Surrogate-Assisted Evolutionary Optimization Frameworks for High-Fidelity Engineering Design Problems

Y. S. Ong[1], P. B. Nair[2], A. J. Keane[2] and K. W. Wong[1]

[1] School of Computer Engineering, Nanyang Technological University, Singapore.
{asysong, askwwong}@ntu.edu.sg
[2] Computational Engineering and Design Group, University of Southampton, Highfield, Southampton, SO17 1BJ, UK.
{P.B.Nair Andy.Keane}@soton.ac.uk

Over the last decade, Evolutionary Algorithms (EAs) have emerged as a powerful paradigm for global optimization of multimodal functions. More recently, there has been significant interest in applying EAs to engineering design problems. However, in many complex engineering design problems where high-fidelity analysis models are used, each function evaluation may require a Computational Structural Mechanics (CSM), Computational Fluid Dynamics (CFD) or Computational Electro-Magnetics (CEM) simulation costing minutes to hours of supercomputer time. Since EAs typically require thousands of function evaluations to locate a near optimal solution, the use of EAs often becomes computationally prohibitive for this class of problems. In this paper, we present frameworks that employ surrogate models for solving computationally expensive optimization problems on a limited computational budget. In particular, the key factors responsible for the success of these frameworks are discussed. Experimental results obtained on benchmark test functions and real-world complex design problems are presented.

1 Introduction

Design of complex engineering systems encompasses a wide range of activities whose goal is to determine the optimum characteristics of a product before it is manufactured. A strong capability to engineer reliable and high quality products is necessary in all engineering design companies to stay competitive in an increasingly global economy, which is constantly exposed to high commercial pressures. Good engineering design know-how results in lower time to market and better quality at lower cost.

In many areas such as aerospace design, the design process has been transformed by the introduction of massive computing power and advances in in-

formation technology, computational sciences and artificial intelligence. These advances are leading to interesting new ways of managing the design process yielding reduction in design cycle times, cost saving and improvements in product quality. Engineering design optimization is an emerging technology whose application both shortens design-cycle time and identifies new designs that are not only feasible, but also increasingly optimal with respect to pre-specified design criteria.

In recent years, Evolutionary Algorithms (EAs) have been applied with a great degree of success to complex design optimization problems [1], [2], [3]. Their popularity lies in their ease of implementation and the ability to locate close the globally optimum designs. However, for many real-life design problems, thousands of calls to the analysis codes may be required to locate a near optimal solution when conventional evolutionary algorithms are employed. A continuing trend in science and engineering is the use of increasingly high-fidelity accurate analysis codes in the design process. For example, modern Computational Structural Mechanics (CSM), Computational Electro-Magnetics (CEM) and Computational Fluid Dynamics (CFD) solvers have been shown to be astonishingly accurate. Such analysis codes play a central role in the design process since they aid designers and scientists in validating designs and also enable them to study the effect of altering key design parameters on product performance. However, moves towards the use of accurate analysis models results in high computational costs in the design optimization process, which consequently leads to longer design cycle times.

In many application areas coupled multidisciplinary system design analysis requiring CSM, CEM or CFD simulations may take up many minutes to hours of supercomputer time. Hence, the overwhelming part of the total run time in such complex engineering design optimization process is taken up by runs of the computationally expensive analysis codes. This poses a serious impediment to the practical application of high-fidelity analysis codes driven by evolutionary design optimization to complex design problems in science and engineering. It is thus desirable to retain the appeal of evolutionary design optimization algorithms that can handle computationally expensive design problems and produce high quality designs under limited computational budgets. Since the design optimization cycle time is directly proportional to the number of calls to the analysis solvers, an intuitive way to reduce the search time of evolutionary optimization algorithms is to replace as often as possible calls to the computationally expensive high-fidelity analysis solvers with lower-fidelity models that are computationally less expensive.

In this chapter, our focus is on surrogate-assisted evolutionary frameworks for solving computationally expensive optimization problems under limited computational budgets. Some key factors responsible for the success of these frameworks are presented. Further, since a nearly linear improvement in design search efficiency may be achieved via straightforward parallelism of population based EAs, all design points within a single EA population should be evaluated simultaneously across multiple compute nodes. Parallelism is thus

considered a desirable feature of any framework for optimization of computationally expensive engineering design problems.

The remainder of this chapter is organized as follows. We begin with a brief overview of surrogate-assisted optimization in complex engineering design. Section 3 presents an overview of surrogate modeling techniques commonly used in the literature. Section 4 presents in greater detail some of the evolutionary frameworks recently proposed for optimization of computationally expensive problems on a limited computational budget using surrogate models. In particular, the key factors responsible for the success of these frameworks are discussed. Experimental results obtained on synthetic functions and real-world complex design problems are also presented. Finally, section 6 summarizes our main conclusions.

2 Surrogate-Assisted Design Optimization

Optimization is a mature technology that has been studied extensively by many researchers over the last decade. Over the years, it has evolved considerably and many algorithms and implementations are now available and used in the engineering optimization community. Optimization algorithms in the literature can be broadly classified into three categories: (1) conventional numerical optimization methods, (2) stochastic optimization methods and (3) hybrid methods. In this section, we present a brief overview of surrogate-assisted optimization strategies. In particular, we consider a general nonlinear programming problem of the form:

$$\begin{aligned} \textbf{Minimize}: \quad & f(\mathbf{x}) \\ \textbf{Subject to}: \quad & g_i(\mathbf{x}) \leq 0, i = 1, 2, \ldots, p \\ & \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u \end{aligned} \tag{1}$$

where $\mathbf{x} \in R^d$ is the vector of design variables, and $\mathbf{x}_l$ and $\mathbf{x}_u$ are vectors of lower and upper bounds, respectively, while p is the number of inequality constraints.

Here, our focus is on cases where the evaluation of $f(\mathbf{x})$ and/or $g(\mathbf{x})$ is computationally expensive, and it is desired to obtain a near optimal solution on a limited computational budget.

2.1 Conventional Numerical Optimization

In conventional numerical optimization methods, it is now standard practice for computationally cheap surrogate models to be used in lieu of exact models to reduce computational cost. Conventional numerical methods commonly used in engineering design include steepest-descent methods, conjugate-gradient, quadratic programming, pattern search methods and linear approximation methods [4], [5], [6].

Gradient-based optimization algorithms make use of line searches to locate a new iterate and hence the issue of range of validity of the approximation models or the control of approximation errors is directly addressed by using *ad hoc* move limits or a trust region framework. As shown by Alexandrov et al. [7], the trust-region strategy for adaptively controlling the move limits guarantees convergence under some mild assumptions on the accuracy of the surrogate model. Other general surrogate-assisted frameworks for managing the use of approximation models in non-gradient based numerical optimization methods algorithms such as pattern search algorithms have also been proposed in the literature; see, for example, Booker et al. [8] and Serafini [9]. One important reason these frameworks have been widely accepted and used is attributed to the theoretical guarantee of convergence to a local optima of the exact problem. Surrogate-assisted conventional numerical optimization methods have been applied with much success to complex engineering design optimization problems, see for example, [8], [9], [10], [11], [12], [13]. A more detailed survey of the state-of-the-art can be found in Simpson et al. [14]

2.2 Evolutionary Optimization

Conventional numerical optimization methods have the known advantage of their efficiency, however, they are very sensitive to the starting point selection and are very likely to stop at non-global optima. The search for algorithms that are capable of escaping from local optima has led to the development of stochastic optimization techniques via the introduction of probabilistic factors in the search process that encourage global exploration. In addition, stochastic techniques, unlike conventional numerical optimization methods, produce new design points that do not use information about the local slope of the objective function and are thus not prone to stalling at local optima. Further, they have shown considerable potential in the solution of optimization problems characterized by non-convex and disjoint or noisy solution spaces. Modern stochastic optimizers which have attracted much attention in recent years include simulated annealing; tabu search; genetic algorithms; evolutionary programming and evolution strategies [15], [16], [17], [18].

These stochastic methods have been successfully applied to mechanical and aerodynamic problems, including turbine blade design [19], multi-disciplinary rotor blade design [20], multi-level aircraft wing design [3], military airframe preliminary design [21] and large flexible space structures design [22]. However, a well-known drawback of EAs in complex engineering design optimization is the need for a large number of calls to the computationally expensive analysis solver in order to locate a near optimal solution. The history of recent developments in conventional numerical optimization methods for engineering design problems indicates that the most influential factor for their widespread use has been the ease with which surrogate models can be incorporated to achieve substantial savings in the computational cost. Hence, the question of how to integrate such approximation models with evolutionary search pro-

cedures needs to be addressed in order to study their practical applicability for design problems, where computational cost is a critical issue. It should be noted that global surrogate models can readily be used with any search method. They can however be inefficient as problem dimension rises. They are also more difficult to set up and tune when compared to local models.

A study of the literature reveals that few studies have addressed the issue of incorporating local approximation models in design procedures based on EAs. One main reason being that since EAs make use of probabilistic recombination operators, controlling the step size of design changes (to control the accuracy of approximate fitness predictions) in any surrogate-assisted evolutionary algorithm will not be as straight-forward as in conventional numerical optimization algorithms. Hence earlier research efforts related to evolutionary optimization have focused on the use of problem specific knowledge to increase the computational efficiency [23],[24]. Even though such problem specific heuristics can be effectively used to achieve performance improvements, there are finite limits to the improvements achievable by such techniques. Robinson and Keane demonstrated the use of variable-fidelity analysis models in EAs for aeronautical design [25]. A computational framework for integrating a class of single-point approximation models with GAs was also proposed in [26]. However, these frameworks are restricted to a special class of approximation models that are domain specific.

For more general surrogate-assisted evolutionary frameworks, several efforts have been made over recent years, particularly using GAs. Ratle [27] examined a strategy for integrating GAs with Kriging models or Design and Analysis of Computer Experiments (DACE) approximations. This work uses a heuristic convergence criterion to determine when an approximate model must be updated. The same problem was revisited by El-Beltagy et al. [28], where it is argued that the issue of balancing the concerns of optimization with those of design of experiments must be addressed. Jin et al. [29] presented a framework for coupling ES and neural network-based surrogate models. This approach uses both the expensive and approximate models throughout the search, with an empirical criterion to decide the frequency at which each model should be used. In Song [30], a real-coded GA was coupled with Kriging in firtree structural optimization.

2.3 Hybrid Evolutionary Optimization

Evolutionary algorithms are capable of exploring and exploiting promising regions of the search space. They can, however, take a relatively long time to locate the exact local optimum in a region of convergence (and may sometimes not find the optimum with sufficient precision). Torn and Zilinskas [31] observe that two competing goals govern the design of global search methods: exploration is important to ensure global reliability; i.e., every part of the domain is searched enough to provide a reliable estimate of the global optimum; however, exploitation is also important since it concentrates the search effort

around the best solutions found so far by searching their neighborhoods to produce better solutions.

Many recent search algorithms achieve these two goals using a combination of dedicated global and local searches. These are commonly known as hybrid methods. Hybrid Evolutionary Algorithm-Local Search methods (EA-LSs), which incorporate local improvement procedures with traditional EAs may thus be used to improve the performance of EAs in search. Such hybrids have been used successfully to solve a wide variety of engineering design problems and experimental studies show that they not only often find better solutions than simple GAs, but also that they may search more efficiently [31], [32], [33], [34], [35]. In diverse contexts, hybrid EA-LSs are also known as Memetic Algorithms. There are two basic strategies for using Memetic Algorithms: Lamarckian learning forces the genotype to reflect the result of improvement by placing the locally improved individual back into the population to compete for reproductive opportunities; and Baldwinian learning, where the improvement procedures are only used to change the fitness landscape, but the solution that is found is not encoded back into the genetic string.

A strategy for coupling ES with local search and quadratic response surface methods was proposed in Liang et al. [36]. However the use of the exact analysis codes to perform local searches results in significantly high computational costs. Further, when working with multimodal high dimensional problems the accuracy of quadratic models may become questionable. A parallel hybrid EA framework that leverages surrogate models for solving computationally expensive design problems with general constraints was proposed by the authors in [1] and further extended in [37] to incorporate gradient information.

3 Surrogate Modeling

Surrogate models or metamodels are (often statistical) models that are built to approximate computationally expensive simulation codes. Surrogate models are orders of magnitude cheaper to run, and can be used in lieu of exact analysis during evolutionary search. Further, the surrogate model may also yield insights into the functional relationship between the input $\mathbf{x}$ and the output y. If the true nature of a computer analysis code is represented as

$$y = f(\mathbf{x}), \tag{2}$$

then a surrogate model is an approximation of the form

$$\hat{y} = \hat{f}(\mathbf{x}), \tag{3}$$

such that $y = \hat{y} + \epsilon$, where ϵ represents the approximation error.

There exist a variety of techniques for constructing surrogate models; see, for example, the texts by Vapnik [38] and Bishop [39] for excellent expositions

of this area. One popular approach in the design optimization literature is least-squares regression using low-order polynomials, also known as response surface methods. A statistically sound alternative for constructing surrogate models of deterministic computer models is Bayesian interpolation, which is sometimes referred to as design and analysis of computer experiments (DACE) modeling in the statistics literature [40], Gaussian process regression in the neural networks literature [41] and Kriging in the geostatistics literature. Artificial neural networks, including Multi-layer Perceptions, Radial Basis Functions (RBF) Networks and multivariate regression splines [42] have also been employed for constructing surrogate models in engineering design optimization. A comprehensive review of different approximation concepts is provided in [43] and a comparison of various techniques can be found in [14], [44], [45].

Here, we will be primarily concerned with approximating deterministic computer models that we assume do not suffer from numerically induced convergence or discretization noise, and hence perfectly interpolating models are most germane to our concerns. Thus we present only a brief overview of interpolation using RBFs, gradient-enhanced RBFs and Bayesian interpolation. Of course, in many real-world applications care must be taken to deal with any numerical noise present in the solution and how this may be dealt with.

3.1 Radial Basis Function Interpolation

Let $D = \{\mathbf{x}^i, y^i, i = 1, 2, \ldots, n\}$ denote the training dataset, where $\mathbf{x} \in R^d$ is the input vector and $y \in R$ is the output. An interpolating RBF model for the dataset D can be written in the form

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(||\mathbf{x} - \mathbf{x}^i||), \tag{4}$$

where $K(||\mathbf{x}-\mathbf{x}^i||) : R^d \rightarrow R$ is a radial basis kernel and $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \in R^n$ denotes the vector of weights.

Typical choices for the kernel include linear splines, cubic splines, multiquadrics, thin-plate splines, and Gaussian functions [39]. The structure of some commonly used radial basis kernels and their parameterization are shown in Table 1. Given a suitable kernel, the weight vector can be computed by solving the linear algebraic system of equations $\mathbf{K}\alpha = \mathbf{y}$, where $\mathbf{y} = \{y^1, y^2, \ldots, y^n\} \in R^n$ denotes the vector of outputs and $\mathbf{K} \in R^{n \times n}$ denotes the Gram matrix formed using the training inputs (i.e., the ijth element of $\mathbf{K}$ is computed as $K(||\mathbf{x}^i - \mathbf{x}^j||)$).

For problems with multiple outputs, for example, problems with multiple objectives and constraints, the weight vector can be efficiently computed for all the outputs of interest once the matrix $\mathbf{K}$ is decomposed.

Table 1. Radial Basis Kernels

Linear Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|$
Thin Plate Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|^k ln\|\|\mathbf{x}-\mathbf{c}_i\|\|^k$
Cubic Splines	$\|\|\mathbf{x}-\mathbf{c}_i\|\|^3$
Gaussian	$e^{-\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i}}$
Multiquadrics	$\sqrt{1+\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i}}$
Inverse Multiquadrics	$(1+\frac{\|\|\mathbf{x}-\mathbf{c}_i\|\|^2}{\beta_i})^{-\frac{1}{2}}$

3.2 Hermite Interpolation Using Radial Basis Functions

In areas such as CFD, it is possible to efficiently compute the sensitivities of the objective and constraint functions using adjoint methods. For such problems, it may be useful to construct gradient-enhanced RBF approximations using the idea of Hermite interpolation. Clearly, this would lead to surrogate models that are more accurate than those which are constructed using function values only.

To illustrate the idea of Hermite interpolation, let us denote the training dataset by $D_g = \{\mathbf{x}^i, y(\mathbf{x}^i, \nabla y(\mathbf{x}^i)\}, i = 1, 2, \ldots, n$, where $\nabla y = \{\partial y/\partial x_1, \partial y/\partial x_2, \ldots, \partial y/\partial x_d\} \in R^d$ denotes the partial derivatives of the output $y(\mathbf{x})$ with respect to the components of the input vector. Then, a Hermite interpolant can be written in terms of a set of RBFs as follows

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(||\mathbf{x}-\mathbf{x}^i||) + \sum_{i=1}^{n}\sum_{j=1}^{d} \tilde{\alpha}_{ij} \frac{\partial K(||\mathbf{x}-\mathbf{x}^i||)}{\partial x_j}, \tag{5}$$

where α_i and $\tilde{\alpha}_{ij}$ $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, d$ are a set of $n(d+1)$ undetermined weights.

It is worth noting here the implicit assumption that the kernel K is differentiable at least twice. Now since the training dataset contains $y(\mathbf{x})$ and $\nabla y(\mathbf{x})$ at n points, we can arrive at a total of $n(d+1)$ linear algebraic equations. This set of equations can be solved to compute the undetermined weights in the Hermite interpolant. It can also be noted that in comparison to the standard RBF approximation approach presented earlier, the Hermite interpolation technique results in a much larger system of equations; for details see [53].

3.3 Bayesian Interpolation and Regression

A statistically rigorous alternative to RBF approximation is the idea of Bayesian interpolation or regression which is also referred to as Gaussian process regression in the neural networks literature and Kriging in the geostatistics literature. The standard starting point for a Bayesian regression

model assumes the presence of an unknown true modeling function $y(\mathbf{x})$ and an additive noise term ν to account for anomalies in the observed data, i.e.,

$$\hat{y}(\mathbf{x}) = y(\mathbf{x}) + \nu \tag{6}$$

The standard analysis requires the specification of a prior probability on the modeling function and the noise model. From a stochastic process viewpoint, the collection $\mathbf{y} = \{y^1, y^2, \ldots, y^n\}$ is called a Gaussian process if any subset of $\mathbf{y}$ has a joint Gaussian distribution. More specifically,

$$P(\mathbf{y}|\mathbf{C}_n, D\}) = \frac{1}{Z}\exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^T\mathbf{C}_n^{-1}(\mathbf{y}-\boldsymbol{\mu})\right) \tag{7}$$

where $\mathbf{C}_n \in \mathbb{R}^{n\times n}$ is a covariance matrix parameterized in terms of hyperparameters $\boldsymbol{\theta}$, i.e., $\mathbf{C}_n(i,j) = K(\mathbf{x}^i, \mathbf{x}^j; \boldsymbol{\theta})$ and $\boldsymbol{\mu}$ is the process mean. The Gaussian process is characterized by this covariance structure since it incorporates prior beliefs both about the true underlying function as well as the noise model. Note that any positive-definite parameterized radial basis kernel K can be employed as a covariance function. Most studies in the literature use the following Gaussian correlation function

$$K(\mathbf{x}^i, \mathbf{x}^j) = \exp\left(-(\mathbf{x}^i - \mathbf{x}^j)^T\boldsymbol{\Theta}(\mathbf{x}^i - \mathbf{x}^j)\right) + \mathcal{N}, \tag{8}$$

where $\boldsymbol{\Theta} = \text{diag}\{\theta_1, \theta_2, \ldots, \theta_d\} \in \mathbb{R}^{d\times d}$ is a diagonal matrix of undetermined hyperparameters. $\mathcal{N}$ is a noise model employed for regression problems; for example, when the noise is assumed to be output dependent, $\mathcal{N} = \theta_{d+1}$, where θ_{d+1} is an additional hyperparameter.

The hyperparameters in the Bayesian surrogate model can be estimated using the evidence maximization framework in which the following maximum likelihood estimation (MLE) problem is solved to determine the most probable hyperparameters $\boldsymbol{\theta}_{MP}$ for the given data.

$$\underset{\boldsymbol{\theta}}{\textbf{Maximize}}\ L(\boldsymbol{\theta}) = -\frac{1}{2}\log\det\mathbf{C}_n - \frac{1}{2}\mathbf{y}_n^T\mathbf{C}_n^{-1}\mathbf{y}_n - \frac{n}{2}\log 2\pi \tag{9}$$

where $L(\boldsymbol{\theta})$ denotes the log likelihood for a Gaussian process.

Since computing $L(\boldsymbol{\theta})$) and its gradients involves inverting a dense $n \times n$ covariance matrix (requiring $\mathcal{O}(n^3)$ resources) at each iteration, MLE of $\boldsymbol{\theta}$ using a gradient based optimizer can be prohibitively expensive even for moderately sized data (e.g., say a few thousand data points). Further, the likelihood function may also be multimodal.

Once the hyperparameters have been computed by solving the MLE problem, predictions can be readily made for a new testing point. To illustrate this, assume that $\mathbf{y}_n$ represents the set of n targets, $\mathbf{C}_n$ the corresponding covariance matrix and that the process to be modeled has zero mean, i.e., $\boldsymbol{\mu} = 0$. Given a new point $\mathbf{x}^{n+1}$, it can be shown that the prediction y^{n+1} has a conditional probability distribution given by :

$$P(y^{n+1}|D, \mathbf{C}_n, \mathbf{x}_{n+1}) = \frac{1}{Z}\exp\left(-\frac{(y^{n+1} - \hat{y}^{n+1})^2)}{2\hat{\sigma}^2}\right) \tag{10}$$

where,

$$\hat{y}^{n+1} = \mathbf{k}_{n+1}^T(\mathbf{x})\mathbf{C}_n^{-1}\mathbf{y}_N \tag{11}$$

$$\sigma^2 = K(\mathbf{x}^{n+1}, \mathbf{x}^{n+1}; \boldsymbol{\theta}) - \mathbf{k}_{n+1}^T(\mathbf{x})\mathbf{C}_n^{-1}\mathbf{k}_{n+1}, \tag{12}$$

where $\hat{y}^{n+1}$ and σ^2 is the prediction for the posterior mean and the variance, respectively, and $\mathbf{k}_{n+1} = \{K(\mathbf{x}_{n+1}, \mathbf{x}_1), K(\mathbf{x}_{n+1}, \mathbf{x}_2), \ldots, K(\mathbf{x}_{n+1}, \mathbf{x}_n)\} \in \mathbb{R}^n$. The posterior variance can be interpreted as an error bar on the predictions made using the Bayesian interpolation model.

4 Surrogate-Assisted Evolutionary Optimization

In spite of the increasing research effort on surrogate-assisted evolutionary optimization frameworks, existing strategies for integrating approximation models with EAs have met with limited success in applications to real-world problems. Some of the key factors responsible for this are:

- The *curse of Dimensionality* results in significant difficulties in constructing accurate surrogate models.
- The lack of massive parallelism in the existing strategies.
- The inability to handle problems with general nonlinear inequality and equality constraints.
- Little emphasis on the global convergence properties of surrogate-assisted evolutionary optimization frameworks.

In this section, we present an overview of some of the recent evolutionary frameworks proposed to address these limitations.

4.1 Surrogate-Assisted Coevolutionary Search

In recent years, coevolutionary computation has been applied with a great degree of success to function optimization, neural network training, and concept learning [46], [47]. Its success lies in the ability to apply divide-and-conquer strategies. For example, in the context of optimization, the variables in the original problem are decomposed into a number of subsets. Subsequently, species that independently handle each subset of variables are evolved simultaneously to locate the optima of the original problem. Since coevolutionary search is based on the divide-and-conquer paradigm, it may be possible to circumvent the *curse of dimensionality* inherent in surrogate modeling techniques.

The RBF surrogate-assisted coevolutionary search procedure proposed in [48] represents an effort to tackle the *curse of dimensionality*, which has limited the success of global surrogate modeling on multimodal problems with

many variables. The steps involved in the proposed surrogate-assisted coevolutionary optimization algorithm are outlined in Figure 1. In the proposed search procedure, standard coevolutionary search proceeds with the initialization of a population of designs for s number of species. The search space of each species is then decomposed into clusters using the standard k-means algorithm and augmented with the elite member as representatives from the other species before fitness evaluations based on the exact analysis model are conducted. Subsequently, evaluation of individuals is based on surrogate models that are constructed on the fly at each ecosystem generation independently and for each species using RBF approximation. The search process within each specie switches back to the exact analysis code when the coevolutionary search on the surrogates stalls. This algorithm has two user specified parameters - (1) the number of species, s, and (2) the number of cluster centers, m. It is found that the accuracy of the surrogate model is improved when m is increased. In the limiting case, when m equals the population size for a species, the fitness of all the individuals is evaluated exactly. Similarly when s is unity the process becomes a traditional evolutionary algorithm.

By dividing the original problem variables among multiple species, the number of inputs and hence the dimensionality of each surrogate model is greatly reduced since each species handles only a subset of the original design variables. However, while this divide-and-conquer approach enables us to tackle the *curse of dimensionality*, a well-known property of coevolutionary search [49] is that high epistatic interactions between the variables can lead to a significant degradation of the convergence rate. In the GA literature, epistasis refers to the variable interdependencies or linkages between the variables of a function. A function of n variables has zero epistasis when there are no interdependencies between the variables.

Studies on the applicability of the RBF surrogate-assisted coevolutionary algorithm to solve computationally expensive optimization problems under limited computational budget were conducted on benchmark test functions in [48] for varying degrees of epistasis. The average convergence trends over 20 runs of the standard Genetic Algorithm (GA), standard Coevolutionary Genetic Algorithm (CGA for s=10) and Surrogate Coevolutionary Genetic Algorithm (SCGA s=10 and m=5) when applied to the low epistasis 20-variable Rastrigin function are summarized in Figure 2. We see that the SCGA converges significantly sooner to near global optimal in comparison to the standard GA and CGA. Further studies on the convergence rate obtained with different m were found to be insignificant. Subsequent investigations conducted on the same Rastrigin function with high epistasis show that in spite of the induced epistasis, the SCGA continues to perform much better than both the GA and CGA on such problems, see Figure 3.

Application of the RBF surrogate-assisted coevolutionary algorithm to the design of a realistic space structure also arrives at a higher quality design than the conventional GA and CGA when a constraint is imposed on the computational budget available for optimization. Figure 4 shows the average

BEGIN
Initialize: Generate population of designs for s species.
Set fitness function:= Surrogate for all species.
While (*computational budget not exhausted*)
For species $i = 1$ to s
Choose representatives from all the other species. The elite member of each species is used.
If (*fitness function == Surrogate*)
· Decompose the design subspace into m cluster centers using the *k-means* algorithm.
· Form collaboration between cluster centers with the representatives from other species and evaluate them using the exact analysis model.
· Build Surrogate based on the m exact points.

For each individual j in population i
Form collaboration between individual j with the representatives from other species.
Evaluate new individual j using Surrogate.
End For
Else
For each individual j in population i
Form collaboration between individual j with the representatives from other species.
Evaluate new individual j using the exact model.
End For
End If
If (*Surrogate Stalls*)
fitness function := Exact Model
Else
fitness function := Surrogate
End If
If (*Implement Elitism*)
Apply standard EA operators to create a new population for species i.
End If
End For
End While
END

Fig. 1. Surrogate Assisted Coevolutionary Optimization

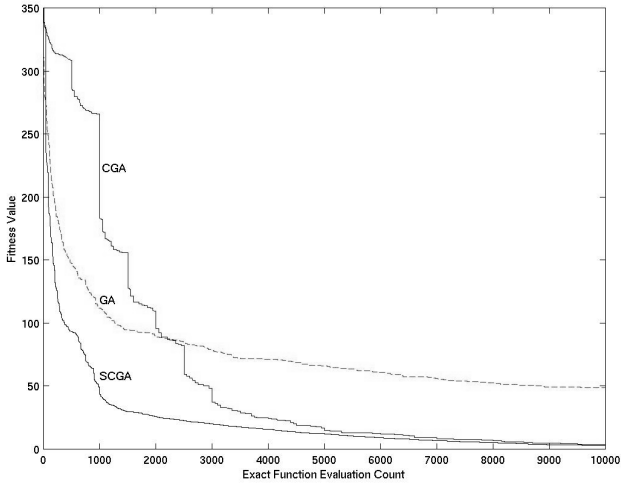

Fig. 2. Convergence trends of GA, CGA with species sizes s=10 and SCGA with s=10 and m=5 when applied on the low epistasis 20-Variable Rastrigin function.

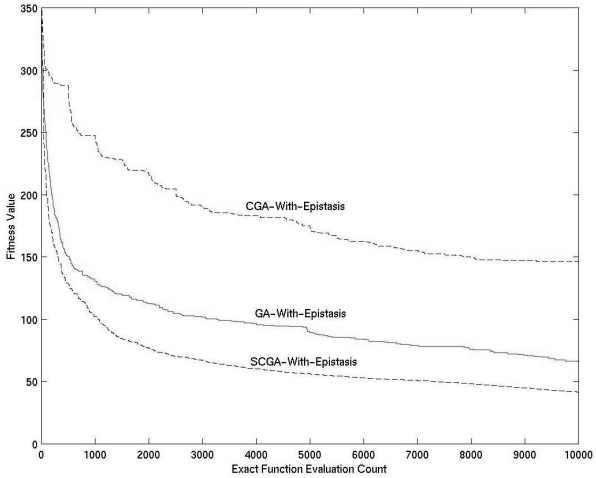

Fig. 3. Convergence trends of GA, CGA with species sizes s=10 and SCGA with s=10 and m=2 when applied on the high epistasis 20-Variable Rastrigin function.

convergence trends of the conventional GA, CGA and SCGA (for s=2 and m=15) as a function of the number of exact analysis.

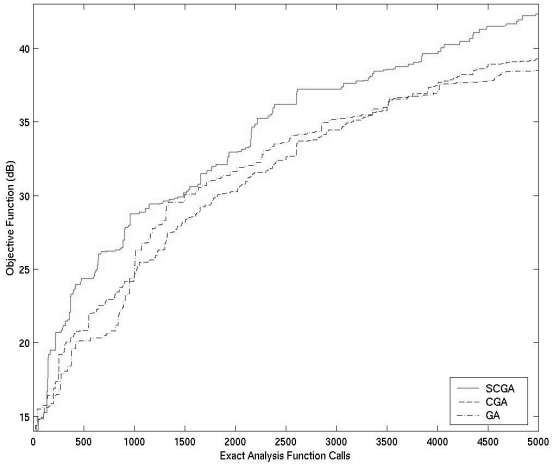

Fig. 4. Convergence trends of GA, CGA and SCGA with s=2 and m=15 when applied to the design of a two-dimensional non-periodic cantilevered space structure to achieve passive vibration suppression.

The space structure considered is a two-dimensional cantilevered structure, subjected to transverse excitation at joint F near the fixed end, see Figure 5. The objective of the design problem is to suppress the vibration response at joint R over the frequency range 100-200Hz. This isolates any instrumentation package mounted at joint R on the space structure from external vibrations arising in the main body of the satellite. The design is parameterized in terms of the coordinates of the structural joints, which are allowed to vary between $\pm$ 0.25 m from the baseline values, with the coordinates of joint R being kept fixed. This leads to a nonlinear multi-modal design problem of 40 geometric design variables with high epistatic linkages among some of the variables. A finite element method is used to compute the free-vibration natural frequencies and mode shapes of the structure. The exact analysis method takes about 100 seconds to compute. In comparison, each RBF approximation takes less than a fraction of a second. The initial space structure and optimized design of the structure using SCGA are illustrated in Figure 5.

4.2 Local Surrogate-Assisted Hybrid Evolutionary Search

Another promising approach to mitigate the effect of the *curse of dimensionality* is the idea of employing local surrogate models that are constructed using design data points that lie in the vicinity of an initial guess [1]. This

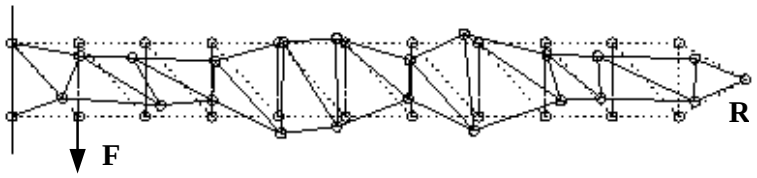

Fig. 5. Initial (represented by dotted lines) and Optimized Shape of the Two-Dimensional Space Structure.

local learning technique may be regarded as an instance of the transductive inference paradigm, which has been the focus of recent research in statistical learning theory [38], [50].

Traditionally, surrogate models are constructed using inductive inference, which involves using a training dataset to estimate a functional dependency and then using the computed model to predict the outputs at the points of interest. However, when constructing surrogate models for optimization, we are specifically interested in ensuring that the models predict the objective and constraint function values accurately at the sequence of iterates generated during the search - how well the model performs at other points in the parameter space is of no concern in this specific context. Transductive inference thus offers an elegant solution to this problem by directly estimating the outputs at the point of interest in one step; the reader is referred to Vapnik's text [38] for a detailed theoretical analysis of its superior generalization capabilities over standard inductive inference.

Surrogate-assisted evolutionary algorithms using local models can be found in [1], [2], [37]. This idea of constructing local models is similar in spirit to the multipoint approximation technique proposed by Toropov et al. [51] and the moving least-squares approximation technique [52].

RBF Local Surrogate-Assisted Hybrid Genetic Algorithm

The essential backbone of the framework proposed in [1] is a parallel evolutionary algorithm coupled with a feasible sequential quadratic programming (SQP) solver in the spirit of Lamarckian learning. Further a trust-region approach is used for interleaving use of the exact models for the objective and constraint functions with computationally cheap surrogate models during local search. The basic steps of the proposed algorithm are outlined in Figure 6. In the proposed algorithm, local surrogate models were constructed using radial basis functions motivated by the principle of transductive inference. Transduction is implemented by constructing radial basis networks using data points in the local neighborhood of an optimization iterate. In other words, instead of constructing global surrogate models, a local model is created on the fly whenever the objective and constraint functions must be estimated at a design point during local search. The localized training data can be readily

selected from a search engine database containing previous iterates, which is continuously updated as the search progresses.

BEGIN
Initialize: Generate a database containing a population of designs.
(*Optional: upload a historical database if one exists*)
While (*computational budget not exhausted*)
Evaluate all individuals in the population using the exact models.
For each non-duplicated individual in the population
· Apply trust-region enabled feasible SQP solver to each individual in the population by interleaving the exact and local surrogate models for the objective and constraint functions.
· Update the database with any new design points generated during the trust-region iterations and their exact objective and constraint function values.
· Replace the individuals in the population with the locally improved solution in the spirit of Lamarckian learning.
End For
Apply standard EA operators to create a new population.
End While
END

Fig. 6. Proposed algorithm for integrating local surrogate models with hybrid EAs for optimization of computationally expensive problems.

Studies on the commonly used Rastrigin benchmark test problem for various sizes of nearest neighbor design point sets (employed to construct the local surrogate model) on the convergence behaviors were conducted and compared with the global surrogate framework proposed by Ratle [27], see Figures 7. The results obtained on commonly used benchmark test functions show that the global surrogate framework displays early sign of stalling. This is consistent with other independent studies in the literature [27], [28], [29] which suggest that when global surrogate models are applied to high-dimensional and multimodal functions, the search generally tends to stall early on. Such an effect is a result of the *curse of dimensionality*, which often leads to early convergence at false global optima of the surrogate model. In contrast, the results obtained using the proposed algorithm clearly demonstrate that solutions close to the global optima can be obtained on a limited computational budget. As surrogates are used only for local searches, i.e., as the exact model is used for all analysis conducted at the EA level, the chances for convergence to false global optima are greatly reduced. In addition, the use of the trust-region framework maintains convergence close to the local optima of the original problem during the SQP steps. Preliminary studies on the number of nearest neighbors design points employed to construct the local surrogate model seems to indicate that using more neighboring points leads to faster convergence during the early stages of search, but has a tendency to stall at

later stages [1]. Hence, a simple strategy for adaptively selecting the number of nearest neighbors during the search was proposed as:

$$m = (m_{min} + m_{max})\frac{t_c}{t_t} \tag{13}$$

where m_{min} is the population size and m_{max} is the maximum number of design points to be used in surrogate modeling. t_c and t_t are the current time spent and the computational budget specified by the user, respectively.

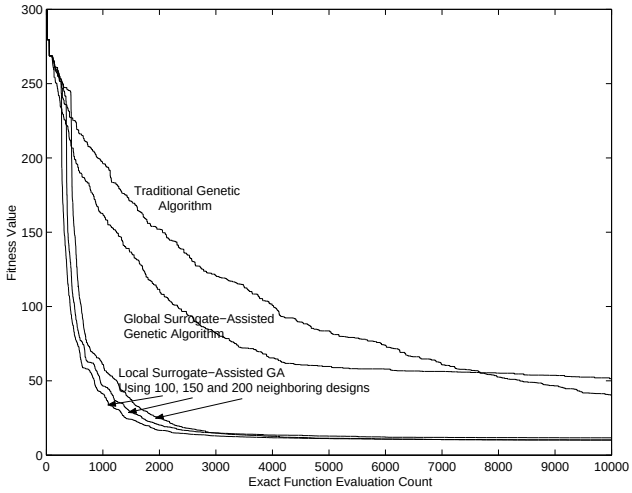

Fig. 7. Averaged convergence trends for various sizes of nearest neighbors design point for construction of local surrogate model; 100, 150 and 200, in comparison with the Traditional GA and Global Surrogate Modeling algorithm in [27] on the 20-Variable Rastrigin function.

Application of a RBF Local Surrogate-Assisted Hybrid GA was further demonstrated in [1] based on a realistic transonic civil transport aircraft wing design problem. The objective of the design problem is minimization of wing D/q as calculated by using the linearized potential code VSAERO, with target lift, wing weight, volume, pitch-up margin and root triangle layout chosen to be representative of a 220 seat wide body airliner. The parameters used to describe the design problem considered consist of the free-stream velocity and coefficient of lift of the wing together with a number of wing geometry variables. The planform geometry is shown in Figure 8 and has 11 design parameters in total. In order to prevent the optimizer from driving the designs to unworkable extremes, four nonlinear inequality constraints are placed on the wings designed.

From these studies, a higher quality design satisfying all the constraints was obtained at a much lower computational budget as compared to the stan-

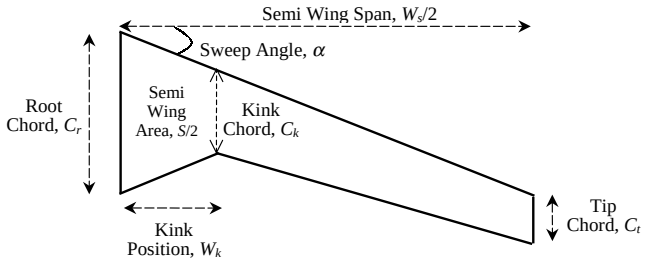

Fig. 8. Transonic civil transport aircraft wing planform geometry.

dard GA. The convergence trends of the best run on the wing problem are presented as a function of wall time in Figure 9.

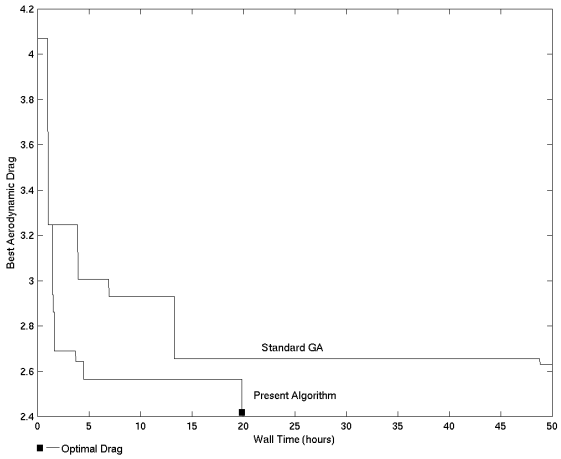

Fig. 9. Optimal convergence trends as a function of wall time for the Aerodynamic Wing Design Problem using the VSAERO code and surrogate models.

These timing plots were based on a total of eight processors being used for parallel computations, due to the availability of only eight licenses for the VSAERO code. VSAERO is a linearized potential code with coupled viscous boundary layer and as employed here, with added correction for compressibility. It is computationally expensive requiring approximately 11 minutes of computational time per drag evaluation. It is worth noting that during local search, surrogate models were constructed for the objective function and the four inequality constraints simultaneously.

Gradient-enhanced Local Surrogate-Assisted Hybrid Genetic Algorithm

In an extension to [1], Ong et al. [53] consider using adjoint CFD solvers and gradient-enhanced Hermite Interpolating RBF for approximation. Since an adjoint CFD solver is used, all the derivatives of the objective and constraint functions at any given design point (including the initial guess) with respect to the design variables can be directly available at a low computational cost. The key idea in [53] is to employ Hermite interpolation techniques to construct gradient-enhanced radial basis function networks so that more accurate surrogate models can be found than those based on function values only.

Numerical studies on airfoil parametric design showed that in comparison to the traditional GA, both the RBF Local Surrogate-Assisted GA and the Gradient-enhanced Hermite Interpolating RBF Local Surrogate-Assisted GA converge to better designs on a limited computational budget [53]. Figure 10 summarizes their design histories on a 2D airfoil design problem.

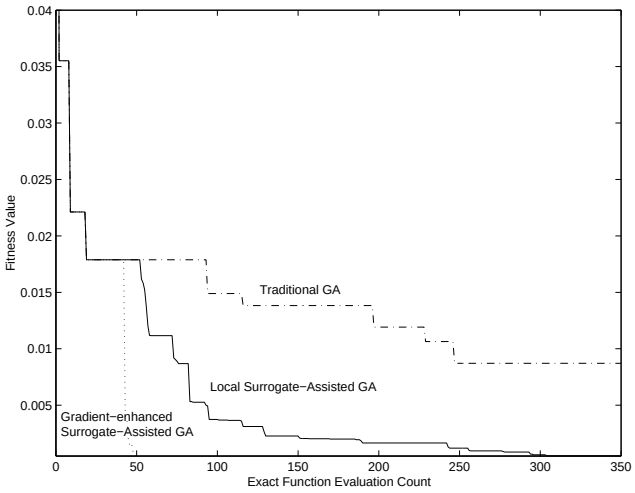

Fig. 10. Convergence trends of traditional GA, Local Surrogate-Assisted GA, Gradient-enhanced Local Surrogate-Assisted GA on 2D airfoil design problem.

The 2D airfoil problem considered (for Mach number 0.5 and 2-degree Angle of Attack (AOA)) is an inverse pressure design problem with 24 design variables and constitutes a good synthetic problem for validating the global convergence guarantee of the proposed algorithm, as the chosen target solution of a NACA 0015 airfoil is known in advance. In the synthetic problem, a single exact Adjoint CFD analysis takes approximately 30 minutes to compute while local surrogate model construction using Gaussian RBF takes a fraction of a

second. Based on the empirical results presented, the search using a traditional GA fails to converge to the target optimal design; see Figures 10 and 11.

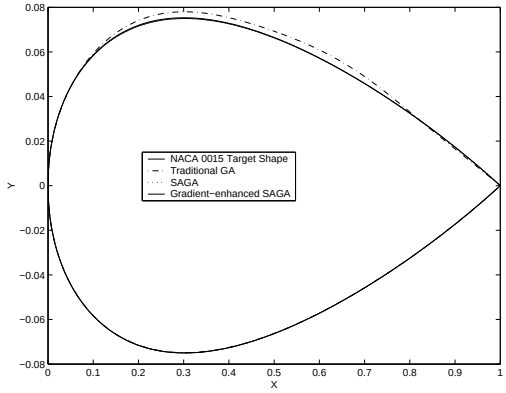

Fig. 11. Comparison of target shapes and final design using the Traditional GA, Local Surrogate-Assisted GA and Gradient-enhanced Local Surrogate-Assisted GA at end of 300, 290 and 50 design cycles, respectively.

The local surrogate-assisted GA converges to the exact NACA 0015 target shape within 290 design cycles, requiring much less computational effort than the traditional GA. On the other hand, the use of exact gradient information in the Hermite Interpolating RBF Gradient-enhanced Local Surrogate-Assisted GA implementation leads to significantly faster convergence, i.e., taking only 50 design cycles. The faster convergence is attributed to the improvement in the accuracy of the local surrogate models.

4.3 Convergence Properties

Global convergence is often defined in the surrogate-assisted optimization literature as the mathematical assurance that the iterates produced by an algorithm, when started from an arbitrary initial guess, will converge to a stationary point or local optima of the original high-fidelity expensive analysis code. It is of theoretical interest to make general mathematical statements about the analytical robustness of any surrogate-assisted evolutionary algorithm in optimization. However, to-date few studies on global convergence guarantees on surrogate-assisted EAs have appeared in the literature. [1] and [37] represent recent efforts to develop EAs that inherit the convergence properties or mathematical robustness of the trust region framework for generalized approximation models.

Alexandrov et al [7] showed that to guarantee global convergence, the following consistency conditions need to be satisfied by the approximation model at the initial guess.

$$\hat{f}(\mathbf{x}_c^k) = f(\mathbf{x}_c^k) \tag{14}$$

$$\nabla\hat{f}(\mathbf{x}_c^k) = \nabla f(\mathbf{x}_c^k) \tag{15}$$

If an interpolating surrogate model is used only the zero-order consistency condition, i.e., Eqn. 14, is satisfied at the initial guess. To satisfy Eqn. 15, the exact sensitivities of the objective and constraint functions are required, which would be computationally prohibitive for many complex design problems. Convergence analysis of trust-region algorithms when only inexact gradient information is available has been considered by Carter [54] and Toint [55]. Leveraging these results, Arian et al. [56] presented a theoretical analysis for unconstrained optimization using surrogates to show that under mild assumptions, convergence can still be guaranteed. In particular, the condition the surrogate model needs to satisfy is that the predicted direction of descent approximates the 'true' direction sufficiently well in the limit. This result can be readily extended to nonlinear programming problems with general constraints by adopting an augmented Lagrangian formulation on the lines of that presented by Rodriguez et al. [57]. In summary, global convergence can be guaranteed only when some assumptions are made regarding the descent direction computed using the surrogate model.

On the other hand, if Hermite interpolants are employed as surrogates during local search [37], both the zero-order and first-order consistency conditions are met. Hence, global convergence can be guaranteed provided sufficient number of iterations are carried out during local search. It is worth pointing out here that these observations on global convergence are of theoretical interest alone since in practical situations the specified computational budget may not allow for a large number of iterations.

5 Conclusions

The study of surrogate-assisted optimization algorithms for tackling computationally expensive high-fidelity engineering design problems is a research area that has attracted much attention in recent years. Much of the earlier work in this area has concentrated on using surrogates, mainly in conventional numerical optimization techniques. In contrast, surrogate-assisted evolutionary optimization is a relatively new research topic that is yet to draw sufficient attention. Nevertheless, an increasing amount of activity in this area is now evident. In this chapter, we have briefly reviewed some of these existing frameworks. Further, we have highlighted some of the key factors that are responsible for the limited success of Surrogate-Assisted evolutionary optimization frameworks on real world applications. An overview of recent frameworks designed to mitigate these problems is also presented. Experimental results obtained on benchmark test functions and real-world complex design problems are presented. These studies indicate that the approaches presented

here allow for the possibility of arriving at near-optimal solutions on limited computational budgets in a range of scenarios.

A well-known strength of evolutionary algorithms is their ability to partition the population of individuals among multiple compute nodes. It would be important for any surrogate-assisted evolutionary framework to retain or further extend the intrinsic parallelism of traditional evolutionary algorithms. Grid Computing [58] has recently been perceived as the enabling technology for collaborative design and the embarrassing parallelism in the evolutionary search [59]. The benefits of Grid computing in the context of evolutionary design optimization are expected to be numerous. Besides the ability to tap into vast compute power, it provide access to almost limitless heterogenous resources. For example, specialized analysis codes, approximation tools and optimization algorithms possessed by different design teams that span across geographically distributed locations may be shared and better utilized. Hence it makes good sense to pursue further research on Grid-enabled Surrogate-Assisted Evolutionary Optimization Frameworks as a cost-effective and computationally tractable solution in high-fidelity complex engineering design.

References

1. Ong Y S, Nair P B, Keane A J (2003) Evolutionary Optimization of Computationally Expensive Problem via Surrogate Modeling. American Institute of Aeronautics and Astronautics Journal, Vol. 41, No. 4, pp. 687-696
2. Giannakoglou K C (2002) Design of Optimal Aerodynamic Shapes using Stochastic Optimization Methods and Computational Intelligence. International Review Journal Progress in Aerospace Sciences, Vol. 38, pp. 43-76
3. Keane A J, Petruzzelli N (2000) Aircraft wing design using GA-based multilevel strategies. Proceedings of the 8th AIAA/USAF/NASSA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA-2000-4937, Long Beach, pp. A00-40171
4. Siddall J N (1982) Optimal Engineering Design: Principles and Applications, Marcel Dekker, Inc., New York
5. Schwefel H P (1995) Evolution and Optimum Seeking, John Wiley&Sons
6. Lawrence C T, Tits A L (1996) Nonlinear Equality Constraints in Feasible Sequential Quadratic Programming. Optimization Methods and Software, Vol. 6, pp. 265-282
7. Alexandrov N, Dennis J E, Jr, Lewis R M, Torczon V (1998) A Trust Region Framework for Managing the use of Approximation Models in Optimization. Structural Optimization, Vol. 15, No. 1, pp. 16-23
8. Booker A J, Dennis J E Jr, Frank P D, Serafini D B, Torczon V, Trosset M W (1998) A Rigorous Framework for Optimization of Expensive Functions by Surrogates. Structural Optimization, 17(1):1-13
9. Serafini D B (1998) A Framework for Managing Models in Nonlinear Optimization of Computationally Expensive Functions. PhD Thesis, Rice University
10. Stelmack M A, Batill S M, Beck B C(2000) Design of an Aircraft Brake Component Using an Interactive Multidisciplinary Design Optimization Framework. ASME Journal of Mechanical Design, Vol. 122, No. 1, pp. 70-76

11. Xu S, Grandhi R V (2000) Ultipoint Approximation Development: Thermal Structural Optimization Case Study. International Journal for Numerical Methods in Engineering, Vol. 48, pp. 1151-1164
12. Canfield R A (2002) Multi-Point Cubic Surrogate Functions for Sequential Approximate Optimization. Third ISSMO/AIAA Internet Conference on Approximations in Optimization
13. Andersson J, Redhe M (2003) Response Surface Methods and Pareto Optimization in Crashworthiness Design. Proceedings of DETC'03 ASME 2003 Design Engineering Technical Conference and Computers and Information in Engineering Conference, Chicago, USA
14. Simpson T W, Booker A J, Ghosh D, Giunta A A, Koch P N, Yang R J (2002) Approximation Methods in Multidisciplinary Analysis and Optimization: A Panel Discussion. Proceedings of the Third ISSMO/AIAA Internet Conference on Approximations in Optimization, pp. 14-25
15. Kirkpatrick S, Gelatt C D, Vecchi M P (1983) Optimization by simulated annealing. Science, Vol. 220, No. 4598, pp. 671-680
16. Back T, Hoffmeister F, Schwefel H (1991) Survey of evolution strategies. editors: Belew R, Booker, Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 2-9
17. Davis L (1991) Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York
18. Michalewicz Z (1994) Genetic Algorithms + Data Structures = Evolutionary Programs. Springer-Verlang, New York, 2nd edition
19. Olhofer M, Arima T, Sonoda T, Sendho B (2000) Optimisation of a stator blade used in a transonic compressor cascade with evolution strategies. Adaptive Computing in Design and Manufacture (ACDM), Springer Verlag, pp. 45-54
20. Hajela P, Lee J (1995) Genetic algorithms in multidisciplinary rotor blade design. In Proceedings of 36th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Material Conference, New Orleans, pp. 2187-2197
21. Parmee I C, Cvetkovi6 D, Watson A H, Bonham Ch R(2000) Multi objective satisfaction within an interactive evolutionary design environment. Evolutionary Computation, Vol. 8, No. 2, pp. 197-222
22. Nair P B, Keane A J (2001) Passive Vibration Suppression of Flexible Space Structures via Optimal Geometric Redesign. AIAA Journal 39(7), pp. 1338-1346
23. Furuya H, Haftka R (1996) Combining genetic and deterministic algorithms for locating actuators on space structures. In Proceedings of AIAA 36th Structures, Structural Dynamics, and Materials Conference and Adaptive Structures Forum
24. Nagendra S, Haftka R T, Gurdal Z, Watson L T (1996) Improved Genetic Algorithms for the Design of Stiffened Composite Panels. Computers and Structures, Vol. 58, No. 3, pp. 543-555.
25. Robinson G M, Keane A J (1998) A Case for Multi-Level Optimisation in Aeronautical Design. In Proceedings of the RAeS Conf. on Multidisciplinary Design and Optimisation, The Royal Aeronautical Society (ISBN 1 85768 074 X)
26. Nair P B, Keane A J, Shimpi R P (1998) Combining Approximation Concepts With Genetic Algorithm-Based Structural Optimization Procedures. Proceedings of the 39th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, CA, AIAA-98-1912, pp. 1741-1751

27. Ratle A (2001) Kriging as a Surrogate Fitness Landscape in Evolutionary Optimization. Artificial Intelligence for Engineering Design Analysis and Manufacturing, Vol. 15, No. 1, pp. 37-49
28. El-Beltagy M A, Nair P B, Keane A J (1999) Metamodelling Techniques For Evolutionary Optimization of Computationally Expensive Problems: Promise and Limitations. Proceeding of the Genetic and Evolutionary Compution Conference (GECCO99)
29. Jin Y, Olhofer M, and Sendho B (2002) A Framework for Evolutionary Optimization with Approximate Fitness Functions. IEEE Transactions on Evolutionary Computation,Vol. 6, No. 5, pp. 481-494
30. Song W B (1998) Shape Optimisation Of Turbine Blade Firtrees. PhD Thesis, University of Southampton
31. Torn A, Zilinskas A (1989) Global Optimization. Lecture Notes in Computer Science, Vol. 350, Springer-Verlag
32. Moscato P (1989) On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms. Tech. Rep. Caltech Concurrent Computation Program, Report. 826, California Institute of Technology, Pasadena, California, USA
33. Houck C, Joines J, Kay M (1996) Utilizing Lamarckian Evolution and the Baldwin Effect in Hybrid Genetic Algorithms. NCSU-IE Technical Report 96-01, Meta-Heuristic Research and Applications Group, Department of Industrial Engineering, North Carolina State University
34. Vicini A, Quagliarella D (1999) Airfoil and wing design using hybrid optimization strategies. AIAA Journal, Vol.37, No.5
35. Ong Y S, Keane A J (2003) Meta-Lamarckian in Memetic Algorithm. IEEE Transactions On Evolutionary Computation, in press, expected to appear in April 2004.
36. Liang K H, Yao X, Newton C (2000) Evolutionary Search of Approximated N dimensional Landscapes. International Journal of Knowledge-Based Intelligent Engineering Systems, Vol. 4, No. 3, pp. 172-183
37. Ong Y S, Lum K Y, Nair P B, Shi D M, Zhang Z K (2003) Global Convergence Unconstrained And Bound Constrained Surrogate-Assisted Evolutionary Search In Aerodynamic Shape Design, Congress on Evolutionary Computation, Special Session on Design Optimisation with Evolutionary Computation, Canberra, Australia, Vol. 3, pp. 1856-1863.
38. Vapnik V (1998) Statistical Learning Theory. John Wiley and Sons, NY
39. Bishop C (1995) Neural Networks for Pattern Recognition. Oxford University Press
40. Sacks J, Welch W J, Mitchell T J, Wynn H P (1989) Design and Analysis of Computer Experiments. Statistical Science, Vol. 4, No. 4, pp. 409-435
41. Williams C K I, Rasmussen C E (1996) Gaussian Processes for Regression. Advances in Neural Information Processing Systems, edited by Touretsky D S, Mozer M C, Hasselmo M E, MIT Press
42. Friedman J H (1991) Multivariate Adaptive Regression Splines. (with discussion). Annals of Statistics
43. Barthelemy J F, Haftka R T (1993) Approximation Concepts for Optimum Structural Design - A Review. Structural Optimization, 5, pp. 129-144
44. Giunta A A, Watson L T (1998) A Comparison of Approximation Modeling Techniques: Polynomial versus Interpolating Models. AIAA 98-4758

45. Jin R, Chen W, Simpson T W (2001) Comparative Studies of Metamodeling Techniques under Multiple Modeling Criteria. Structural and Multidisciplinary Optimization, Vol. 23, No. 1, pp. 1-13
46. Potter M, Jong K D (1994) A cooperative coevolutionary approach to function optimization. In Parallel Problem Solving from Nature-PPSN III, pp. 249-257
47. Potter M, Jong K D (1995) Evolving neural networks with collaborative species. In Proc. of Summer Computer Simulation Conference, pp. 340-345
48. Ong Y S, Keane A J, Nair P B (2002) Surrogate-Assisted Coevolutionary Search. 9th International Conference on Neural Information Processing, Special Session on Trends in Global Optimization, pp. 2195-2199
49. Potter M (1997) The Design and Analysis of a Computational Model of Cooperative Coevolution. Ph.D. thesis, George Mason University, Fairfax, VA
50. Chapelle O, Vapnik V, Weston J (1999) Transductive Inference for Estimating Values of Functions. Advances in Neural Information Processing Systems, Vol. 12, pp. 421-427
51. Toropov V V, Filatov A A, Polynkin A A (1993) Multiparameter Structural Optimization Using FEM and Multipoint Explicit Approximations. Structural Optimization, Vol. 6, No. 1, pp. 7-14
52. Levin D (1998) The Approximation Power of Moving Least-Squares. Mathematics of Computation, Vol. 67, No. 224, pp. 1517-1532
53. Ong Y S, Nair P B, Lum K Y, Zhang Z K (2004) Hybrid Evolutionary Algorithm for Aerodynamic Design Using Hermite Radial Basis Function Interpolants. American Institute of Aeronautics and Astronautics Journal, In communication
54. Carter R G (1991) On the Global Convergence of Trust-Region Algorithms Using Inexact Gradient Information. SIAM Journal of Numerical Analysis, Vol. 28, No. 1, pp. 251-265
55. Toint P L (1988) Global Convergence of a Class of Trust Region Methods for Nonconvex Minimization in Hilbert Space. IMA Journal of Numerical Analysis, Vol. 8, No. 2, pp. 231-252
56. Arian E, Fahl M, Sachs E W (2000) Trust-Region Proper Orthogonal Decomposition for Optimal Flow Control. NASA/CR-2000-210124
57. Rodriguez J F, Renaud J E, Watson L T (1998) Convergence of Trust Region Augmented Lagrangian Methods Using Variable Fidelity Approximation Data. Structural uzzelli, Optimization, Vol. 15, No. 3-4, pp. 141-156
58. Foster, Kesselman C (1999) The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufman Publishers
59. Ho Q T, Ong Y S, Cai W T (2003) "Gridifying" Aerodynamic Design Problem Using GridRPC. Second International Workshop on Grid and Cooperative Computing. Shanghai, China, to appear in Lecture Notes in Computer Science Proceedings List