

**Reduced Sampling for Construction of Quadratic Response Surface  
Approximations Using Adaptive Experimental Design**

Victor M. Pérez

Department of Aerospace and Mechanical Engineering  
University of Notre Dame  
Notre Dame, IN 46556

John E. Renaud

Department of Aerospace and Mechanical Engineering  
University of Notre Dame  
Notre Dame, IN 46556  
jrenaud@nd.edu

Layne T. Watson

Departments of Computer Science and Mathematics  
Virginia Polytechnic Institute and State University  
Blacksburg, VA 24061  
ltw@cs.vt.edu

Short title: Reduced sampling for adaptive experimental design

## Abstract

**Purpose** — To reduce the computational complexity per step from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$  for optimization based on quadratic surrogates, where  $n$  is the number of design variables.

**Design/methodology/approach** — Applying nonlinear optimization strategies directly to complex multidisciplinary systems can be prohibitively expensive when the complexity of the simulation codes is large. Increasingly, response surface approximations, and specifically quadratic approximations, are being integrated with nonlinear optimizers in order to reduce the CPU time required for the optimization of complex multidisciplinary systems. For evaluation by the optimizer, response surface approximations provide a computationally inexpensive lower fidelity representation of the system performance. The *curse of dimensionality* is a major drawback in the implementation of these approximations as the amount of required data grows quadratically with the number  $n$  of design variables in the problem. In this paper a novel technique to reduce the magnitude of the sampling from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$  is presented.

**Findings** — The technique uses prior information to approximate the eigenvectors of the Hessian matrix of the response surface approximation and only requires the eigenvalues to be computed by response surface techniques. The technique is implemented in a sequential approximate optimization algorithm and applied to engineering problems of variable size and characteristics. Results demonstrate that a reduction in the data required per step from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$  points can be accomplished without significantly compromising the performance of the optimization algorithm.

**Originality/value** — A reduction in the time (number of system analyses) required per step from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$  is significant, even more so as  $n$  increases. The novelty lies in how only  $\mathcal{O}(n)$  system analyses can be used to approximate a Hessian matrix whose estimation normally requires  $\mathcal{O}(n^2)$  system analyses.

**Keywords** Response surface approximations, Sequential approximate optimization, Quadratic approximations, Multidisciplinary design optimization, Adaptive experimental design, Extended adaptive experimental design

**Paper type** Research paper

## Nomenclature

$A$	Array of experimental design points.
$\mathcal{A}$	Database of experimental design points and corresponding function values.
$E^n$	Real $n$ -dimensional Euclidean space.
$f$	Objective function.
$\tilde{f}^k$	Approximation to $f$ near $x^k$ .
$g$	Inequality constraint vector.
$g_i$	$i$ th inequality constraint.
$\tilde{g}^k$	Approximation to $g$ near $x^k$ .
$H_c^k$	Hessian matrix of function $c(x)$ at $x^k$ .
$\tilde{H}_c^k$	Approximation to Hessian matrix $H_c^k$ .
$n$	Dimension of design space.
$r$	Penalty parameter.
$U$	Orthogonal eigenvector matrix.
$x$	Vector of design variables.
$x^k$	Current design point at $k$ th iteration.

$\Delta$	Trust region radius.
$\epsilon$	SAO convergence criterion.
$\lambda$	Vector of Lagrange multipliers.
$\Phi$	Augmented Lagrangian function.
$\Psi_i$	Alternative form for inequality constraint $g_i$ .
$\rho$	Trust region ratio.
$\mathcal{U}$	Transformed design space.
AED	Adaptive experimental design.
CA	Contributing analysis.
CSSO	Concurrent subspace optimization.
DOE	Design of experiments.
EAED	Extended adaptive experimental design.
OA	Orthogonal array.
RSA	Response surface approximation.
SA	System analysis.
SAO	Sequential approximate optimization.

## 1. Introduction

The increasing demand for better simulation software has fueled the development of higher fidelity analysis software in the engineering community. This computer software is often expensive to execute, even with the availability of multiple processor computers and parallel systems. More complex simulations can be achieved when two or more of these analysis software packages are coupled together to create a multidisciplinary problem. A single function evaluation for such a system may require several iterative evaluations of the contributing analyses due to the input/output coupling of the software packages.

The solution of general nonlinear constrained optimization problems has been a topic of research for many years in the mathematical community. As a result a rich variety of optimization algorithms and software has been developed with continuous improvements. Some of these algorithms are very efficient and effective in solving smooth optimization problems. In optimization algorithms based on higher order information the system analysis has to be called several times at each iteration. When the analysis is performed for one of these multidisciplinary problems, the optimization process becomes very expensive.

In most analysis software, as finite element analysis or computational fluid dynamics, numerical integration, domain discretization, and numerical roundoff error produce noise in the output of the computer program. Applying gradient-based optimization algorithms to solve optimization problems based on these noisy outputs may create numerical difficulties such as premature convergence to spurious local optima.

For these two reasons—expense and output noise—the use of approximations in optimization is highly attractive. The approximations can smooth noisy behavior while providing an inexpensive analysis for the optimization process. The type of approximations used for optimization vary in nature and scope. By nature they can be grouped as physics-based approximations (Knill et al., 1999); parametric approximations such as artificial neural networks (Cheney and Light, 2000), low order polynomials (Hosder et al., 2001), and radial basis functions (Cheney and Light, 2000); and statistical approximations such as krigging models (Sacks et al., 1989; Welch and Sacks, 1991). By scope they can be divided into either global approximations, meant to represent the whole design

space, or local, valid in a reduced design space. A comprehensive survey of approximations in the context of optimization is given by Sobieski and Haftka (1997).

Often, second order polynomials (Taylor series) are used for local approximations. One of the main drawbacks for the use of local quadratic approximations in  $n$  dimensions is the need of sampling the high fidelity analysis an  $\mathcal{O}(n^2)$  number of times, precisely, at least  $(n+2)(n+1)/2$  data points are needed to determine the  $(n+2)(n+1)/2$  coefficients in a quadratic polynomial. In this paper, an extension to the adaptive experimental design algorithm introduced by Pérez et al. (2002) is presented to eliminate the need of  $\mathcal{O}(n^2)$  samplings (system analyses, possibly multidisciplinary) and replace them with  $\mathcal{O}(n)$  effective ones. For large  $n$ , this reduction in work per iteration is huge, and will result in significant cost savings, provided the number of iterations does not grow by  $\mathcal{O}(n)$ .

The paper is organized as follows. The remainder of this section presents the use of quadratic approximations in optimization, and underlines the importance of the need to reduce the order of the sampling due to what is known as the *curse of dimensionality*. The following sections introduce the reader to the adaptive experimental design (AED) technique developed by the authors, and describe the extended adaptive experimental design (EAED) algorithm proposed in this paper. The next section presents some test problems and results, with some closing remarks in the final section.

### 1.1 Quadratic approximations in optimization

Quadratic approximations are an important component in nonlinear programming. Almost all gradient based optimization algorithms use some type of second order information for the objective function and constraints. Due to the high cost or impossibility of computing the true Hessian matrix at each design point, optimization algorithms generally use approximations to the Hessian matrix. These are point approximations, i.e., second order Taylor approximations at the current design point. The most common type of approximations are Hessian updates, in which the optimizer uses zeroth and first order information to update the value of the approximate Hessian matrix. Examples of this type of update formula are the BFGS (named after its developers Broyden-Fletcher-Goldfarb-Shanno), the DFP (Davidon-Fletcher-Powell) formula, and the SR1 (symmetric-rank one) update (Nocedal and Wright, 1999). An alternate approach to compute quadratic approximations has been proposed by Canfield (2001), in which zeroth and first order information from the current and all previous design points is used to compute the approximation.

In surrogate-based optimization quadratic polynomials have been used to approximate the objective function and constraints locally. In the DAKOTA Toolkit (Eldred et al., 2001) Giunta and Eldred (2000) implemented a sequential approximate optimization (SAO) framework where second order polynomials were computed by the least squares method based on sampling the high fidelity analyses, smoothing the noise of the simulation. This full second order polynomial approximation can be forced to match the zeroth or first order information at the current design point, as in Chang et al. (1993) and Lewis and Nash (2000).

The quadratic response surface approximation (RSA) used by Pérez et al. (2002), Pérez and Renaud (2000), Pérez et al. (2000), and Rodríguez et al. (1998a, 1998b; 2001) represents a different approach. While the zeroth and first order terms are point information, the second order matrix is computed by sampling a local region around the current design point. The RSA constructed in this way provides zeroth and first order matching at the current design, which is important for convergence purposes. The second order response is influenced by designs sampled across the sampling region, which allows the algorithm to take larger steps during optimization and smooths

(via least squares fitting) the noise in the system. When the algorithm gets closer to the minimum, a reduction of the sampling region assures that the second order information will get closer to the true Hessian (since the least squares fitting essentially reproduces the Taylor series approximation). The second order (Taylor series) information is called the true Hessian, or simply Hessian, matrix  $H$ , approximations to which (via least squares fitting or otherwise) are denoted with a tilde:  $\tilde{H}$ .

The response surface approximation to  $f$  at the current design point  $x^k$  is given by

$$\tilde{f}^k(x) = f(x^k) + \nabla f(x^k)^T \Delta x^k + \frac{1}{2} \Delta x^k \tilde{H}_f^k \Delta x^k, \quad (1)$$

where  $\Delta x^k = x - x^k$ . Note that  $\tilde{H}_f^k$  is based not just on information at the point  $x^k$ , but on information from an experimental design centered at  $x^k$ . Constraints  $g_i$  would be similarly approximated by  $\tilde{g}_i^k$  at  $x^k$ .

### 1.2 The curse of dimensionality

The number of independent coefficients in a Hessian matrix is  $n(n+1)/2$ . This  $\mathcal{O}(n^2)$  number of independent coefficients is a drawback, known as the *curse of dimensionality*, for the application of this technique to large optimization problems. To partially alleviate this concern, the data can be collected in parallel as in Burgee et al. (1996) or Pérez et al. (2002) or a *variable fidelity* sampling can be performed as in Rodríguez et al. (2001) for multidisciplinary problems.

A natural way to reduce the sampling cost is to reduce the number of coefficients to be fitted to construct the Hessian. The simplest approach is to compute only the main diagonal terms and set to zero the value of the off-diagonal terms. However, due to the highly coupled nonlinear nature of engineering and in particular MDO problems, this approach does not take into account important information regarding interaction of the design variables.

Another approach (Balabanov et al., 1999; Giunta et al., 1997; Knill et al., 1999) is to fit the data with a reduced term second order polynomial, using only those interaction terms that are statistically significant. The reduced term structure is ascertained first by fitting low fidelity data with a full quadratic. Then insignificant terms are dropped, and the reduced term polynomial is fit to medium or high fidelity data. This process is reported to work very well in practice, but requires the existence of variable fidelity models of the system.

### 1.3 Adaptive experimental design

Pérez et al. (2002) investigate the use of information already available from the previous approximation to reduce the size of the experimental design while maintaining the quality of the approximation. As a result, the total cost of the optimization is reduced. This is accomplished by carrying over the  $\mathcal{O}(n^2)$  eigenvector information from a previous Hessian.

At the very first iteration, a least squares approximation  $\tilde{H}$  (throughout, tildes are used to denote approximations) to the full  $n \times n$  matrix of second order terms  $H$  is produced using an  $\mathcal{O}(n^2)$  sampling array. The spectral decomposition of this matrix is

$$\tilde{H} = U \tilde{H}_U U^T, \quad (2)$$

where the columns of  $U$  are orthonormal eigenvectors and  $\tilde{H}_U$  is a diagonal matrix of eigenvalues. For the next few iterations, an experimental array of  $\mathcal{O}(n)$  is used to sample the design space.

Before performing the least squares fitting, the eigenvector matrix  $U$  is used as a transformation matrix to rotate the design space to new coordinates

$$x^U = U^T x, \quad (3)$$

in which coordinate system the Hessian matrix approximation  $\tilde{H}_U$  is diagonal. In the transformed space, only the main diagonal terms of the Hessian matrix  $\tilde{H}_U$  are computed, i.e., the quadratic model in the transformed coordinates has no second order cross terms  $x_i^U x_j^U$  ( $i \neq j$ ). Then transforming back to the original coordinate system, a full  $n \times n$  matrix  $\tilde{H}$  is obtained. If the curvature of the function (precisely, the eigensystem of its Hessian matrix) has not changed much during the past iteration, this approximate Hessian  $\tilde{H}$  will be similar to that obtained from a least squares estimate  $H^*$  of the full  $n \times n$  Hessian using  $\mathcal{O}(n^2)$  samples, as the off-diagonal terms of the Hessian  $H^*$  in the transformed design space (coordinates  $x^U$ ) will vanish or at least will be small compared to the main diagonal terms. The transformation matrix  $U$  can be kept as long as the curvature of the function does not change too much. When the curvature changes enough to invalidate the quadratic model, the full  $n \times n$  matrix  $\tilde{H}$  of second order terms has to be updated (re-estimated with  $\mathcal{O}(n^2)$  sampling) and a new transformation matrix can be computed. This approach is called adaptive experimental design (AED). Note that a quadratic model for a particular  $f$  or  $g_i$  may remain valid over many sequential approximate optimization iterations, since those iterations are driven by the most rapidly changing models for  $f$  and the  $g_i$ .

In Pérez et al. (2002) the adaptive experimental design methodology was implemented within the trust-region augmented Lagrangian algorithm of Rodríguez et al. (1998b). The implementation was tested using a suite of MDO test problems. Results show that the AED methodology can be applied to engineering problems, significantly reducing the amount of data (analyses) required to fit a full quadratic function.

Though the results in Pérez et al. (2002) show that the AED methodology results in considerable savings compared to full  $n \times n$  Hessian least squares approximations, a full order reduction is not being accomplished. At the beginning of the optimization and after every several iterations, a full  $n \times n$  Hessian estimation must be performed. As a result, several costly full Hessian  $\mathcal{O}(n^2)$  estimates have to be performed. The present paper significantly extends the AED methodology to achieve a full order reduction in the sampling size per iteration from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$  by avoiding full  $n \times n$  Hessian estimations.

## 2. Extended adaptive experimental design (EAED)

Spectral (eigenvalue) decomposition of the Hessian matrix gives two types of information. The eigenvectors form an orthogonal basis that defines the curvature orientation. The eigenvalues provide the (signed) magnitude of the curvature along the eigenvector directions. If the proper orientation (eigenvectors) is known an update of the main diagonal terms (eigenvalues) in the transformed space ( $x^U$  coordinates) is sufficient to determine the full  $n \times n$  Hessian matrix in the untransformed space ( $x$  coordinates).

Computing the Hessian of a quadratic approximation can then be viewed as two complementary tasks. First, find the orientations of the eigenvectors and second, compute the (signed) magnitudes of the curvature along those orientations. The (signed) magnitudes of the curvature can be approximated by sampling the design space, an  $\mathcal{O}(n)$  task. In order to keep an  $\mathcal{O}(n)$  sampling size per iteration throughout the entire optimization, the eigenvectors have to be approximated by means other than least squares based on  $\mathcal{O}(n^2)$  sampling. Based on the experience in

quasi-Newton methods, a plausible choice is to use first order quasi-Newton Hessian updates to update the eigenvectors.

Assume that at any given point  $x^k$  during the optimization, zeroth ( $c(x^k)$ ) and first ( $\nabla c(x^k)$ ) order information is known, and also that zeroth, first, and second ( $\tilde{H}^{k-1}$ ) order information is available for the last design point  $x^{k-1}$ . An update of the eigenvectors can be accomplished in two steps: first update  $\tilde{H}^{k-1}$  to  $\tilde{H}^k$  by a quasi-Newton formula requiring only the first order information at  $x^{k-1}$  and  $x^k$ , and then compute the spectral decomposition of  $\tilde{H}^k$ . Then the normal AED algorithm can be applied and a full  $n \times n$  Hessian approximation  $\tilde{H}^{k+1}$  obtained at the next iterate  $x^{k+1}$  using only  $\mathcal{O}(n)$  sampling. The resulting algorithm, stated in Algorithm 1, is called *extended adaptive experimental design* (EAED). Note that AED and EAED differ substantially in how  $\tilde{H}^k$  and  $\tilde{H}^{k+1}$  are computed, when there is a curvature change between  $x^{k-1}$  and  $x^k$ . AED would first obtain  $\tilde{H}^k$  by the AED  $\mathcal{O}(n)$  sampling in the transformed space, and then recompute  $\tilde{H}^k$  by least squares estimation using  $\mathcal{O}(n^2)$  samples.  $\tilde{H}^{k+1}$  would then have the same eigenvectors as  $\tilde{H}^k$ . EAED would, in this same context of a curvature change, recompute  $\tilde{H}^k$  by using a quasi-Newton update of  $\tilde{H}^{k-1}$  with only  $\mathcal{O}(n)$  sampling.  $\tilde{H}^{k+1}$  would then have the same eigenvectors as this quasi-Newton update  $\tilde{H}^k$ .

Figure 1 illustrates the difference between a quadratic approximation (least squares full Hessian approximation using  $\mathcal{O}(n^2)$  samples), the AED approximation using  $\mathcal{O}(n)$  samples, and the EAED approximation also using only  $\mathcal{O}(n)$  samples and the SR1 update defined in (7) below, for a simple function

$$f(x_1, x_2) = \frac{1}{2} \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + .1x_1^2x_2$$

approximated after a move from  $x^{k-1} = (0, 0)$  to  $x^k = (1, 1)$ . The sampling points in this example are  $(0,0)$ ,  $(2,0)$ ,  $(2,2)$ ,  $(0,2)$  for the full  $2 \times 2$  Hessian approximation, and the latter three for the AED and EAED approximations. All approximations are centered at the point  $(1,1)$ , using the exact function and gradient information there. Near  $x = (1, 1)$ ,  $f$  and all three approximations are very similar, so for clarity the differences are shown near  $(2,2)$ , a point in the direction of the move (the vector  $(1,1)$ ). Observe the significant difference between the AED and EAED approximations. At  $(1,1)$ , both the true and AED Hessian matrices are positive definite, while the EAED Hessian matrix is indefinite, and overall the AED contours appear more similar to those of  $f$  than the EAED contours. Note, however, that along the direction  $(1,1)$  of the move, the EAED approximation is significantly more accurate than the AED approximation. (This directional accuracy derives from the  $\mathcal{O}(n)$  quasi-Newton SR1 update.)

The entire philosophy of AED is to sample the design space intelligently, concentrating samples in the direction of the SAO movement. Thus Figure 1 clearly shows (note the number and values of the contours in the diagonal  $(1, 1)$  direction) that EAED better supports the goal of AED within SAO than pure AED alone does. Note that since  $f$  and  $g_i$  may not be quadratic and the SAO steps may not be small, the quasi-Newton updated  $\tilde{H}^k$  and its eigenvectors may not accurately approximate the true information at  $x^k$ , nor converge to the true information over a sequence of steps. However, as long as the (quasi-Newton update derived) eigenvectors and the (least squares approximated) eigenvalues of  $\tilde{H}^k$  adequately model the function's second order information in the step directions, SAO progress can be made at only  $\mathcal{O}(n)$  cost per step. Since only relatively few such update steps are taken between full  $\mathcal{O}(n^2)$  Hessian approximations, asymptotic convergence of the eigensystem of  $\tilde{H}^k$  to that of  $H^k$  is irrelevant, as demonstrated by the numerical results in Section 3.

---

**Algorithm 1:** Extended Adaptive Experimental Design algorithm.

---

0. Execute the AED algorithm (Pérez et al. (2002) described in the previous section) unchanged until a point  $x^k$  is reached at which AED would re-estimate the approximate Hessian  $\tilde{H}^k$  using  $\mathcal{O}(n^2)$  samples (from any of  $f$  or the  $g_i$ ). Instead do Steps 1–6 and then resume the AED algorithm.
  1. Compute the updated Hessian  $\tilde{H}_*^k$  as a quasi-Newton update of  $\tilde{H}^{k-1}$  using, e.g., (6) or (7) below.
  2. Spectrally decompose  $\tilde{H}_*^k$  to find its eigenvector matrix  $U^k$  as in (2).
  3. Sample the neighborhood of the current design point using an  $\mathcal{O}(n)$  experimental array to gather the database  $\mathcal{A}$  ( $\mathcal{O}(n)$  points and corresponding function values for any of  $f$ ,  $g_i$  whose Hessians need to be recomputed).
  4. Transform the points in the database  $\mathcal{A}$  using  $U^k$  according to (3).
  5. Compute the diagonal Hessian  $\tilde{H}_U^k$  by least squares approximation in the transformed coordinates.
  6. Transform  $\tilde{H}_U^k$  to the original coordinates to obtain the required full  $n \times n$  Hessian matrix  $\tilde{H}^k$  using (2).
- 

### 2.1 First order Hessian updates

First order Hessian updates have been widely used in optimization for three decades and their behavior and properties are well known and documented (Nocedal and Wright, 1999; Conn et al., 2000). Suffice it to say that such updates do apply to the present context, and produce *effective* approximations for optimization without necessarily producing *accurate* approximations of the true Hessians (Conn et al., 2000). In the present paper two update formulas are being implemented: the BFGS update and the symmetric rank 1 (SR1) update. Both are well known and their use and properties for quasi-Newton methods for nonlinear optimization are well understood (see for example Nocedal and Wright, 1999). The main difference between them is that BFGS generates a first order, rank 2 positive definite update, while SR1 is a first order rank 1 update not guaranteed to be positive definite. BFGS is well suited for unconstrained quasi-Newton methods, where positive definiteness is a requirement for line search type optimization algorithms, so an iterate is guaranteed to exist. On the other hand, SR1 is better for trust region algorithms that do not require positive definiteness in the Hessian approximation (recall that the true Hessian at a constrained minimum point need not be positive semidefinite).

Given the gradients at two design points  $x^k$  and  $x^{k+1}$ , define

$$s^k = x^{k+1} - x^k, \tag{4}$$

$$y^k = \nabla f(x^{k+1}) - \nabla f(x^k). \tag{5}$$

The BFGS Hessian update formula is (omitting all the tildes)

$$H^{k+1} = H^k - \frac{H^k s^k s^{kT} H^k}{s^{kT} H^k s^k} + \frac{y^k y^{kT}}{y^{kT} s^k}, \tag{6}$$

and the SR1 update formula is

$$H^{k+1} = H^k + \frac{(y^k - H^k s^k)(y^k - H^k s^k)^T}{(y^k - H^k s^k)^T s^k}. \tag{7}$$



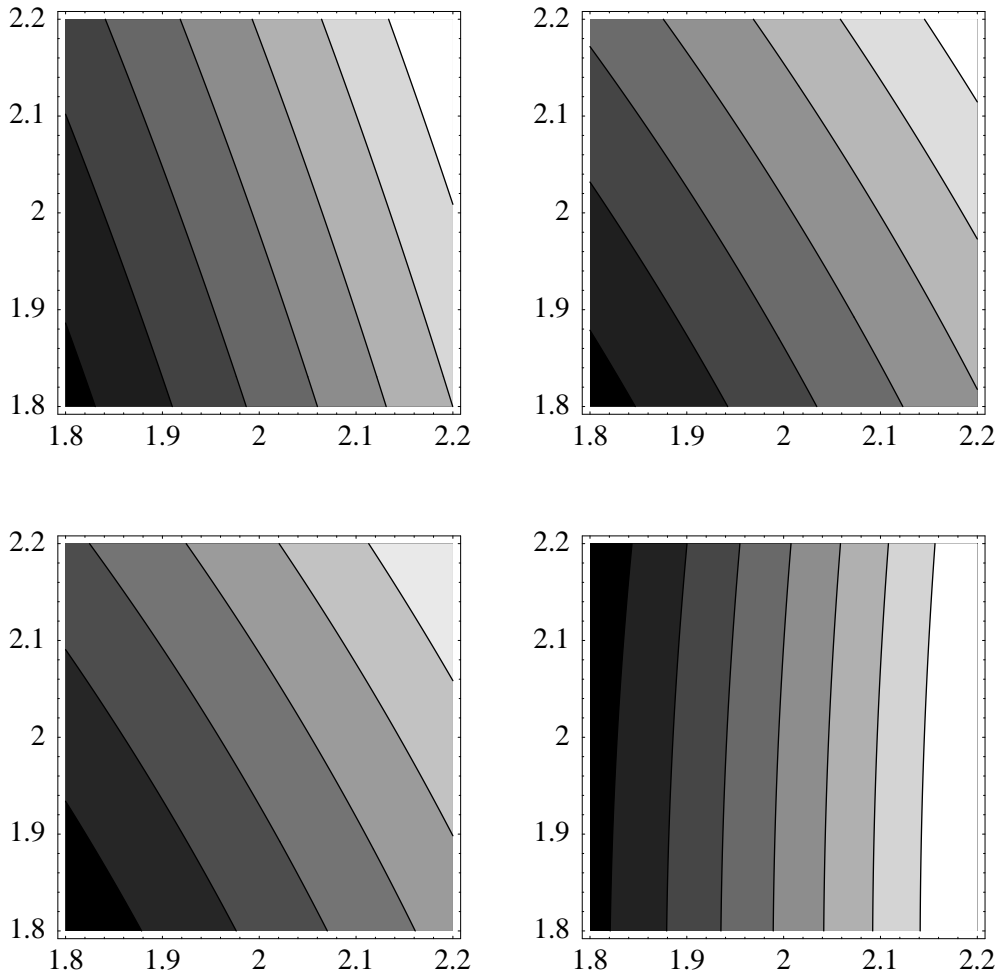


FIGURE 1. Contour plots comparing  $f(x_1, x_2)$  (top left) to full Hessian (least squares quadratic, top right), AED (bottom left), and SR1 based EAED (bottom right) approximations for a move from  $(0,0)$  to  $(1,1)$ . The contour lines and shading are the same for all plots

Note that the SR1 update can be numerically unstable because of the denominator, and hence the update formula (7) must be used carefully in practice. Similarly the BFGS update for indefinite  $H^k$  can be unstable, since  $s^k$  can be an isotropic vector for the indefinite hermitian form  $s^{kT} H^k s^k$  (the term can be zero). Thus both (6) and (7) must be implemented carefully in a trust region context (see for example Nocedal and Wright, 1999).

Both BFGS and SR1 belong to a general class of update formulas known as the Broyden family. The EAED presented in this paper can make use of any Hessian update, since only eigenvector information is obtained by this means. An example of an alternate Hessian update is the Canfield approach (2001), which explicitly takes into account all the previous zeroth and first order information for the update.

## 2.2 EAED algorithm details

In the general case, a response surface approximation of the objective function and each

constraint would be computed. Each function has to be treated individually. A complete discussion of the implementation of the AED when several functions have to be approximated can be found in Pérez et al. (2002). In this section a detailed description of the EAED implementation is provided. The generalities of SAO will only be outlined. For the sake of clarity it is assumed that a bound constrained problem

$$\min_x f(x) \quad \text{subject to} \quad x_{min} \leq x \leq x_{max}, \quad (8)$$

is to be solved, therefore a single function has to be approximated. The generalization to a fully constrained problem is trivial and is demonstrated in the test problems. A description of the EAED implementation in a general SAO framework is shown in Algorithm 2.

---

**Algorithm 2:** EAED Algorithm in a SAO Framework.

---

1. Given  $x^0$ , evaluate  $f(x^0)$  and  $\nabla f(x^0)$ .
  2. Set local move limits.
  3. Set  $k := 0$ .
  4. Using an  $\mathcal{O}(n)$  size experimental array  $A$ , sample the local design space around  $x^k$  and build the database of function values.
  5. If  $k > 0$  go to Step 6, else set  $U^0 := I$  and go to Step 8.
  6. If new curvature information is required (because, e.g., the last step was rejected) compute the updated Hessian matrix  $\tilde{H}_*^k$  according to the desired update formula (6) or (7). Otherwise to to Step 8.
  7. Perform an eigenvalue decomposition of  $\tilde{H}_*^k$  according to (2). The eigenvector matrix  $U^k$  will be used as the transformation matrix.
  8. Transform the experimental array:  $A_U = U^{kT} A$ .
  9. Using least squares fitting, compute the components of the diagonal matrix  $\tilde{H}_U^k$ .
  10. Perform a back transformation to get  $\tilde{H}^k = U^k \tilde{H}_U^k U^{kT}$ .
  11. Perform a minimization on the response surface approximation  $\tilde{f}^k(x)$  subject to the local move limits. The new local optimum point is  $x^{k+1}$ .
  12. Compute  $f(x^{k+1})$  and  $\nabla f(x^{k+1})$ .
  13. Accept or reject  $x^{k+1}$  (by trust region criteria, cf. the SAO details) (Pérez et al., 2002). If accepted, set  $k := k + 1$ .
  14. Update local move limits.
  15. If  $\|x^k - x^{k-1}\| \leq \epsilon$  stop, else go to Step 4.
- 

At the very first iteration, only the main diagonal Hessian terms are approximated by a response surface (an alternative would be to start with a full  $\mathcal{O}(n^2)$  Hessian approximation). At the following iterations, the previous Hessian approximation is updated (when necessary as dictated by curvature changes) using either BFGS or SR1. A spectral decomposition of the updated Hessian is performed to extract the eigenvector matrix. The eigenvector matrix is used to transform the design space, compute the main diagonal Hessian terms via least squares estimation using an  $\mathcal{O}(n)$  database, and then transform back to a full  $n \times n$  Hessian. Call these techniques EAED-BFGS and EAED-SR1, respectively. It is important to underscore the difference between the well known quasi-Newton methods and the trust region framework for approximate optimization that is being implemented in this paper. Traditional quasi-Newton techniques perform an update of the Hessian based on zeroth and first order information at the current and previous design points. The

updated matrix is an approximation of the Hessian at the current design point. In the methodology presented in this paper, zeroth and first order information from the current and previous points are used to update the eigenvectors. The eigenvalues are computed using a least squares regression approach for RSA based on response sampling about the current design. Note that this RSA approach provides a Hessian matrix approximation that is influenced by data sampled over the whole of the sampling region, not just data at the optimizer’s iterates.

### 3. Test problems

To demonstrate the capability of the proposed EAED method to construct response surface approximations for optimization with a reduction in the sampling size to  $\mathcal{O}(n)$  per step, three problems with different characteristics are solved. Two of them are single discipline problems and one is a true MDO problem. In this study, a comparison of the following techniques is performed.

1. Full Hessian (FH). Here all  $(n + 1)n/2$  coefficients of the Hessian matrix are computed at each iteration by least squares approximation. This is the traditional approach requiring a database of size  $\mathcal{O}(n^2)$  to construct the response surface approximation.
2. Extended adaptive experimental design with BFGS Hessian update (EAED-BFGS). The EAED algorithm is used to reduce the number of Hessian coefficients to be fit to  $n$  as in Pérez et al. (2002). The required transformation matrix  $U$  is obtained by a BFGS update of the previous Hessian approximation.
3. Extended adaptive experimental design with SR1 Hessian updates (EAED-SR1). This is the same as EAED-BFGS, but the transformation matrix  $U$  is obtained by a SR1 update of the previous Hessian approximation.

#### 3.1 Barnes problem

This is a small mathematical problem known as the Barnes problem, described in detail in Pérez et al. (2002). It has two design variables and three constraints. The size of the problem allows easy visualization of the results and convergence for some of the Hessian coefficients. The initial point is  $x^0 = (65, 1)$  and the solution is  $x^* = (49.52, 19.62)$ .

The performance results, measured by the number of iterations required to converge to the solution and number of accurate digits given in Pérez et al. (2002), are shown on Table 1. The number of iterations refers to the number of times a response surface approximation was constructed, which translates to  $\lceil(3/2)n\rceil$  SAs or  $\lceil(3/4)(n + 1)n\rceil$  SAs per iteration for EAED or FH, respectively. The most important result is that a reduction in the sample size in the construction of the response surface approximation does not affect the performance of the optimization. EAED-BFGS and EAED-SR1 require almost the same number of iterations to converge as does the full  $n \times n$  Hessian update approach, although they require a factor of  $\mathcal{O}(n)$  fewer samples per iteration. An extra run was performed in which only the main diagonal terms of the Hessian were computed—this corresponds to AED for which  $U = I$  and no  $\mathcal{O}(n^2)$  re-estimations of  $H$  are done during SAO. The increase in the number of iterations to converge for the main diagonal case (AED) can be explained by the lack of the off diagonal terms in the Hessian matrix approximation. The inclusion of the main diagonal technique (AED with  $U = I$ ) in this comparison is with the sole purpose of showing the disadvantages of using such a technique with engineering problems. AED with  $U \neq I$  is clearly better, but then each re-estimation of the full Hessian when the curvature

Approach	Iterations
Full Hessian	15
EAED-BFGS	16
EAED-SR1	15
AED ( $U = I$ )	21

TABLE 1. Performance results for the Barnes problem

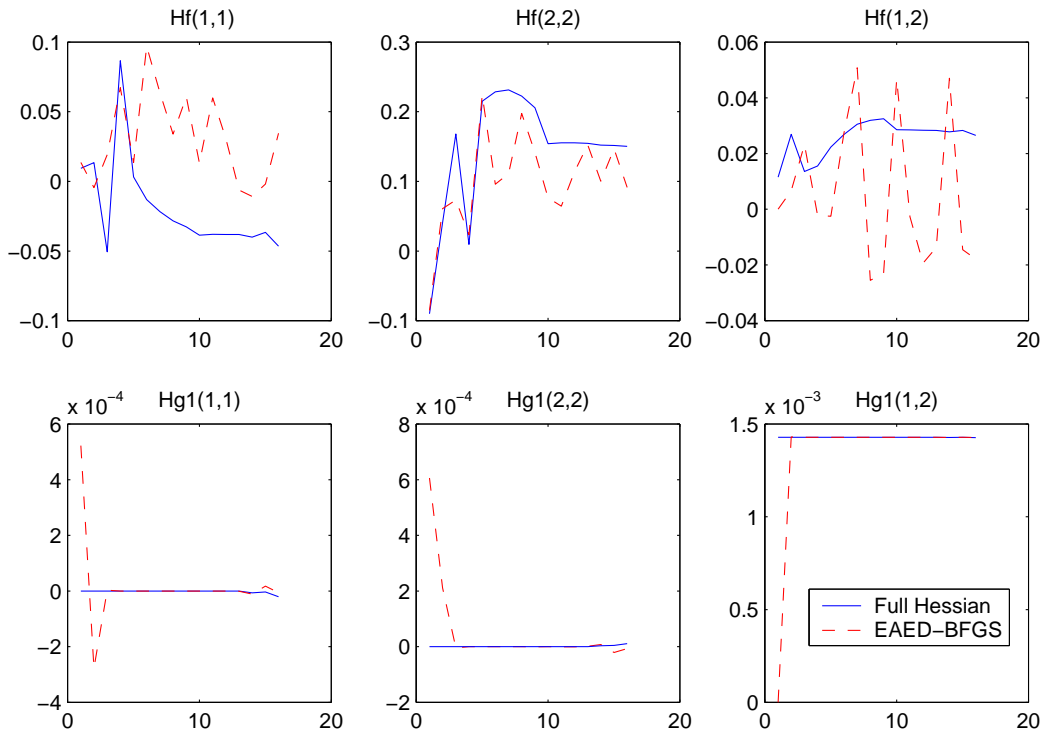


FIGURE 2. Hessian coefficients using EAED-BFGS

changes in AED costs  $\mathcal{O}(n^2)$  samples. In nearly linear problems or those where interactions between the variables are negligible, the main diagonal and AED approaches may compete with the proposed techniques.

Though this is a small problem and no spectacular savings in the number of samplings can be shown, it can provide us with an overview of what happens to the approximate Hessian coefficients as the optimization runs. To this end, the values of the approximate Hessian coefficients for the objective function and the first constraint (with zero-valued diagonal terms) for the methods described above are shown. The values of the coefficients are compared to those from a full Hessian construction (FH).

Figure 2 shows the history for the coefficients of the approximate Hessian matrix using EAED-BFGS. The three plots at the top correspond to the coefficients for the objective function. Note that at the end of the optimization the coefficients of the EAED-BFGS Hessian  $\tilde{H}_{\text{BFGS}}$  do not converge to those of the full Hessian approximation  $\tilde{H}_{\text{FH}}$ . Moreover, the coefficients oscillate and at the end  $(\tilde{H}_{\text{BFGS}})_{1,1}$  and  $(\tilde{H}_{\text{BFGS}})_{1,2}$  even have different signs from their counterparts  $(\tilde{H}_{\text{FH}})_{1,1}$  and  $(\tilde{H}_{\text{FH}})_{1,2}$ . Recall that the BFGS update is positive definite. This is very useful in a line search for unconstrained minimization, but in this constrained optimization case a positive definite update can

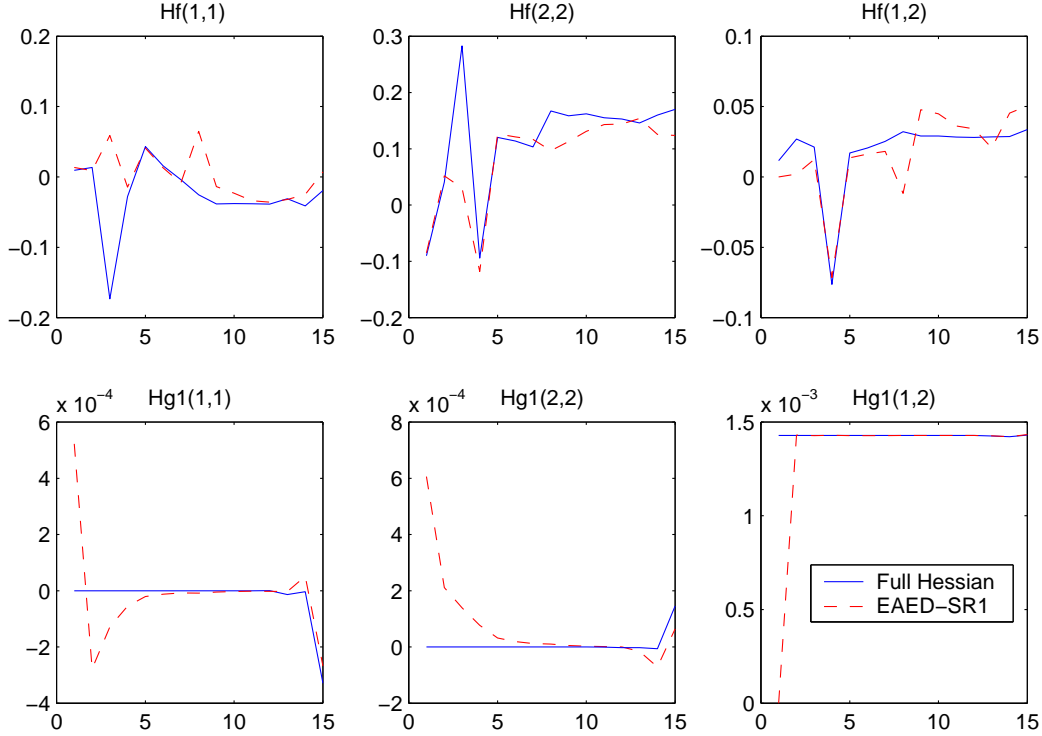


FIGURE 3. Hessian coefficients using EAED-SR1

bias the transformation matrix  $U$ . Interestingly enough, at the end of the optimization the Hessian matrix computed by EAED-BFGS is positive definite while the FH Hessian is not. Regardless, the Hessian approximation  $\tilde{H}_{\text{BFGS}}$  behaves sufficiently like  $\tilde{H}_{\text{FH}}$  in the SAO step directions to result in rapid convergence (this behavior is typical of quasi-Newton methods (Conn et al., 2000)) comparable to that of FH.

The coefficients for the first constraint are shown in the bottom three plots of Figure 2. Note that the main diagonal terms are zero while the off diagonal term has a small value. In this case after three iterations EAED-BFGS is capable of predicting the right values.

In Figure 3 the coefficients' history is plotted for the EAED-SR1 technique. For  $f$ , the EAED-SR1 approximate Hessian  $\tilde{H}_{\text{SR1}}$  follows the FH approximate Hessian  $\tilde{H}_{\text{FH}}$  much better than the EAED-BFGS approximation did. For the case of the constraint, the values are accurate after a couple of iterations. In general, many more samples near the solution point would be required before  $\tilde{H}_{\text{SR1}}$  would very accurately approximate  $\tilde{H}_{\text{FH}}$  at the solution (this is mathematically obvious, since  $\mathcal{O}(n^2)$  independent pieces of information cannot be recovered from only  $\mathcal{O}(n)$  information samples). Note that since Figures 2 and 3 show traces for different updates, the sequences of points visited are different (until the final converged solution), and thus the traces of the full Hessian approximation  $\tilde{H}_{\text{FH}}$  are different until the end.

### 3.2 High performance low cost structure (HPLCS)

This is a structural optimization problem in which the objective of the design is to minimize the weight of the structure while the payloads are sustained at their maximum. The multiobjective optimization is transformed into a single objective optimization via a cost performance index. The problem was introduced in Wujek et al. (1995) and consists of a total of  $n = 17$  design variables

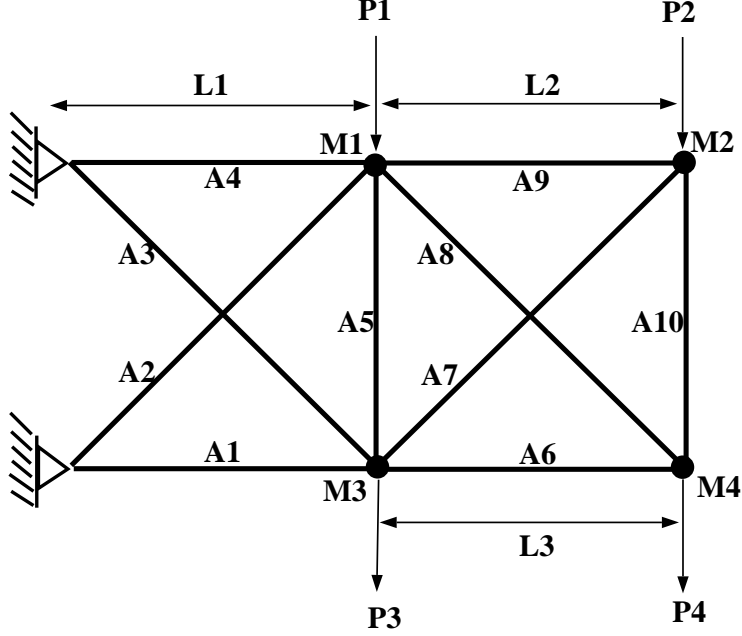


FIGURE 4. High performance low cost structure test problem

Approach	Iterations	
	P1	P2
Full Hessian	39	32
EAED-SR1	51	44
EAED-BFGS	54	45

TABLE 2. Iterations to converge for the HPLCS test problem

(cross sections, truss longitudinals, and payloads) and 13 inequality constraints. Figure 4 shows the structure to be optimized.

In order to compare the EAED algorithm to the conventional full Hessian algorithm (FH), two different experimental arrays have to be used: one that requires  $\mathcal{O}(n^2)$  points for FH and another  $\mathcal{O}(n)$  for EAED. For the HPLCS, the full Hessian approximation requires 153 coefficients to be fitted. An orthogonal array with 162 points (19 variables, 9 levels, strength 2) was used. For the EAED approximation an orthogonal array with 20 points (19 variables, 2 levels, strength 2) was used.

Though the original problem as stated in Wujek et al. (1995) is composed of three disciplines, they are linked by simple feed forward coupling. This allows the treatment of the analysis as a single code, therefore only high fidelity state data is queried.

Both EAED-SR1 and EAED-BFGS are tested and compared against the full Hessian (FH) algorithm. Table 2 presents the results for the number of iterations required to converge from two different starting points, P1 and P2. Compared to the full Hessian approach, the number of iterations required for convergence using EAED is greater. This means that there was a toll for the lower quality of the approximations generated via EAED. However, the real impact can be seen in the total number of function calls as depicted in Figure 5. The cost of the optimization is measured relative to the number of function calls. A function call is defined as either a system analysis (in this case a single code invocation), a gradient evaluation (Grad), or a database query (DB). The number of function calls per iteration is fixed for each of these categories.

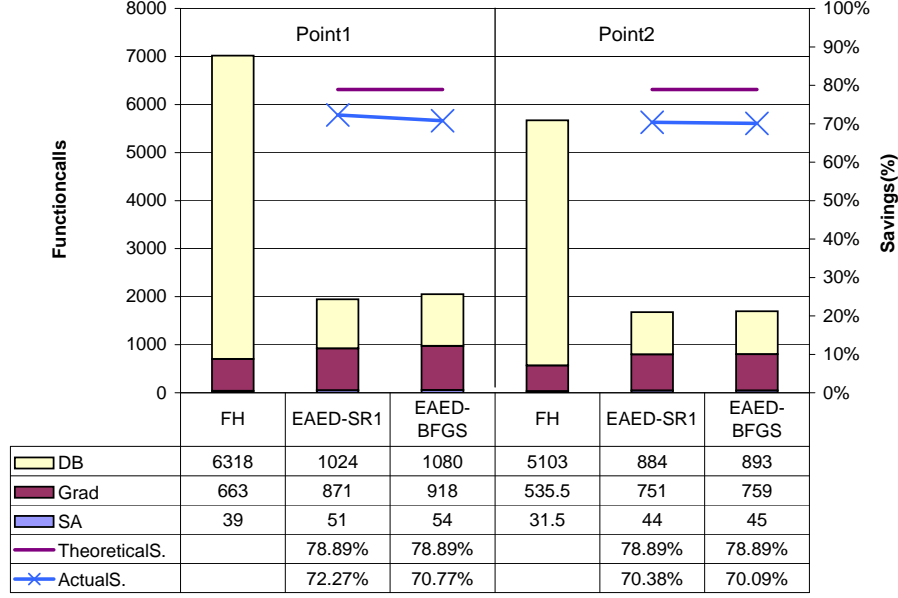


FIGURE 5. Cost in number of function calls and percent savings for the HPLCS problem

In Figure 5 one observes a dramatic decrease from FH to EAED in the number of function calls required for the optimization. This is mainly due to a reduction in the cost of the database query (the Hessian estimation component of the construction of a response surface approximation). Although there is a slight increase for EAED in the number of function calls required for the system analysis and sensitivities evaluation, the number for database generation is significantly smaller than that for FH. A total reduction of around 70% in the cost of the optimization from the FH cost is achieved with both EAED-SR1 and EAED-BFGS. The figure also shows a comparison between theoretical and actual reduction in the cost. The theoretical reduction is the expected savings for a fixed number of iterations (39, 32 for P1, P2) with a fixed function call distribution per iteration. The increase in the number of iterations shown on Table 2 is reflected in the difference between the actual and the theoretical savings.

### 3.3 Controls-augmented structure (CAS)

The controls-augmented structure (CAS) is a fully coupled MDO problem consisting of two subsystems: structures and controls. A cantilever beam is subjected to static and dynamic loads. At the tip there are two controllers, one for the lateral and one for the rotational displacements. The beam is split into five finite elements. The width and height of each element as well as a proportionality constant for the controls comprise the set of design variables. The problem involves a total of 11 design variables and 43 states.

The information flow makes the problem fully coupled and a solution of the system for a given set of design variables can only be obtained by iteration. There are seven inequality constraints bounding the two first natural frequencies, maximum stress, and static and dynamic displacements, both lateral and rotational. Figure 6 shows the initial configuration of the controls-augmented structure, initially introduced by Sobieszczanski-Sobieski et al. (1991) and extensively used in research papers.

The implementation of this problem introduces a new measure for the EAED algorithm. In both the Barnes and the HPLCS problems the data for a response surface approximation was

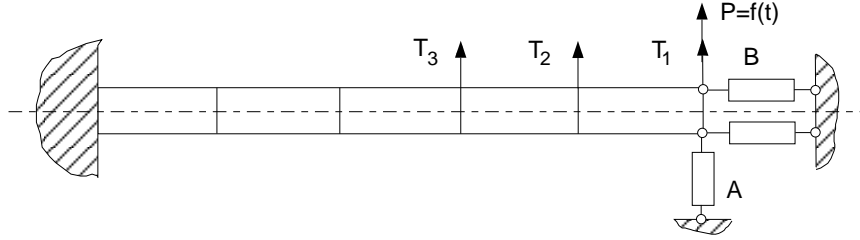


FIGURE 6. *Controls-augmented structure*

Approach	Medium Fidelity		Variable Fidelity	
	P1	P2	P1	P2
Full Hessian	31	33	30	34
EAED-SR1	32	34	38	35
EAED-BFGS	33	33	43	41

TABLE 3. *Iterations to converge for the CAS test problem*

obtained by executing a single analysis code. In the case of the CAS, a single (system analysis) function evaluation has to perform several sequential calls to the independent subdisciplines or contributing analyses (CAs). The cost of a single function call is therefore high and the use of sequential approximate optimization is justified. The data collected is of variable fidelity since the disciplines are linearly decoupled and each design point is evaluated by these decoupled CAs (see Rodríguez et al., 2001). Each point in the experimental design database is evaluated by the decoupled CAs.

As in the case of the HPLCS, the optimization was performed from two different starting points, one in the feasible (P1) and one in the infeasible (P2) region of the design space. For each starting point the optimization was carried out both with data of variable and medium fidelity. The number of iterations measures how many times the database was gathered and the full system analysis performed. Table 3 shows the number of iterations for the four cases described above, two starting points (P1 and P2) for two types of data fidelity. It can be seen that using EAED increases from FH the number of iterations required to converge. This is expected since the amount of data gathered per iteration is smaller for EAED than for FH, and the model is cruder, especially at the beginning, when nothing is known about the orientation of the eigenvectors. However, this does not reflect the true cost of the optimization. Figures 7 and 8 show the cost in number of CA calls for both medium and variable fidelity databases (data for feasible starting point P1 is shown on the left). In the case of the medium fidelity database, the savings of EAED over FH are just below 60%. Both starting points show the same trend. As for the HPLCS problem, no significant difference is shown between the performance of EAED-SR1 and EAED-BFGS. The theoretical savings for this problem are smaller than for the HPLCS problem due to the fact that the system analysis requires a large number of CA calls in contrast to one CA call in the HPLCS case. The variable fidelity results show the same tendency, though in this case the database generation is even cheaper than in the medium fidelity case (Rodríguez et al., 2001), therefore the savings are still less impressive but not insignificant: between 26% and 48%.



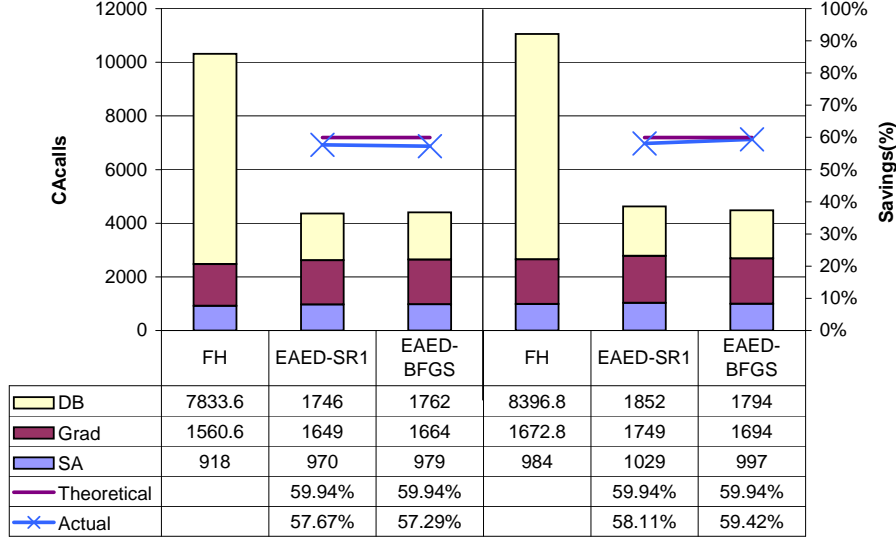


FIGURE 7. Cost in number of CA calls and percent savings for the CAS problem; medium fidelity sampling

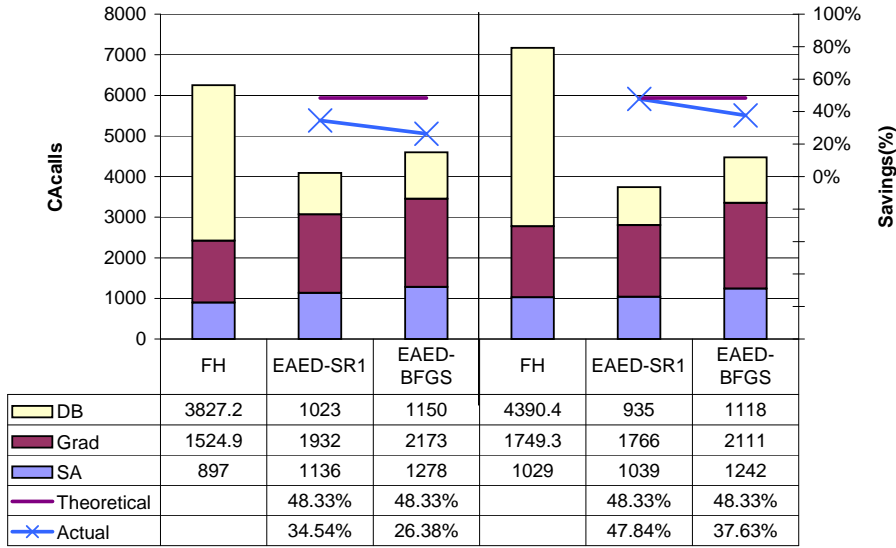


FIGURE 8. Cost in number of CA calls and percent savings for the CAS problem; variable fidelity sampling

#### 4. Concluding remarks

A variant on both AED and EAED not considered here is to use purely quasi-Newton updating between  $\mathcal{O}(n^2)$  Hessian recalculations, without any  $\mathcal{O}(n)$  least squares approximations at all. This is perfectly reasonable for low-noise high-fidelity function values, but not in the assumed context here of noisy function values and variable fidelity databases. Some sort of least squares smoothing,  $\mathcal{O}(n)$  or otherwise, is indispensable. Such noise filtering is, in fact, one of the main justifications for surrogate based optimization, and underlies much of the current work in multidisciplinary design optimization.

The proposed extended adaptive experimental design (EAED) algorithm is a powerful technique for reducing the computational cost of constructing quadratic response surface approximations. The technique reduces the size of the database required per optimization step for constructing

a quadratic response surface approximation in  $n$  dimensions from  $\mathcal{O}(n^2)$  to  $\mathcal{O}(n)$ . The methodology has been shown to have comparable performance (in terms of number of iterations) in sequential approximate optimization to that of constructing full quadratic response surface approximations where  $\mathcal{O}(n^2)$  sampling is used. The reduced order approximation captures the essential information of a full quadratic approximation leading to similar results with smaller sampling size required per iteration. Results for different sizes of test problems have demonstrated scalability of the technique to medium size problems. The EAED algorithm can also be used to build improved Hessian updates for quasi-Newton nonlinear programming methods. In this context, the sampling would be performed in a very small neighborhood of the current design point.

## Acknowledgments

The authors acknowledge the support for this work of the following grants and institutions: NSF grants DMI-9812857, DMI-0114975, DMI-9979711, DMI-0422719, DMI-0355391, and EIA99-74956, NASA grants NAG13-03019 and NAG1-2240, and CONACyT, Mexico.

## References

- Balabanov, V. O., Giunta, A. A., Golovidov, O., Grossman, B., Mason, W. H., Watson, L. T., and Haftka, R. T. (1999), “Reasonable design space approach to response surface approximation”, *Journal of Aircraft*, Vol. 36, No. 1, pp. 308–315.
- Burgee, S., Giunta, A. A., Balabanov, V., Grossman, B., Mason, W. H., Narducci, R., Haftka, R. T., and Watson, L. T. (1996), “A coarse grained parallel variable-complexity multidisciplinary optimization paradigm”, *Internat. J. Supercomputer Applications and High Performance Computing*, Vol. 10, pp. 269–299.
- Canfield, R. A. (2001), “Sequential Multipoint Quadratic Approximation for Numerical Optimization”, *Proc. 42nd AIAA/ASME/ASCE/ AHS/ASC Structures, Structural Dynamics, and Materials Conference*, AIAA 2001-1498, Seattle, WA.
- Chang, K. J., Haftka, R. T., Giles, G. L., and Kao, P. J. (1993), “Sensitivity-based scaling for approximating structural response”, *J. Aircraft*, Vol. 30, pp. 283–288.
- Cheney, W., and Light, W. (2000), *A Course in Approximation Theory*, Brooks/Cole, Pacific Grove, CA.
- Conn, A. R., Gould, N. I. M., and Toint, P. L. (2000), *Trust-Region Methods*, SIAM, Philadelphia.
- Eldred, M. S., Giunta, A. A., van Bloemen Waanders, B. G., Wojkiewicz, S. F., Hart, W. E., and Alleva, M. P. (2001), “DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis. Version 3.0 Users Manual,” Tech. Rep. SAND2001-3796P, Sandia National Labs., Albuquerque, NM.
- Giunta, A. A., Balabanov, V., Haim, D., Grossman, B., Mason, W. H., Watson, L. T., and Haftka, R. T. (1997), “Multidisciplinary optimisation of a supersonic transport using design of experiments theory and response surface modelling”, *Aeronautical Journal*, Vol. 101, pp. 347–356.
- Giunta, A. A. and Eldred, M. S. (2000), “Implementation of a Trust Region Model Management Strategy in the Dakota Optimization Toolkit”, *Proc. 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA0.
- Hosder, S., Watson, L. T., Grossman, B., Mason, W. H., Kim, H., Haftka, R. T., and Cox, S. (2001), “Polynomial response surface approximations for the multidisciplinary design optimization of a high speed civil transport”, *Optimization and Engineering*, Vol. 2, pp. 431–452.
- Knill, D. L., Giunta, A. A., Baker, C. A., Grossman, B., Mason, W. H., Haftka, R. T., and Watson, L. T. (1999), “Response surface models combining linear and Euler aerodynamics for supersonic transport design”, *Journal of Aircraft*, Vol. 36, No. 1, pp. 75–86.
- Lewis, R. M. and Nash, S. G. (2000), “A multigrid approach to the optimization of systems governed by differential equations”, *Proc. 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA 2000-4890, Long Beach, CA.
- Nocedal, J. and Wright, S. J. (1999), *Numerical Optimization*, Springer, New York.
- Pérez, V. M., Apker, T. B., and Renaud, J. E. (2002), “Parallel Processing in Sequential Approximate Optimization”, *Proc. 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, AIAA 2002-1589, Denver, Colorado.
- Pérez, V. M. and Renaud, J. E. (2000), “Decoupling the Design Sampling Region from the Trust Region in Approximate Optimization”, *Proc. International Mechanical Engineering Congress & Exposition*, vol. 63 of AD, Orlando, FL.

- Pérez, V. M., Renaud, J. E., and Gano, S. E. (2000), “Constructing Variable Fidelity Response Surface Approximations in the Usable Feasible Region”, *Proc. 8th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium*, AIAA 2000-4888, Long Beach, CA.
- Pérez, V. M., Renaud, J. E., and Watson, L. T. (2002), “Adaptive Experimental Design for Construction of Response Surface Approximations”, *AIAA Journal*, Vol. 40, No. 12, pp. 2495–2503.
- Rodríguez, J. F., Pérez, V. M., Padmanabhan, D., and Renaud, J. E. (2001), “Sequential Approximate Optimization Using Variable Fidelity Response Surface Approximations”, *Structural and Multidisciplinary Optimization*, Vol. 22, pp. 24–34.
- Rodríguez, J. F., Renaud, J. E., and Watson, L. T. (1998a), “Convergence of Trust Region Augmented Lagrangian Methods Using Variable Fidelity Approximation Data”, *Structural Optimization*, Vol. 15, No. 3-4, pp. 141–156.
- Rodríguez, J. F., Renaud, J. E., and Watson, L. T. (1998b), “Trust Region Augmented Lagrangian Methods for Sequential Response Surface Approximation and Optimization”, *Journal of Mechanical Design*, Vol. 120, No. 1, pp. 58–66.
- Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), “Design and Analysis of Computer Experiments”, *Statistical Science*, Vol. 4, No. 4, pp. 409–435.
- Sobieszczanski-Sobieski, J., Bloebaum, C. L., and Hajela, P. (1991), “Sensitivity of Control-augmented Structure Obtained by a System Decomposition Method”, *AIAA Journal*, Vol. 29, No. 2, pp. 264–270.
- Sobieszczanski-Sobieski, J. and Haftka, R. T. (1997), “Multidisciplinary aerospace design optimization: Survey of recent developments”, *Structural Optimization*, Vol. 14, No. 1, pp. 1–23.
- Welch, W. J. and Sacks, J. (1991), “A System for Quality Improvement via Computer Experiments”, *Communications in Statistics—Theory and Methods*, Vol. 20, pp. 477–495.
- Wujek, B. A., Renaud, J. E., Batill, S. M., and Brockman, J. B. (1995), “Concurrent Subspace Optimization using Design Variable Sharing in a Distributed Design Environment”, *Proc. Design Engineering Technical Conference, Advances in Design Automation, ASME DE*, S. Azarm (ed.), Vol. 82.