtain the initial condition for the reduced state, $\mathbf{x}_r(0)$. If the test space is the same as the trial space, i.e. $\mathbf{\Psi} = \mathbf{\Phi}$, the reduced system (2.3) is obtained via a Galerkin projection. If the test space is different from the trial space, the reduced system (2.3) is obtained via a Petrov-Galerkin projection.

While the model reduction methodology applies to both steady and unsteady systems, in this paper we consider steady systems of the form

$$\mathbf{A}(\mathbf{z})\mathbf{x} = \mathbf{b}(\mathbf{z}), \quad \mathbf{y} = \mathbf{C}(\mathbf{z})\mathbf{x}, \tag{2.4}$$

where the matrices $\mathbf{A}(\mathbf{z}) \in \mathbb{R}^{n\times n}$, $\mathbf{b}(\mathbf{z}) \in \mathbb{R}^{n\times 1}$, and $\mathbf{C}(\mathbf{z}) \in \mathbb{R}^{q\times n}$ in (2.4) depend nonlinearly on the set of parameters $\mathbf{z}$. The parameters $\mathbf{z}$ could be, for example, coefficients of the PDEs, such as thermal conductivities, or shape parameters. The residual of the reduced state is then defined as

$$\mathbf{R}(\mathbf{x}_r, \mathbf{z}) = \mathbf{b}(\mathbf{z}) - \mathbf{A}(\mathbf{z})\mathbf{\Phi}\mathbf{x}_r, \tag{2.5}$$

and the projection-based model reduction technique yields a reduced system of the form

$$\mathbf{A}_r(\mathbf{z})\mathbf{x}_r = \mathbf{b}_r(\mathbf{z}), \quad \mathbf{y}_r = \mathbf{C}_r(\mathbf{z})\mathbf{x}_r, \tag{2.6}$$

where $\mathbf{A}_r(\mathbf{z}) = \mathbf{\Psi}^T\mathbf{A}(\mathbf{z})\mathbf{\Phi}$, $\mathbf{b}_r(\mathbf{z}) = \mathbf{\Psi}^T\mathbf{b}(\mathbf{z})$, $\mathbf{C}_r(\mathbf{z}) = \mathbf{C}(\mathbf{z})\mathbf{\Phi}$.

In the following we discuss a methodology to construct the test basis $\mathbf{\Psi}$ to obtain a guaranteed stable reduced model for problems that are linear in the state vector as in (2.4). Construction of the trial basis $\mathbf{\Phi}$ will be discussed in Section 3.

2.2. Construction of the test basis. For steady problems of the form (2.4), we first denote $\mathbf{e} = \mathbf{x} - \tilde{\mathbf{x}}$, the state error between the full solution and the reduced basis approximation. The $\mathbf{A}^T\mathbf{A}$–norm is defined as $\|\mathbf{v}\|_{\mathbf{A}^T\mathbf{A}} = \mathbf{v}^T\mathbf{A}^T\mathbf{A}\mathbf{v}, \forall \mathbf{v} \in \mathbb{R}^n$. From the residual-error relation $\mathbf{R}(\mathbf{x}_r, \mathbf{z}) = \mathbf{A}\mathbf{e}$, where $\mathbf{R}$ is defined in (2.5), it is easy to see that the following identity is true,

$$\|\mathbf{R}\|_2 = \|\mathbf{e}\|_{\mathbf{A}^T\mathbf{A}}. \tag{2.7}$$

If Galerkin projection, i.e. $\mathbf{\Psi} \equiv \mathbf{\Phi}$, is employed to find the reduced state $\mathbf{x}_r$, there is no guarantee that the reduced model will be stable if $\mathbf{A}$ is not symmetric positive definite.[1] This is because the reduced equation (2.6) is obtained by requiring the residual to be orthogonal to the reduced trial space, and hence the residual could be arbitrarily large while also being orthogonal to the reduced space.

Alternatively, the approximate solution can be chosen so as to minimize the residual in (2.5). This is related to least-squares weighted residual methods, in particular in the finite element context [4,8,9,19]. Recently, the minimum-residual approach has been successfully used in the reduced-basis context [22,26,27]. The minimum-residual statement can be expressed in terms of the following least-squares minimization,

$$\mathbf{x}_r = \arg \min_{\bar{\mathbf{x}}_r \in \mathbb{R}^m} \|\mathbf{R}\|_2^2 = \|\mathbf{b} - \mathbf{A}\mathbf{\Phi}\bar{\mathbf{x}}_r\|_2^2. \tag{2.8}$$

Setting the gradient of the least-squares function to zero gives the reduced model,

$$(\mathbf{A}\mathbf{\Phi})^T(\mathbf{A}\mathbf{\Phi})\mathbf{x}_r = (\mathbf{A}\mathbf{\Phi})^T\mathbf{b}. \tag{2.9}$$

Thus, the minimum-residual approach is equivalent to a Petrov-Galerkin projection with the test basis $\mathbf{\Psi} = \mathbf{A}\mathbf{\Phi}$. This particular Petrov-Galerkin projection is equivalent to Galerkin projection on the normal equation, and by (2.7) yields a reduced model that minimizes the state error in the $\mathbf{A}^T\mathbf{A}$–norm. We are now in a position to prove that the reduced model (2.9) obtained from the minimum-residual statement, or equivalently from the given Petrov-Galerkin projection, is guaranteed to be stable.

[1] If $\mathbf{A}$ is, however, symmetric positive definite one can prove that the Galerkin projection yields an optimal reduced model in the $\mathbf{A}$–norm, that is, the state error is minimized in the $\mathbf{A}$–norm.

THEOREM 2.1. *Assume* $\mathbf{A}$ *has a bounded condition number, then the reduced model* (2.9) *is stable in the sense that the state error is bounded. In particular, the following bound holds*

$$\|\mathbf{e}\|_2 \leq \frac{1}{\sigma^{\mathbf{A}}_{min}} \|\mathbf{R}(\mathbf{x}_r, \mathbf{z})\|_2, \tag{2.10}$$

where $\sigma^{\mathbf{A}}_{min}$ *is the smallest singular value of* $\mathbf{A}$*, the reduced state* $\mathbf{x}_r$ *is computed from equation* (2.9)*, and the residual* $\mathbf{R}$ *is defined in* (2.5)*.*

Proof: Making use of the inequality for compatible matrix and vector norms on the error-residual relation $\mathbf{e} = \mathbf{A}^{-1}\mathbf{R}$ and using the definition of the singular values of a matrix yields the bound (2.10). Since the reduced state found from equation (2.9) minimizes the residual, the residual is finite (because $\|\mathbf{R}(\mathbf{x}_r, \mathbf{z})\|_2 = \min_{\bar{\mathbf{x}}_r \in \mathbb{R}^m} \|\mathbf{R}(\bar{\mathbf{x}}_r, \mathbf{z})\|_2 \leq \|\mathbf{R}(\mathbf{0}, \mathbf{z})\|_2 = \|\mathbf{b}(\mathbf{z})\|_2$). In addition, $1/\sigma^{\mathbf{A}}_{min}$ is finite due to the bounded condition number assumption on $\mathbf{A}$. We conclude that the 2–norm error in the state is bounded. □

A priori convergence results are standard in the context of finite element methods and recently in the reduced basis context [14, 15, 31]. In this light, we present an *a priori* convergence result for the above steady reduced model.

THEOREM 2.2. *As the reduced basis* $\mathbf{\Phi}$ *is enriched, i.e. more basis vectors are added, the state error in the* $\mathbf{A}^T\mathbf{A}$*–norm is a non-increasing function of the number of reduced basis vectors. In particular, there exists* $m \leq n$ *at which the state error in the* $\mathbf{A}^T\mathbf{A}-$*norm is strictly monotone decreasing.*

Proof: Assume $\mathbf{\Phi} \in \mathbb{R}^{n\times m}$ is the current reduced basis. Now adding a new basis vector ϕ_{m+1}, the new reduced basis is $\tilde{\mathbf{\Phi}} = [\mathbf{\Phi}, \phi_{m+1}] \in \mathbb{R}^{n\times(m+1)}$. The minimum-residual statement of the new reduced model is

$$\mathbf{x}^*_r = \arg \min_{\tilde{\bar{\mathbf{x}}}_r \in \mathbb{R}^{m+1}} \|\mathbf{b} - \mathbf{A}[\mathbf{\Phi}, \phi_{m+1}]\tilde{\bar{\mathbf{x}}}_r\|^2_2. \tag{2.11}$$

Comparing the minimization statements (2.8) and (2.11), and using the fact that $\text{span}\{\mathbf{\Phi}\} \subset \text{span}\{\tilde{\mathbf{\Phi}}\}$ and $\mathbb{R}^m \subset \mathbb{R}^{m+1}$, it can be seen that $\bar{\mathbf{x}}_r$ is a special case of $\tilde{\bar{\mathbf{x}}}_r$, where the last element is zero. That is, in (2.11), the residual norm is minimized in a larger space, and hence the residual should be no larger than that of (2.8). Equivalently, the state error in the $\mathbf{A}^T\mathbf{A}-$norm is no larger when the reduced space is richer. Since the approximate solution is exact if $m = n$, there exists $m \leq n$ such that when more basis vectors are added the state error is smaller. □

Note that from the residual error relation $\mathbf{R} = \mathbf{Ae}$ and the output error relation $\mathbf{y} - \mathbf{y}_r = \mathbf{Ce}$, we conclude that as the reduced basis is enriched, the residual and the output error are also improved. This *a priori* convergence result is important for the theoretical development of our adaptive sampling method discussed in the next section.

3. Computation of the trial basis.

To determine the reduced model (2.3), we must determine the trial basis, $\mathbf{\Phi}$, that yields a sufficient reduction in the size of the system while accurately representing full-scale outputs over the parametric input space. In the case of the POD, the reduced basis is formed as the span of a set of state solutions, commonly referred to as snapshots. These snapshots are computed by solving the system (2.1) for selected values of the parameters $\mathbf{z}$. The quality of the resulting reduced-order model is highly dependent on the choice of parameters for which snapshots are computed. Two issues arise in selecting an appropriate sample set of parameters. First, choosing where and how many samples to generate is, in general, an ad-hoc process. One can use knowledge of the application at hand to determine representative parameters; however, there exist no guarantees on the quality of the resulting reduced-order model. Second, in the case that the parametric input space is of high dimension, the number of high-fidelity system solves required to adequately cover the space can become prohibitively large. Work has been done to address the first issue using adaptive sampling methods to create reduced models for optimal control applications [1, 2, 18, 20]. In the case of a high-dimensional parametric input space, the computational cost of determining the reduced basis by these techniques becomes prohibitive unless some sparse sampling strategy is employed.

3.1. Greedy sampling method. A recently proposed approach to address the challenge of sampling a high-dimensional parameter space to build a reduced basis is the greedy algorithm [14, 15, 31]. The basic idea of the greedy algorithm is to adaptively choose samples by placing the new sample point at the location in parameter space where the estimated error in the reduced model is maximal. The approach taken is to determine a trial parameter set with n_{trial} parameters. At each point in this parameter set, the current reduced model is solved to determine the reduced states, which are then used to compute an error estimate. The parameter location in the trial set at which the error estimator is maximal is selected as the next sample point. Full model information is then generated via a full-scale system solve at this location, the basis is updated using the resulting snapshot information, and the reduced model is updated. These steps are then repeated, using the updated reduced model to compute the reduced states at each of the n_{trial} parameters. In [14, 15], the greedy sampling algorithm was combined with an *a posteriori* error estimator and applied to optimal control and inverse problems. As described in the next section, we formulate the greedy sampling problem as a model-constrained optimization problem over the continuous space of parameters.

3.2. Greedy optimization problem. In each cycle of the greedy algorithm, the key step is to determine the location in parameter space at which the error in the reduced model outputs is maximal. For the general problem (2.1), given a current basis $\mathbf{\Phi}$, and a time horizon of interest t_f, we find the location in parameter space of maximum output error by solving the optimization problem

$$\max_{\mathbf{x},\mathbf{x}_r,\mathbf{z}} \mathcal{G} = \frac{1}{2}\int_0^{t_f} \|\mathbf{y} - \mathbf{y}_r\|_2^2 \, dt \tag{3.1}$$

where

$$\begin{aligned}
\mathbf{R}(\dot{\mathbf{x}}(t), \mathbf{x}(t), \mathbf{z}, \mathbf{u}(t)) &= \mathbf{0}, && (3.2)\\
\mathbf{x}(0) &= \mathbf{x}^0, && (3.3)\\
\mathbf{y} &= \ell(\mathbf{x}(t), \mathbf{z}, \mathbf{u}(t)), && (3.4)\\
\mathbf{\Psi}^T\mathbf{R}(\mathbf{\Phi}\dot{\mathbf{x}}_r(t), \mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t)) &= \mathbf{0}, && (3.5)\\
\mathbf{x}_r(0) &= \mathbf{\Psi}^T\mathbf{x}^0, && (3.6)\\
\mathbf{y}_r &= \ell(\mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t)), && (3.7)\\
\mathbf{z}_{\min} \leq \mathbf{z} &\leq \mathbf{z}_{\max}, && (3.8)
\end{aligned}$$

where $\mathbf{z}_{\min}$ and $\mathbf{z}_{\max}$ are respectively lower and upper bounds on the parameter vector $\mathbf{z}$. Since the cost functional involves the error between the full and the reduced outputs, both the full model (3.2)–(3.4) and the reduced model (3.5)–(3.7) are needed to define the error and therefore are imposed as constraints in the optimization problem.

To solve the optimization problem (3.1)–(3.8), each optimization iteration may well be expensive, because the constraints include the full model. If a tight output error bound exists, it could be used as the cost functional instead of the true output error [15, 26, 31]. In that case, the constraints comprise only the reduced model and the bound constraints, and solving the optimization problem is much less expensive since it involves no full system solves. However, for a general problem, a good error bound may not be available. Alternatively, an error indicator, for example the squared norm of the residual, $\|\mathbf{R}(\mathbf{\Phi}\dot{\mathbf{x}}_r(t), \mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t))\|_2^2$, can be employed. In such cases, denote the output error bound or the residual error indicator as $\mathcal{Q}(\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t))$, and the optimization problem now reads

$$\max_{\mathbf{x},\mathbf{x}_r,\mathbf{z}} \mathcal{G} = \int_0^{t_f} \mathcal{Q}(\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t))\ dt \tag{3.9}$$

where

$$\mathbf{\Psi}^T\mathbf{R}(\mathbf{\Phi}\dot{\mathbf{x}}_r(t), \mathbf{\Phi}\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t)) = \mathbf{0}, \tag{3.10}$$

$$\mathbf{x}_r^0 = \mathbf{\Psi}^T\mathbf{x}^0, \tag{3.11}$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max}. \tag{3.12}$$

The analysis of the method, presented in Section 4, is developed based on the optimization problem using the true output error, (3.1)–(3.8). However, by removing the full model constraints and using $\mathcal{Q}(\mathbf{x}_r(t), \mathbf{z}, \mathbf{u}(t))$ instead of $\frac{1}{2}\|\mathbf{y} - \mathbf{y}_r\|_2^2$ in the cost functional, the results also hold for the optimization problem (3.9)–(3.12).

3.3. Greedy adaptive sampling via model-constrained optimization. The model-constrained adaptive sampling approach iteratively builds the basis by solving a sequence of optimization problems of the form (3.1)–(3.8) (or (3.9)–(3.12)). We denote the parameter vector that solves the maximization problem (3.1)–(3.8) (or (3.9)–(3.12)) by $\mathbf{z}^*$. Next, we compute the solution $\mathbf{x}(t)$ of the full system at this (locally) worst-case parameter value $\mathbf{z}^*$. This solution information is added to the basis $\mathbf{\Phi}$, for example using the POD. The procedure is then repeated by solving the optimization problem (3.1)–(3.8) (or (3.9)–(3.12)) with the updated basis $\mathbf{\Phi}$. Thus, we are using a systematic, adaptive error metric based on the ability of the reduced-order model to capture the outputs of interest in order to choose the sample locations that are (locally) the worst case scenarios. This adaptive sampling approach is summarized in the following algorithm.

Algorithm 3.1.

Model-Constrained Adaptive Sampling Procedure

1. *Given a current reduced basis $\mathbf{\Phi}$ and initial guess $\mathbf{z}^0$, solve the optimization problem* (3.1)–(3.8) *(or* (3.9)–(3.12)*) to find the location in parameter space at which the error is maximized, i.e. find $\mathbf{z}^* = \arg\max \mathcal{G}(\mathbf{z})$.*
2. *If $\mathcal{G}(\mathbf{z}^*) < \varepsilon$, where ε is the desired level of accuracy, then terminate the algorithm. If not, go to the next step.*
3. *With $\mathbf{z} = \mathbf{z}^*$, solve the full system* (3.2) *to compute the state solutions $\mathbf{x}(t)$, $t = (0, t_f)$. Use the span of these state solutions to update the basis $\mathbf{\Phi}$. Go to Step 1.*

In the case of a steady problem, in Step 3 we add just one vector to the basis (using appropriate orthogonalization). For unsteady problems, we may add several basis vectors at each cycle, depending on the amount of information contained in the state solutions $\mathbf{x}(t)$ (determined, for example using POD). The number of basis vectors in this case is chosen so that $\mathcal{G}(\mathbf{z}^*) < \varepsilon$.

Initially, there are no basis vectors in the reduced basis, $\mathbf{\Phi} = \emptyset$, and the reduced model is a zero-order approximation of the full model. Step 1 of Algorithm 3.1 starts with an initial guess $\mathbf{z}^0$ in the parameter space and moves iteratively towards a local maximizer. To avoid convergence to a local maximum close to a previous sample location, and hence to explore the parameter space better, a random initialization of the optimization variables $\mathbf{z}$ is used. An initial guess is accepted only if it is sufficiently far from the previous sample locations and its corresponding cost functional is larger than ε. In particular, the smallest allowable distance between an initial guess and all the existing sample locations is chosen to be $0.5 \min_i\{z_{\max_i} - z_{\min_i}\}$.

3.4. Solution of the greedy optimization problem. In order to reduce the offline cost of constructing the reduced model and make the method tractable for large-scale systems, it is important to use an efficient optimization algorithm that allows us to exploit the structure of the system. In order to solve the constrained optimization problem (3.1)–(3.8), we choose to solve an equivalent bound-constrained optimization problem in the $\mathbf{z}$ variables by eliminating the state variables $\mathbf{x}$ and $\mathbf{x}_r$. That is, we replace $\max_{\mathbf{x},\mathbf{x}_r,\mathbf{z}} \mathcal{G}(\mathbf{x}, \mathbf{x}_r, \mathbf{z})$ with $\max_{\mathbf{z}} \mathcal{G}(\mathbf{x}(\mathbf{z}), \mathbf{x}_r(\mathbf{z}), \mathbf{z}) = \max_{\mathbf{z}} \mathcal{G}(\mathbf{z})$, where the dependence of $\mathbf{x}$ and $\mathbf{x}_r$ on $\mathbf{z}$ is implicit through the full equation (3.2) and reduced state equation (3.5).

The bound constrained optimization problem is given by

$$\max_{\mathbf{z}} \mathcal{G}(\mathbf{z}) \tag{3.13}$$

subject to

$$\mathbf{z}_{\min} \le \mathbf{z} \le \mathbf{z}_{\max}. \tag{3.14}$$

We use the Coleman-Li subspace trust-region interior-reflective Newton framework [6] to solve the bound-constrained optimization problem (3.13)–(3.14) efficiently. That is, we use the conjugate gradient (CG) method to determine the subspace over which the linear system of equations arising at each Newton step is solved, and globalize by a trust region scheme (e.g. [24]). This method combines the rapid locally-quadratic convergence rate properties of Newton's method, the effectiveness of trust region globalization for treating ill-conditioned problems, and the Eisenstat-Walker idea of preventing oversolving [12].

The cost of the Coleman-Li method in the context of model-constrained optimization problems such as (3.13)–(3.14) is dominated by the computation of $\boldsymbol{\nabla}\mathcal{G}$, i.e. the gradient of $\mathcal{G}$ with respect to $\mathbf{z}$, and the Hessian-vector products arising at each CG step. The gradient can be computed efficiently—i.e. at the cost of a pair of (full- and reduced-scale) model solves—using an adjoint method. To illustrate, we give expressions for adjoint-based computation of the gradient in the case of steady problems of the form (2.4). We first restate the greedy optimization problem (at a typical adaptive cycle) in the model-constrained form:

$$\max_{\mathbf{x},\mathbf{x}_r,\mathbf{z}} \mathcal{G} = \frac{1}{2}\|\mathbf{C}(\mathbf{z})\mathbf{x} - \mathbf{C}_r(\mathbf{z})\mathbf{x}_r\|_2^2, \tag{3.15}$$

where

$$\mathbf{A}(\mathbf{z})\mathbf{x} = \mathbf{B}(\mathbf{z}), \tag{3.16}$$

$$\mathbf{A}_r(\mathbf{z})\mathbf{x}_r = \mathbf{B}_r(\mathbf{z}), \tag{3.17}$$

$$\mathbf{z}_{\min} \le \mathbf{z} \le \mathbf{z}_{\max}. \tag{3.18}$$

Next, we define the Lagrangian functional

$$\begin{aligned}\mathcal{L}(\mathbf{x},\mathbf{x}_r,\mathbf{z},\boldsymbol{\lambda},\boldsymbol{\lambda}_r) &= \mathcal{G}(\mathbf{x},\mathbf{x}_r,\mathbf{z}) + \boldsymbol{\lambda}^T[\mathbf{A}(\mathbf{z})\mathbf{x} - \mathbf{B}(\mathbf{z})] \\ &\quad + \boldsymbol{\lambda}_r^T[\mathbf{A}_r(\mathbf{z})\mathbf{x}_r - \mathbf{B}_r(\mathbf{z})]\,, \end{aligned} \tag{3.19}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^n$ and $\boldsymbol{\lambda}_r \in \mathbb{R}^m$ are the full and the reduced adjoint variables that respectively enforce the full and the reduced equations. Requiring the gradient of the Lagrangian with respect to the full states $\mathbf{x}$ to vanish yields the full adjoint equations,

$$\mathbf{A}^T(\mathbf{z})\boldsymbol{\lambda} = \mathbf{C}^T(\mathbf{z})\,[\mathbf{C}_r(\mathbf{z})\mathbf{x}_r - \mathbf{C}(\mathbf{z})\mathbf{x}]\,, \tag{3.20}$$

while requiring the gradient of the Lagrangian with respect to the reduced states $\mathbf{x}_r$ to vanish yields the reduced adjoint equations,

$$\mathbf{A}_r^T(\mathbf{z})\boldsymbol{\lambda}_r = \mathbf{C}_r^T(\mathbf{z})\,[\mathbf{C}(\mathbf{z})\mathbf{x} - \mathbf{C}_r(\mathbf{z})\mathbf{x}_r]\,. \tag{3.21}$$

Finally, the ith component of $\boldsymbol{\nabla}\mathcal{G}$ is given by the derivative of the Lagrangian with respect to z_i,

$$\begin{aligned}\frac{\partial \mathcal{G}}{\partial z_i} &= \left[\frac{\partial \mathbf{C}}{\partial z_i}(\mathbf{z})\mathbf{x} - \frac{\partial \mathbf{C}_r}{\partial z_i}(\mathbf{z})\mathbf{x}_r\right]^T [\mathbf{C}(\mathbf{z})\mathbf{x} - \mathbf{C}_r(\mathbf{z})\mathbf{x}_r] + \boldsymbol{\lambda}^T\left[\frac{\partial \mathbf{A}}{\partial z_i}(\mathbf{z})\mathbf{x} - \frac{\partial \mathbf{B}}{\partial z_i}(\mathbf{z})\right] \\ &\quad + \boldsymbol{\lambda}_r^T\left[\frac{\partial \mathbf{A}_r}{\partial z_i}(\mathbf{z})\mathbf{x}_r - \frac{\partial \mathbf{B}_r}{\partial z_i}(\mathbf{z})\right]. \end{aligned} \tag{3.22}$$

Thus, in order to compute $\boldsymbol{\nabla}\mathcal{G}$, we (1) solve the full and reduced state equations (3.16) and (3.17) for the full and reduced state variables $\mathbf{x}$ and $\mathbf{x}_r$; (2) substitute them into the right sides of the full and reduced adjoint equations (3.20) and (3.21), and solve the latter for the full and reduced adjoint variables $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}_r$; and (3) evaluate the gradient expression (3.22) using the computed full and reduced state and adjoint variables.

The Hessian-vector product as required by CG is computed in a matrix-free manner; because it is a directional derivative of the gradient, its computation similarly involves solution of state-like and adjoint-like equations. Therefore, the optimization algorithm requires solution of only a pair of state and adjoint systems at each CG iteration.

4. Analysis of the Adaptive Sampling Approach.

In this section, we prove that a solution of the optimization problem (3.13)–(3.14), and hence the optimization problem (3.1)–(3.8) or (3.9)–(3.12), exists, and discuss the uniqueness of that solution. We also show that the adaptive sampling methodology does not revisit previously sampled points.

4.1. Existence and uniqueness of a maximizer in each greedy optimization problem.

To begin, let us recall the following theorem.

Theorem 4.1. *If $\mathcal{G}(z_1,\dots,z_d) : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ is continuous and Ω is a compact subset of $\mathbb{R}^d$, then there is a maximum point $\bar{\mathbf{z}} \in \Omega$ such that $\mathcal{G}(\mathbf{z}) \leq \mathcal{G}(\bar{\mathbf{z}}), \forall \mathbf{z} \in \Omega$.*

Proof: The proof can be found, for example, in [28]. □

This theorem shows that the greedy optimization problem in each adaptive cycle has at least one solution by the following corollary.

Corollary 4.2. *In each adaptive cycle, assume that the cost functional is a continuous function in the parameter $\mathbf{z}$. Then there exists at least one solution for the optimization problem* (3.13)–(3.14)*.*

Proof: Denote Ω as the parameter set defined by the bound constraints (i.e. a d-cell [28]). Then, it can be proved that a d-cell is compact. Since the states are eliminated, the cost function $\mathcal{G}(\mathbf{z})$ is only a function of the parameter $\mathbf{z}$. Therefore, if the cost functional $\mathcal{G}(\mathbf{z})$ is a continuous function on Ω, Theorem 4.1 applies and Corollary 4.2 is proved. That is, there exists a solution (a global maximizer according to the theorem) to the optimization problem (3.13)–(3.14) in each adaptive cycle. □

Clearly uniqueness is not guaranteed in the general case since there may be many local maximizers.

4.2. Properties of the adaptive sampling approach. The following theorem and corollary show that the adaptive sampling strategy does not revisit previously sampled points.

THEOREM 4.3. *Assume that the full model is linear in state* $\mathbf{x}$. *Then, in the* k^{th} *adaptive cycle, the cost functional is less than* ε *at all the maximizers found in the previous cycles* $\bar{k} < k$.

Proof: Recall that in Step 3 of Algorithm 3.1 the span of the state solutions at the local maximizers found in previous cycles $\bar{k} < k$ are used as basis vectors such that the cost functional at these local maximizers is less than ε. Furthermore, we proved in Section 2 that, for problems that are linear in state $\mathbf{x}$, as the reduced basis is enriched, the error in the reduced model cannot increase. As a result, the cost functional in the k^{th} adaptive cycle is less than ε at all the maximizers found in the previous cycles $\bar{k} < k$. □

As a consequence of the above theorem, the corollary below is an important result for the adaptive sampling approach.

COROLLARY 4.4. *Assume that the full model is linear in state* $\mathbf{x}$. *Then, the adaptive sampling approach will never sample at previously-sampled points in the parameter space.*

Proof: By definition in (3.1), the cost functional is non-negative. To prove the corollary, it is sufficient to show that in the kth adaptive cycle the maximizer must be different from the maximizers found in the previous cycles. First, recall that the cost functional in the current adaptive cycle is smaller than ε at all previous maximizers, as proved in Theorem 4.3. Second, we start only at an initial guess where the cost function is greater than ε. Third, the optimization solver accepts an iterate only if the cost functional is larger than that at the previous iterate. Using these three properties, we conclude that the cost functional at a new maximizer must be larger than ε. Therefore, the maximizer found in the kth adaptive cycle must be different from the previous maximizers. □

5. Results.

Results are presented for a steady problem that is linear in state, but has nonlinear dependence on a set of parameters. We first compare the model-constrained adaptive sampling methodology using the true error and an error indicator. The model-constrained approach is then compared to the greedy sampling method and to several statistically-based sampling methods.

5.1. Problem description. The example considered is thermal analysis and design of a heat conduction fin, as shown in Figure 5.1. The vertical post has eight sub-regions denoted as $\Omega_9, \ldots, \Omega_{16}$ with corresponding thermal conductivities $\kappa_9, \ldots, \kappa_{16}$. Each of the four horizontal subfins has two different sub-regions, Ω_{2i-1} and Ω_{2i} for the i^{th} subfin, where $i = 1, \ldots, 4$, and these sub-regions have different thermal conductivities denoted as $\kappa_1, \ldots, \kappa_8$. In addition, the geometries of the sub-regions of the post and the subfins are considered to vary with parameters denoted as $b_1, \dots, b_{17}$, as shown in Figure 5.1. Another parameter of interest is the Biot number, Bi, which characterizes the convective heat loss to the air at the fin surfaces. Therefore, we have in total 34 parameters, which are represented by the vector of parametric inputs $\mathbf{z} = \{z_1, \dots, z_{34}\}$, where $z_i = \kappa_i$, $i = 1, \dots, 16$, $z_{17} = \text{Bi}$, and $z_{17+j} = b_j$, $j = 1, \dots, 17$.

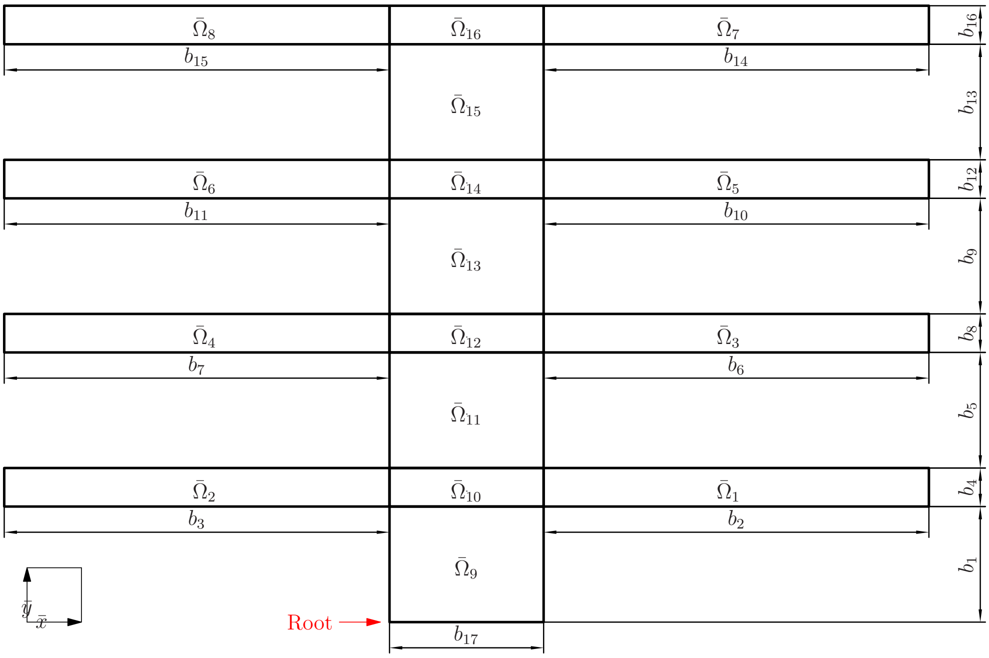


FIG. 5.1. *The thermal fin geometry and subdomain definitions.*

The steady-state temperature distribution within the fin, $w(\mathbf{z})$, is governed by the following elliptic PDE

$$-\kappa_i \nabla^2 w^i = 0 \text{ in } \Omega_i, \quad i = 1, \dots, 16, \tag{5.1}$$

where w^i denotes the restriction of w to Ω_i, $\nabla = \frac{\partial}{\partial \bar{x}}\hat{\mathbf{i}} + \frac{\partial}{\partial \bar{y}}\hat{\mathbf{j}}$, and $\nabla^2 = \frac{\partial^2}{\partial \bar{x}^2} + \frac{\partial^2}{\partial \bar{y}^2}$, where $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ are unit vectors pointing in the $\bar{x}$– and $\bar{y}$–directions. The continuity of temperature and heat flux at the conductivity discontinuity interfaces $\Gamma^{\text{int}}_{ij} = \partial\Omega_i \cap \partial\Omega_j$ for two adjacent regions Ω_i and Ω_j, where $\partial\Omega_i$ and $\partial\Omega_j$ denote the boundary of Ω_i and Ω_j respectively, are ensured by the following interface condition,

$$\left.\begin{array}{rcl} w^i & = & w^j \\ -\kappa_i(\nabla w^i \cdot \hat{\mathbf{n}}^i) & = & -\kappa_j(\nabla w^j \cdot \hat{\mathbf{n}}^j) \end{array}\right\} \text{ on } \Gamma^{\text{int}}_{ij}, \tag{5.2}$$

where $\hat{\mathbf{n}}^i$ and $\hat{\mathbf{n}}^j$ denote the outward unit normals of Ω_i and Ω_j on the interface Γ^{int}_{ij}, respectively. In order to model the convective heat losses on the external surface of a region Ω_i, i.e. $\Gamma^{\text{ext}}_i = \partial\Omega_i \setminus \Gamma^{\text{int}}_{ij}$ and $\Gamma^{\text{ext}}_i \neq \emptyset$, we use the following Robin boundary condition,

$$-\kappa_i(\nabla w^i \cdot \hat{\mathbf{n}}^i) = \text{Bi } w^i \text{ on } \Gamma^{\text{ext}}_i \text{ if } \Gamma^{\text{ext}}_i \neq \emptyset, \; i = 1, \dots, 16. \tag{5.3}$$

Finally, to model the heat source at the root, the Neumann flux boundary condition is imposed as

$$-\kappa_9(\nabla w^9 \cdot \hat{\mathbf{n}}^9) = -1 \text{ on } \Gamma^{\text{root}}. \tag{5.4}$$

The output of interest is chosen to be the average temperature over the entire thermal fin,

$$y = \frac{\sum_{i=1}^{16} \int_{\Omega_i} w \, d\Omega_i}{\sum_{i=1}^{16} \int_{\Omega_i} 1 \, d\Omega_i}. \tag{5.5}$$

The problem is discretized in space with linear triangular finite elements. This yields a system of the form (2.4), where the vector $\mathbf{x}$ consists of the nodal temperature values. The matrices $\mathbf{A}(\mathbf{z})$, $\mathbf{b}(\mathbf{z})$, and $\mathbf{C}(\mathbf{z})$ can be written as affine functions of the parameters,

$$\mathbf{A}(\mathbf{z}) = \sum_{i=1}^{49} \theta_i(\mathbf{z})\mathbf{A}_{q_i}, \tag{5.6}$$

$$\mathbf{b}(\mathbf{z}) = z_{34}\mathbf{b}_q, \tag{5.7}$$

$$\mathbf{C}(\mathbf{z}) = \sum_{i=1}^{16} \eta_i(\mathbf{z})\mathbf{C}_{q_i}, \tag{5.8}$$

where $\mathbf{A}_{q_i}$, $\mathbf{b}_q$ and $\mathbf{C}_{q_i}$ are the appropriate finite element matrices that do not depend on the parameters $\mathbf{z}$, and the θ_i and η_i coefficients are known nonlinear functions of the parameters $\mathbf{z}$. The derivations of these quantities are given in full in [7]. This affine decomposition together with the projection-based model reduction technique enables us to obtain efficient reduced models, i.e. models for which online solution cost is independent of the full model size, since the projection of the finite element matrices can be done in the offline phase. The spatial grid employed in the results presented here has 17,899 nodes; thus the full-scale system has dimension $n = 17,899$. A coarser grid is shown in Figure 5.2 to illustrate adaptation around the reentrant corners.

5.2. Model-constrained adaptive sampling performance. Algorithm 3.1 is applied to determine a reduced model that spans the parametric input space described for the thermal fin problem. We first compare the offline cost and the quality of the reduced model using the model-constrained adaptive sampling approach in Algorithm 3.1 with an error indicator and with the true output error. For both approaches, the same sequence of initial guesses is used in each greedy optimization cycle, and Gram-Schmidt orthogonalization is used to update the reduced basis with the solution at the optimal parameter point $\mathbf{z}^*$. For all problems considered in the following, since it is too computationally expensive to cover the large parameter spaces with full factorial search, we define the maximum output error to be the maximum error between the full and the reduced model outputs over a random set of 10^5 parameters in the parameter space under consideration. We choose ε in Step 2 of Algorithm 3.1 to be $\varepsilon = 10^{-14}$.

The examples considered here have 11 and 21 parameters, that is,

$$\mathbf{z} = \{\kappa_1, \ldots, \kappa_5, \text{Bi}, b_1, \ldots, b_5\}, \tag{5.9}$$

or

$$\mathbf{z} = \{\kappa_1, \dots, \kappa_{10}, \mathrm{Bi}, b_1, \dots, b_{10}\}. \tag{5.10}$$

The baseline parameter values are chosen to be: $\kappa_1 = \kappa_2 = 0.4$; $\kappa_3 = \kappa_4 = 0.6$; $\kappa_5 = \kappa_6 = 0.8$; $\kappa_7 = \kappa_8 = 1.2$; $\kappa_i = 1.0$, $i = 9, \dots, 16$; $\mathrm{Bi} = 1.0$; $b_{4i-3} = 0.75$, $i = 1, \dots, 4$; $b_{4i-2} = 2.5$, $i = 1, \dots, 4$; $b_{4i-1} = 2.5$, $i = 1, \dots, 4$; $b_{4i} = 0.25$, $i = 1, \dots, 4$; $b_{17} = 1.0$. If any parameter is not allowed to vary, it takes its corresponding baseline value.

We use the number of full matrix factorizations as the measure for the offline cost to compare the quality of the reduced models, since this is the dominant cost of the model-constrained adaptive sampling algorithm. For the 11-parameter case, Figure 5.3(a) shows that the required number of matrix factorizations to reach a given error level is approximately an order of magnitude larger for the true-error approach; however, for the same number of basis functions retained in the reduced basis, Figure 5.3(b) shows that using the true error rather than the indicator leads to more efficient (i.e. smaller for a given error level) reduced models. This result might be intuitively expected, since the optimal parameter points based on the true output error should better target reduction of the true output error than those points selected using the error indicator. However, there is no guarantee that this will always be the case, as shown in Figure 5.4 for the case of 21 parameters. Figure 5.4(a) shows that the number of matrix factorizations is again about an order of magnitude larger for the true-error approach. For smaller basis sizes, Figure 5.4(b) shows that the output error is again smaller than for models obtained using the error indicator; however, for larger basis sizes, the true-error and error-indicator approaches give equally good reduced models.

Figures 5.3 and 5.4 demonstrate a general tradeoff in the model-constrained adaptive sampling methodology: if one is willing to invest larger offline cost to compute the reduced model, then using the true error to select the parameter sample points

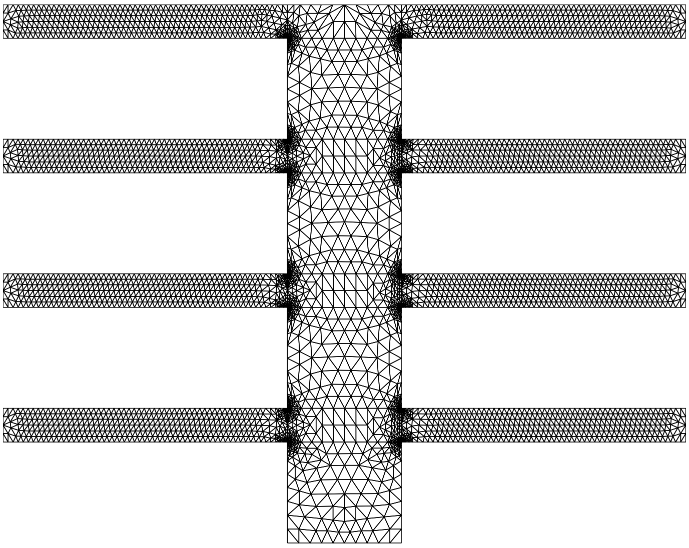


FIG. 5.2. *Typical mesh for discretization of the heat conduction problem. (Actual mesh used for results in Section 5.2 has approximately double the resolution.)*

can lead to more efficient models. For some problems, such as real-time applications, minimizing the size of the reduced model may be critical; in that case, one might choose to use the true-error approach. For very large-scale problems, however, the cost of the true-error approach may be prohibitively high; in that case, the error indicator is an attractive option. In many of the numerical experiments performed for the thermal fin problem, and as demonstrated by the results in Figure 5.4, the difference in quality between the true-error and error-indicator sample points tends to be larger in early greedy cycles. Since the error function becomes more multimodal as the number of greedy cycles increases, the chances of sampling at a local (versus global) maximum are increased, and thus the difference between the error-indicator and true-error approaches may not be as great. One could therefore also conceive of using a combination of the two error metrics, i.e. using the true error for early greedy cycles and the error indicator for later cycles, in an attempt to balance offline cost with the quality of the reduced model.

When using the true error as the objective function, intermediate state and adjoint solutions are available, since solution of each greedy optimization problem requires some number of full forward and adjoint solves. One could consider adding this information to the basis on each greedy cycle, either using Gram-Schmidt orthogonalization or a method such as POD to select a subset of important information. Our numerical experiments showed that including this information improved the offline efficiency of the method; that is, for the same number of full matrix factorizations, one could achieve lower reduced model error. However, including the intermediate information degraded the quality of the reduced models, so that more basis vectors were required to achieve a given level of error. Again, this highlights the tradeoff between offline cost and reduced model quality.

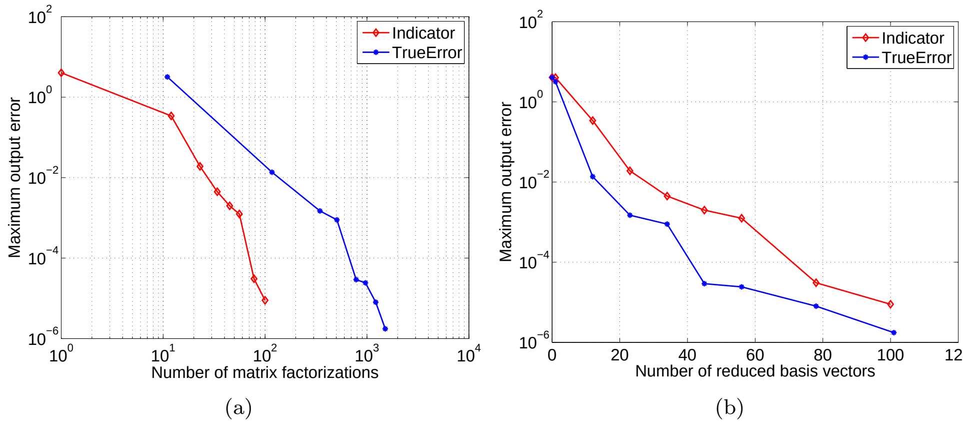
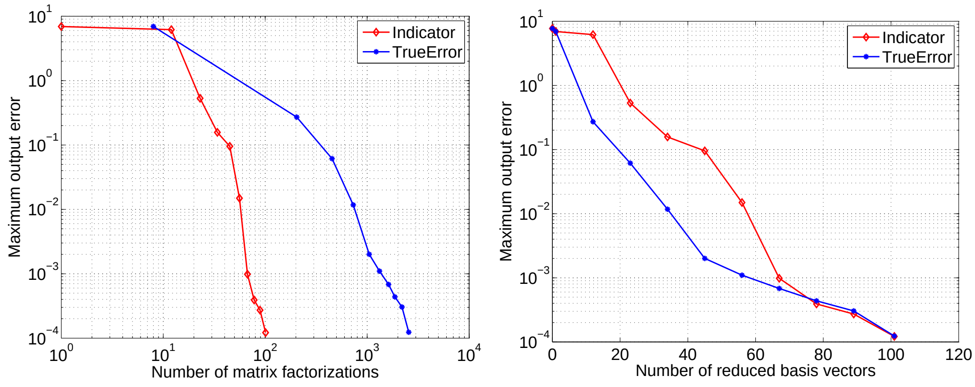


FIG. 5.3. *Error indicator versus true output error for the thermal fin with 11 parameters. The same sequence of initial guesses is used for both true-error and error-indicator approaches, and the Gram-Schmidt procedure is used to update the reduced basis.*

5.3. Comparison with other sampling approaches. We first compare the model-constrained adaptive sampling approach with the greedy sampling method described in Section 3.1. The $\mathcal{L}_2$–norm of the residual is used as the error indicator for both model-constrained and greedy sampling; hence, only snapshots at the optimal parameter points are computed. In both cases, a total of 100 greedy cycles is used; thus, both methods will provide 100 sample points at which the full-scale model is



(a) (b)

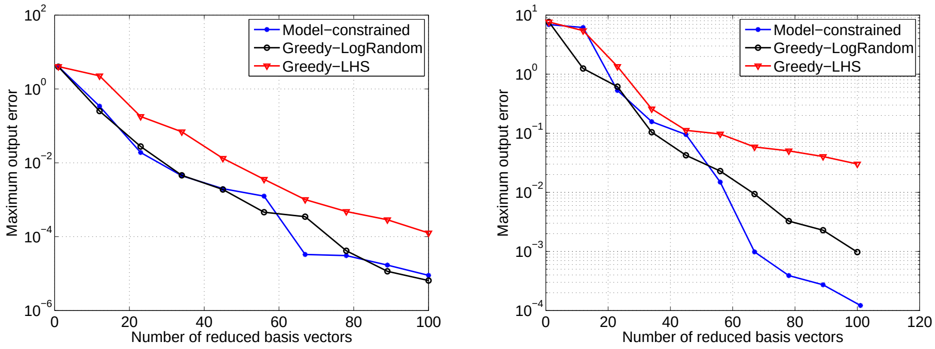
FIG. 5.4. *Error indicator versus true output error for the thermal fin with 21 parameters. The same sequence of initial guesses is used for both true-error and error-indicator approaches, and the Gram-Schmidt procedure is used to update the reduced basis.*

solved to compute the snapshots, which are then used to form the reduced basis. To generate the trial set containing n_{trial} parameters for the greedy sampling approach, we use two different methods, namely logarithmic random sampling and Latin hypercube sampling (LHS). For the model-constrained adaptive sampling approach, we use logarithmic random sampling to generate an initial guess for each greedy optimization problem, which has the form (3.9)–(3.12).

Figures 5.5(a) and 5.5(b) show the comparison for the three methods for the case of 11 and 21 parameters, respectively. It can be seen that the maximum output error obtained using the model-constrained approach is, for the most part, comparable to that of the greedy sampling method with logarithmic random trial points. For the case with 21 parameters, the model-constrained adaptive sampling method is able to achieve an order of magnitude smaller error than the greedy sampling approach for larger reduced model sizes. Using the greedy sampling method with LHS trial points led to larger errors than those obtained using the other two methods.

A key difference between the methods is that the greedy sampling approach finds globally-optimal parameter points within the discrete trial set (via exhaustive search) while the model-constrained approach finds locally-optimal parameter points in the continuous parameter space (by solving an optimization problem with a gradient method). As a result, unless n_{trial} is sufficiently large, the trial set may not adequately cover important parameter regions, particularly as the dimension of the parametric space becomes large. This difference is highlighted by the results in Figure 5.5(b), where even 10^4 trial points were not sufficient for the greedy sampling method to find near-optimal sample points for larger reduced model sizes.

Although the total number of reduced basis vectors, and hence full-scale matrix factorizations, is 100 for both the model-constrained and the greedy sampling methods, the actual offline cost differs substantially between the two. Table 5.1 compares the CPU time required to compute the reduced basis for the three approaches for the case of 21 parameters. It can be seen that the model-constrained approach is approximately 16 times faster than the greedy sampling approaches. This difference is due to the need for exhaustive search over 10^4 trial points on every greedy iteration in the greedy sampling method, which for this case is a significant component of the



(a) 11-parameter case

(b) 21-parameter case

FIG. 5.5. *Model-constrained adaptive sampling approach versus greedy sampling approaches (Greedy-LogRandom: greedy sampling with logarithmic random trial parameter set; Greedy-LHS: greedy sampling with LHS trial parameter set) over 100 greedy cycles. A trial set with* 10^4 *parameter points is used in the greedy sampling approaches.*

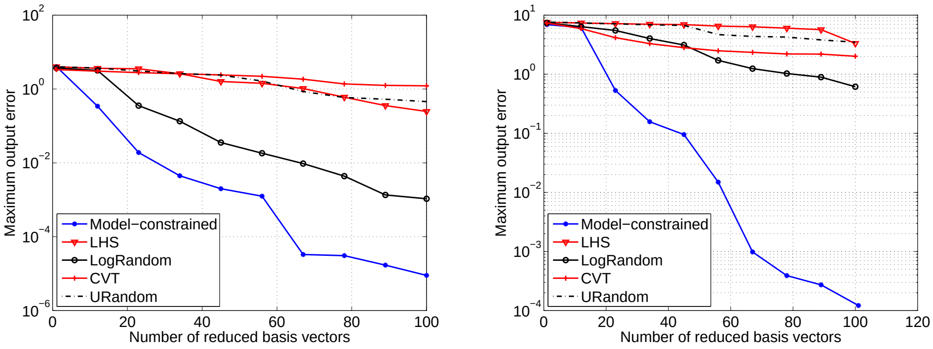
offline cost. This could be a particular concern as the number of parameters, d, and hence the necessary number of trial points, n_{trial}, increases.

TABLE 5.1
The offline cost in CPU time of the model-constrained adaptive sampling approach and the greedy sampling approaches for the case of 21 parameters. 100 greedy cycles are taken for all methods and $n_{trial} = 10^4$ *for the greedy sampling approaches. These computational results were obtained on a dual core personal computer with 3.2GHz Pentium processor.*

	Model-constrained adaptive sampling	Greedy sampling with logarithmic random	Greedy sampling with LHS
CPU time	0.58 hours	8.33 hours	8.33 hours

Next, we compare the model-constrained adaptive sampling method with statistically-based sampling methods in the context of snapshot generation for model reduction. In particular, we compare our model-constrained adaptive sampling with LHS sampling, uniform random sampling, logarithmic random sampling, and CVT sampling. For all methods, we take 100 sample points to generate 100 snapshots, which are then used to form the reduced basis. As can be seen in Figures 5.6(a) and (b) for 11 and 21 parameters, respectively, the model-constrained sampling method outperforms the other methods in the sense that, for a given basis size, the reduced model error is several orders of magnitude smaller than that obtained using the other methods. Furthermore, going from 11 to 21 parameters, the difference in accuracy of the reduced model using model-constrained sample points and those of other methods is larger. This reflects the fact that the model-constrained adaptive sampling is a model-based sampling method; that is, the parameter is sampled where the indicator of the error between the full and the reduced models is locally largest. The other methods, on the other hand, use no knowledge of the underlying model when selecting their sample points. As the dimension of the parametric space increases, it becomes more and more difficult to adequately cover the space with a reasonable number of samples using the statistically-based sampling methods.

Generating the sample points using either logarithmic random sampling or uniform random sampling is very rapid. For the other methods, there is a small amount of additional overhead in LHS (due to stratification and grouping processes), in CVT sampling (due to its iterative nature), and in model-constrained adaptive sampling (due to solving the optimization problem). We also note that while logarithmic random sampling is slightly less expensive (in terms of overhead) than the CVT and LHS sampling methods, it leads to more accurate reduced models in the case of the thermal fin problem.



(a) 11-parameter case

(b) 21-parameter case

FIG. 5.6. *Model-constrained adaptive sampling versus LHS, uniform random sampling (URandom), logarithmic random sampling (LogRandom) and CVT sampling. 100 sample points are used for all methods.*

6. Conclusions.

This paper has demonstrated the effectiveness of the model-constrained adaptive sampling methodology for steady problems that are linear in state, but have nonlinear dependence on a set of parameters that describe geometry and PDE coefficients. While the analysis of the method presented here applies to this problem class, the methodology is broadly applicable to unsteady problems and to general nonlinear systems. In the general nonlinear case, however, one must address the challenge of carrying out online reduced-order model computations in an efficient manner that does not depend on the large-scale system dimension. This could be achieved, for example, by combining the adaptive sampling methodology with the interpolation method of [3].

Acknowledgments. This work was supported in part by the Singapore-MIT Alliance Computational Engineering Programme, Universal Technology Corporation under contract number 04-S530-0022-07-C1, technical contract monitor Dr. C. Cross of Air Force Research Laboratory, the Air Force Office of Sponsored Research under grant FA9550-06-0271, program director Dr. F. Fahroo, and the National Science Foundation under DDDAS grant CNS-0540372, program manager Dr. F. Darema. The authors also acknowledge helpful input from Prof. A.T. Patera of MIT.

REFERENCES

[1] K. Afanasiev and M. Hinze, *Adaptive control of a wake flow using proper orthogonal decomposition*, in Lecture Notes in Pure and Applied Mathematics, vol. 216, Marcel Dekker, 2001, pp. 317–332.

[2] E. Arian, M. Fahl, and E.W. Sachs, *Trust-region proper orthogonal decomposition for optimal flow control*, Tech. Report ICASE 2000-25, Institute for Computer Applications in Science and Engineering, May 2000.

[3] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera, *An "empirical interpolation" method: Application to efficient reduced-basis discretization of partial differential equations*, CR Acad Sci Paris Series., I (2004), pp. 339–667.

[4] P. Bochev and M. Gunzburger, *Finite element methods of least squares type*, SIAM Review, 40 (1998), pp. 789–837.

[5] B. Bond and L. Daniel, *Parameterized model order reduction of nonlbinear dynamical systems*, in Proceedings of the IEEE Conference on Computer-Aided Design, San Jose, CA, November 2005, IEEE.

[6] M.A Branch, T.F. Coleman, and Y. Li, *A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems*, SIAM J. Sci. Comput, 21 (1999), pp. 1–23.

[7] T. Bui-Thanh, *Model-Constrained Optimization Methods for Reduction of Parameterized Large-Scale Systems*, PhD thesis, MIT, June 2007.

[8] Z. Cai, R. Lazarov, T. Manteuffel, and S. McCormick, *First-order system least squares for second-order partial differential equations: Part I*, SIAM Journal on Numerical Analysis, 31 (1994), pp. 1785–1802.

[9] Z. Cai, T. Manteuffel, and S. McCormick, *First-order system least squares for second-order partial differential equations: Part II*, SIAM Journal on Numerical Analysis, 34 (1997), pp. 425–454.

[10] L. Daniel, O.C. Siong, L.S. Chay, K.H. Lee, and J. White, *Multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models*, Transactions on Computer Aided Design of Integrated Circuits, 23 (2004), pp. 678–693.

[11] Q. Du, V. Faber, and M. Gunzburger, *Centroidal Voronoi tessellations: Applications and algorithms*, SIAM Review, 41 (1999), pp. 637–676.

[12] S.C. Eisenstat and H.F. Walker, *Choosing the forcing terms in an inexact Newton method*, SIAM Journal on Scientific Computing, 17 (1996), pp. 16–32.

[13] B. I. Epureanu, E.H. Dowell, and K.C. Hall, *A parametric analysis of reduced order models of potential flows in turbomachinery using proper orthogonal decomposition*, in Proceedings of ASME TURBO EXPO, vol. 2001-GT-0434, New Orleans, LA, June 2001.

[14] M. Grepl, *Reduced-Basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations*, PhD thesis, MIT, Cambridge, MA, June 2005.

[15] M. Grepl and A. Patera, A posteriori *error bounds for reduced-basis approximations of parametrized parabolic partial differential equations*, ESAIM-Mathematical Modelling and Numerical Analysis (M2AN), 39 (2005), pp. 157–181.

[16] S. Gugercin, A. Antoulas, and C. Beattie, *A rational Krylov iteration for optimal $\mathcal{H}_2$ model reduction*, in Proceedings of MTNS 2006, 2006.

[17] M.D. Gunzburger, J.S. Peterson, and J.N. Shadid, *Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data*, Computer Methods in Applied Mechanics and Engineering, 196 (2007), pp. 1030–1047.

[18] M. Hinze and S. Volkwein, *Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control*, in Dimension Reduction of Large-Scale Systems, P. Benner, V. Mehrmann, and D. Sorensen, eds., Lecture Notes in Computational and Applied Mathematics, 2005, pp. 261–306.

[19] B.N. Jiang, *The Least-Squares Finite Element Method: Theory and Applications in Computational Fluid Dynamics and Electromagnetics*, Springer, New York, 1998.

[20] K. Kunisch and S. Volkwein, *Control of Burgers' equation by reduced order approach using proper orthogonal decomposition*, Journal of Optimization Theory and Applications, 102 (1999), pp. 345–371.

[21] H.V. Ly and H.T. Tran, *Modeling and control of physical processes using proper orthogonal decomposition*, Mathematical and Computer Modeling, 33 (2001), pp. 223–236.

[22] Y. Maday, A.T. Patera, and D.V. Rovas, *A blackbox reduced-basis output bound method for noncoercive linear problems*, in Studies in Mathematics and its Applications, D. Cioranescu and J.L. Lions, eds., vol. 31, 2002.

[23] M.D. McKay, R.J. Beckman, and W.J. Conover, *A comparison of three methods for selecting values of input variables in the analysis of output from a computer code*, Technometrics, 21 (1979), pp. 239–245.

[24] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer, New York, 1999.

[25] V.J. Romero, J.S. Burkardt, M.D. Gunzburger, and J.S. Peterson, *Initial evaluation of*

pure and latinized centroidal Voronoi tessellation for non-uniform statistical sampling, in Sensitivity Analysis of Model Output, K.M. Hanson and F.M. Hemez, eds., Los Alamos National Laboratory, 2005, pp. 380–401.

[26] D. Rovas, *Reduced-basis Output Bound Methods for Parametrized Partial Differential Equations*, PhD thesis, Massachusetts Institute of Technology, 2003.

[27] G. Rozza and K. Veroy, *On the stability of the reduced basis method for Stokes equations in parametrized domains.*, submitted to Elsevier Science, (2006).

[28] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, New York, third ed., 1976.

[29] K.C. Sou, A. Megretski, and L. Daniel, *A quasi-convex optimization approach to parameterized model-order reduction*, in IEEE/ACM Design Automation Conference, Anaheim, CA, June 2005.

[30] K. Veroy and A. Patera, *Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: Rigorous reduced-basis* a posteriori *error bounds*, International Journal for Numerical Methods in Fluids, 47 (2005), pp. 773–788.

[31] K. Veroy, C. Prud'homme, D.V. Rovas, and A.T. Patera, *A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations*, in Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL, 2003. AIAA Paper 2003-3847.