Struct Multidisc Optim (2016) 54:3–13
DOI 10.1007/s00158-016-1432-3

REVIEW ARTICLE

Parallel surrogate-assisted global optimization with expensive functions – a survey

Raphael T. Haftka[1] · **Diane Villanueva**[1,2] · **Anirban Chaudhuri**[3]

Received: 10 July 2015 / Revised: 8 March 2016 / Accepted: 9 March 2016 / Published online: 2 April 2016

Abstract Surrogate assisted global optimization is gaining popularity. Similarly, modern advances in computing power increasingly rely on parallelization rather than faster processors. This paper examines some of the methods used to take advantage of parallelization in surrogate based global optimization. A key issue focused on in this review is how different algorithms balance exploration and exploitation. Most of the papers surveyed are adaptive samplers that employ Gaussian Process or Kriging surrogates. These allow sophisticated approaches for balancing exploration and exploitation and even allow to develop algorithms with calculable rate of convergence as function of the number of parallel processors. In addition to optimization based on adaptive sampling, surrogate assisted parallel evolutionary algorithms are also surveyed. Beyond a review of the present state of the art, the paper also argues that methods that provide easy parallelization, like multiple parallel runs, or methods that rely on population of designs for diversity deserve more attention.

Keywords Surrogates · Parallel computing · Global optimization

✉ Raphael T. Haftka
haftka@ufl.edu

Diane Villanueva
dvillanu@gmail.com

Anirban Chaudhuri
anirban.chaudhuri01@gmail.com

[1] Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611, USA

[2] Present address: Universal Technology Corporation, Dayton, OH 45432, USA

[3] Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

1 Introduction

Optimization based on computer simulations of complex systems is commonly carried out for the design of engineering systems such as automotive or aerospace vehicles or their components. This paper considers methods for global optimization when simulations are expensive in terms of elapsed time and/or computational cost, so that only limited number of simulations are possible. The throughput of computers is continually increasing, but as Venkataraman and Haftka (2004) observed, much of this progress does not lead to substantial reductions in the time or cost of simulations. Instead, the increased computer throughput is used to improve the fidelity of models and simulations.

Considered here are situations where the time required for completing a single simulation is often 1 day or more, so that even a 1,000 simulations may take too long to execute unless some parallelization is built into the optimization algorithm. Therefore, in this paper we survey only methods that use parallel sampling algorithms. A general distinction between coarse or fine grained parallelization is made based on the computation/communication ratio. As explained by Alba and Troya (1999), if this ratio is high the algorithm is coarse grained, and fine grained if low. Thus, for finer granularity the potential for parallelism is high, but there is more communication, for example, between the multiple threads performing the simulation. This paper limits itself to coarse grained parallelization in the context of the optimization algorithm. That is, we do not consider parallelization of the simulation itself.

The two most obvious techniques of coarse grain parallelization are multiple parallel optimizations and optimization based on population of designs. Multiple parallel optimizations can proceed by dividing the design space into subregions and optimizing in each of these, though some articles do not explicitly parallelize analysis of the expensive functions for each sub-region (Wang et al. (2001), Zhao and Xue (2011), Villanueva et al. (2013). Alternatively, multiple parallel optimizations may be carried out in the entire design space. This was suggested by Le Riche and Haftka (1993) for genetic algorithms and by Schutte et al. (2007) for particle swarm optimization. The latter demonstrated that multiple short runs increase the probability of reaching the global optimum, and in addition provide a better estimate of this probability than a single run.

The algorithms most associated with the use of populations of designs are nature-inspired algorithms (e.g., genetic algorithms, evolutionary algorithms, particle swarm optimization) that are popular for global optimization. For an extensive review of nature-inspired global optimization algorithms the reader is referred to Yang (2010). This paper is mostly limited to genetic and evolutionary algorithms and their variations. These algorithms naturally lend to parallelization due to the evaluation of the fitness function for the large numbers of individuals in a population over many generations. Additionally, as explained by Alba and Troya (1999), genetic algorithms are naturally parallelized since the operations on candidate solutions are relatively independent from each other, and the population can be divided into sub-regions (Gorges-Schleuter 1989, Pettey et al. 1987, Spiessens and Manderick 1991) to localize competitive selection between subsets of candidate solutions.

While parallelization is used to reduce the optimization time, surrogates are often used to increase the power of the optimization, so that the same number of simulations could be used to get closer to the global optimum or to solve more difficult optimization problems. Surrogates (a.k.a metamodels) are simple algebraic models fit to objective functions and to constraints based on their values and possibly their derivatives at one or more points. This paper seeks to survey global optimization methods that combine the use of parallelization and surrogates. Furthermore, we will focus on methods that progressively refine the surrogate. Methods that first execute all the simulations, fit a surrogate, optimize, and quit are only briefly considered.

There are several other devices that are used for global optimization with expensive simulations, which are not surveyed here. These include the combined use of low fidelity and higher fidelity simulations, and methods that seek to reduce the dimensionality of the design space. These devices do not impact much the issue of the potential of the algorithm to combine surrogates with parallelization, and so they are not discussed in this paper.

The objective of the present paper is to survey the potential of various algorithms for parallelization. This area is not mature yet, so we do not dare to draw conclusions into the relative efficiency of the different approaches. Instead, as we believe that the topic is gaining importance, we seek to point out areas where further research has the potential of additional improvements in taking advantage of parallelization. Studies comparing efficiencies are beginning to take place (e.g., Müller and Shoemaker 2014) and we hope that this paper will further facilitate such comparisons.

The remainder of this paper proceeds with a review of one of the most important features of a global optimization algorithm – the ability to provide both exploration and exploitation points to find the global optimum. The following section provides an overview of how this is achieved in surrogate-based algorithms and nature-inspired algorithms. The methods of providing exploration and exploitation points are important as these are methods that drive the ability to parallelize the algorithm. Sections 3 and 4 then describe how researchers have parallelized the algorithms for adaptive sampling algorithms and evolutionary algorithms, respectively. Section 5 concludes this paper with a discussion on guidelines for choosing a method of parallel surrogate based optimization and possible research directions.

2 Exploitation and exploration in global optimization algorithms

Global optimization algorithms typically combine exploitation and exploration. Exploitation involves zooming on regions where previous simulations are close to the current best (feasible or near feasible) objective function value, often called present best solution (PBS). Exploration involves adding points to sparsely sampled areas of the design space, or regions of high uncertainty when considering the prediction uncertainty. This section will detail some features of global optimization algorithms that are intended to promote exploration and exploitation. We will first address algorithms that use surrogate predictions, and then cover some of the basic exploration/exploitation mechanisms of nature-inspired algorithms. In general, most algorithms considered here are used for unconstrained problems, but the extension to constrained problems is also discussed.

First, we note that when it comes to global optimization algorithms that are deterministic, Jones may have had the most profound influence on the field. His first global optimization algorithm DIRECT (for “divided rectangles”, Jones et al. 1993) divides the design space into boxes and subdivides boxes based on a Pareto optimal curve of the contribution to exploration and exploitation. That is, every box is scored on the value of the objective function (its exploitation score) and its size (its exploration score), and all the boxes that are on the

Pareto front of these scores are divided. The algorithm is thus inherently parallelizable (e.g. Watson and Baker 2001), but the number of Pareto optimal boxes at each iteration varies and is unpredictable. This means that its parallelization payoff is more limited. Because it is deterministic, it does not immediately lend itself to multiple short runs. However, this may be accomplished, for example, by small variation in the design box defining the boundary of the design space. To the best of our knowledge, though, this has not been explored.

2.1 Surrogate-based algorithms

2.1.1 Uncertainty-based criteria

While Jones's DIRECT algorithm does not take advantage of patterns in objective function behavior that may be revealed by surrogate fitting, his next algorithm, EGO (for efficient global optimization, Jones et al. 1998) corrected this deficiency by using a Kriging surrogate to fit the data and direct the compromise between exploration and exploitation. Kriging does not only provide an estimate of the objective function everywhere, but also a normal distribution around that value that characterizes the uncertainty. The commonly used version of EGO employs the uncertainty by selecting as the next simulation the point that maximizes the expected improvement (EI) over the present best solution (an idea introduced earlier by Mockus et al. 1978). The exploration part of the algorithm is enhanced by the fact that EI is actually a conditioned expected improvement, conditioned on improvement actually taking place. This condition favors points with large uncertainty to balance the advantage of points with low values of the surrogate (exploitation points). However, EGO is not easily parallelizable, and so requires ingenious methods to parallelize, as will be described below. Figure 1 shows a one-dimensional example of a Kriging surrogate with uncertainty estimates and the EI over the design space. For the example, shown the maximum EI is where the uncertainty is large and away from the data points, thus providing an exploration point.

Another version of EGO suggested by Jones (2001) is to maximize the probability of improvement beyond a given target (initially introduced by Kushner 1964). Ambitious targets promote exploration and modest ones promote exploitation. This version of EGO has not been popular because of the difficulty of choosing a target, but in a parallel environment this difficulty may be mitigated by the use of multiple targets as discussed later in the paper. In fact, since it is more readily parallelizable, it may be the preferred version in a parallel environment.

Booker et al. (1999) sought out a balanced search strategy, with trial points added because of low surrogate predictions, and other points added where the mean square error of the surrogate was large. In this strategy, the trial site with the largest prediction variance was added to the data set, the prediction variance was subsequently set to zero at this point, and the next trial point was examined.

Krause and Ong (2011) analyzed another way of combining the surrogate prediction with its uncertainty estimate, which is to maximize the surrogate prediction minus a multiple of the prediction variance. This method was first mentioned by Jones (2001). Here again it is not clear what multiple would serve as the best compromise between exploration and exploitation. However, again, this difficulty may be mitigated by the use of multiple values in parallel.

Another approach is to use surrogate prediction and its prediction variance to maximize the expected posterior information gain about the global maximizer. This has led to the development of entropy (Villemonteix et al. 2009, Hennig and Schuler 2012) and predictive entropy (Hernández-Lobato 2014) based search strategies.

If we consider exploration and exploitation as two objectives of the optimization, it may be appropriate to treat the generation of compromise samples as a bi-objective optimization problem. Bischl et al. (2014) proposed this approach in their MOI-MBO (multiobjective infill model based optimization) algorithm. This leads to a natural way to parallelize, which is discussed later.

2.1.2 Distance-based and other criteria

EGO requires an uncertainty structure for the surrogate, so it has been mostly applied with Kriging. However, there are methods that fit a surrogate and balance exploitation and exploration without requiring an uncertainty structure. Gutmann (2001) suggested selecting a target value for the global optimum and placing the next simulation at the position that will cause the least "bumpiness" in the surrogate, as illustrated in Fig. 2. Gutmann suggested cycling through different targets for the optimum. Targets that are close to the PBS will favor exploitation, while very low targets will favor exploration.

Regis and Shoemaker (2005) proposed another method that balances exploration and exploitation in radial-basis surrogate based optimization using a more intuitive measure of exploration—distance from existing points. They titled their method CORS-RBF. They optimize the surrogate under the condition that the optimum is at a given minimum distance from any of the previous simulation points. Like Gutmann, they cycle through a set of minimum distances, with small values corresponding to exploitation and large values corresponding to exploration.

Note that all of these methods require global optimization of an inexpensive function such as EI. These functions have large number of local optima, and it is not clear whether the convergence of the overall global optimization is sensitive to the tightness of the convergence of the inner global optimization.

Regis and Shoemaker (2007b) mitigated some of the requirements for inner optimization with their Stochastic Response Surface (SRS) algorithm. This algorithm generates

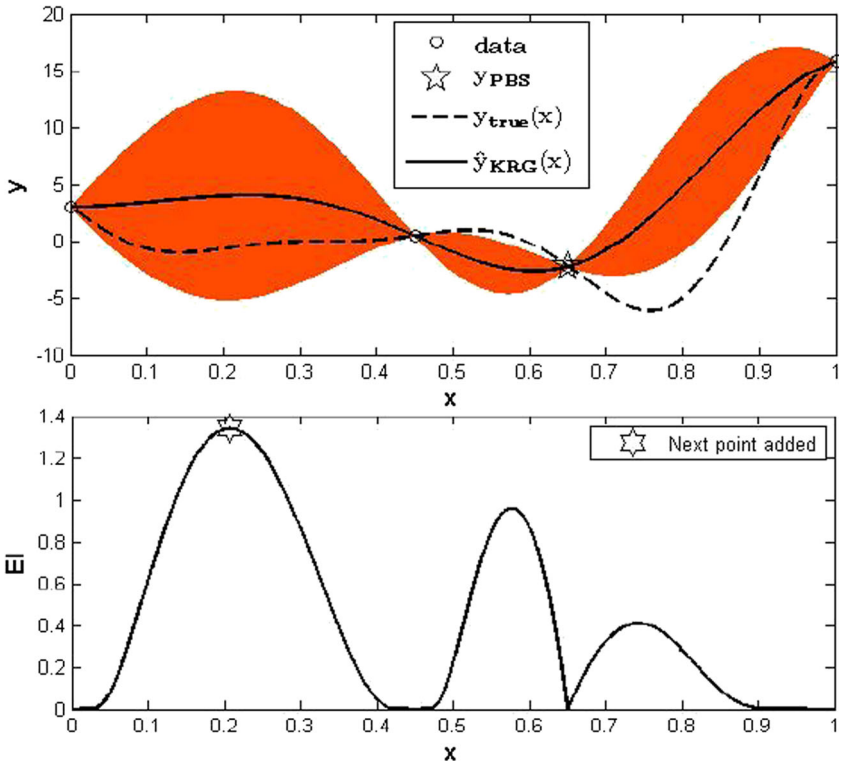

Fig. 1 EGO example showing (*top*) a 1-D function y_{true} approximated by a kriging surrogate, $\boldsymbol{y_{KRG}}$ uncertainty bounds from the kriging surrogate (*in orange*), and present best solution y_{PBS} given four data points; (*bottom*) corresponding EI and maximum EI point

a set of random candidate evaluation points, and one is selected based on a compromise between distance from existing points and the value of the function as predicted by the surrogate. The compromise is governed by weighting the two objectives with weights that cycle from emphasis on the objective to emphasis on the distance.

Hu et al. (2008, 2009) developed a sampling scheme called boundary and best neighbor searching (BBNS) that does not use an uncertainty model. Instead it balances exploration and exploitation by on the one hand looking for neighbors of the best few current samples and on the other hand looking to move towards the boundary of the design space.

Wang et al. (2004) developed the mode-pursuing sampling approach (MPS), which generates a probability function that samples points preferentially where the surrogate has low values, but has a non-zero probability of sampling even when the values are high. The original MPS used a linear spline function as global surrogate with local quadratic surrogates in regions of high point density (promising regions). This approach does not deal well with noisy functions, and it was generalized to deal with noisy function by Wang et al. (2011) by replacing the spline surrogate with a least square support vector regression surrogate.

2.1.3 Problems with constraints

When it comes to adaptive sampling algorithms for constrained optimization, the state of the art is less advanced. Regis and Shoemaker's algorithm is applicable to constrained problems, but EGO and Gutmann's minimum bumpiness algorithms are not, unless combined with penalty function techniques.

There are adaptive sampling algorithms for defining the constraint boundary that is the boundary between the feasible and infeasible domain for that constraint. For example, Ranjan et al. (2008, with corrections in 2011), Bichon et al. (2008), and Picheny et al. (2010), all devised such adaptive sampling

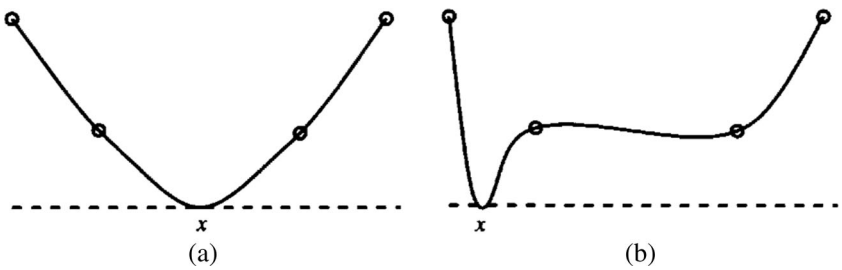

Fig. 2 Fitting a surrogate to four function values, plus a hypothetical minimum value (given by the dashed line). Selecting the position in (a) results in a less bumpy function than the one in (b), from Gutmann (2001)

algorithms based on Kriging and its uncertainty model. Some of these strategies are motivated by reliability calculations rather than optimization, so that the algorithm developed by Bichon et al. (2008) is called EGRA for efficient global reliability analysis. However, these strategies balance exploration and exploitation for one constraint in isolation. That is, the exploitation points are where the constraint is likely to be near zero because the surrogate prediction is small. Exploration points, on the other hand are where the surrogate predictions are not near zero, but the uncertainty is large. For constrained global optimization, one should consider the objective functions and the other constraints when choosing points for improving the surrogate for any single constraints.

When the main challenge of constrained global optimization is feasibility, the superEGO approach (Sasena 2002) may be useful. This approach looks for identifying islands of feasibility by adaptive sampling, followed by local search in each island. Another approach for constrained EGO developed by Basudhar et al. (2012) uses support vector machines for approximating the boundary of the feasible domain.

2.2 Nature-inspired algorithms such as GA and PSO

Population-based algorithms automatically have some exploratory component by virtue of having a population, and the larger the population the larger is this component. In genetic algorithms, exploitation is promoted via selection that is based on fitness and crossover that combines features from parents. Similarly, in particle swarm optimization, exploitation is biasing the motion in the directions of past best results. Relying only on the population for exploration, though, is not considered sufficient. Therefore, mutation operators, as well as randomness in the other operators are added to enhance exploration. The balance between exploration and exploitation is dictated by population size and by the values of probabilities controlling the randomness and mutation type operators.

Further exploration is afforded by diversity enhancing operators, such as niching (Sareni and Krähenbühl (1998), Epitropakis et al. (2011)). This is particularly popular in multi-objective optimization, where good coverage of the Pareto front is needed (Horn et al. (1994)). However, it is used also for single objective optimization. Because of the importance of the balance between exploration and exploitation there has been a number of papers that proposed methods for controlling or tuning that balance.

3 Parallel adaptive sampling algorithms

This section reviews methods that parallelize adaptive sampling surrogate-based global optimization algorithms. First, we examine methods that use a single surrogate to find multiple points per optimization cycle, and move onto how to parallelize via multiple surrogates. This section also covers some algorithms that are parallelized by "zooming" on subregions of the design space. It should be noted that most of the algorithms parallelize the evaluation of the expensive function for multiple points. That is, many seek to add multiple points per optimization cycle, which involves evaluating the expensive cost function for each point in parallel before using all points to refit the surrogate(s).

3.1 Single surrogate

As discussed in the previous section, Jones's EGO algorithm is one of the most popular adaptive sampling surrogate-based global optimization algorithms. Here, we first examine how parallelization has been achieved for two different variants of EGO, followed by other algorithms.

3.1.1 EGO - expected improvement

When a single surrogate (typically Kriging) is used, EGO may be parallelized by looking for multiple good local optima of the expected improvement (EI) as in Sobester et al. (2004). This does not factor in the effect of adding a point on the expected improvement at other points.

When adding several points at once, the question of how to do it optimally is an active area of research. Ginsbourger et al. (2007) tackled first the problem for EGO-EI by brute force. If q samples are desired, it is possible to optimize simultaneously for q points. For that they introduced the multivariate EI (q-EI) and implemented it via Monte Carlo sampling. Since this is very expensive computationally, Ginsbourger et al. (2007, 2010) have suggested two algorithms that deal with this issue. "Kriging Believer" assumes that at the point of maximum EI a simulation will give the value predicted by the surrogate, and then updates the fit and looks for the next point. The process is repeated until the desired number of points is found. The cost of this process is not excessive because as points are added, the Kriging parameters are not re-optimized. Ginsbourger et al. (2007) also proposed an alternate algorithm, dubbed "Constant Liar" that uses a constant value (such as the minimum, mean, or maximum of the function values).

Janusevskis et al. (2012) tackled directly the q-EI problem by Monte Carlo simulation, instead of using the Kriging Believer or Constant Liar alternatives, but that came with considerable cost. Frazier (2012) proposed a q-EI stochastic gradient approach that avoided the explicit calculation of q-EI. Then, Chevalier and Ginsbourger (2013) developed an exact formula for q-EI that is cost effective for modest (up to 10 or so) number of variables.

Viana et al. (2012) compared the use of multiple surrogates, which will be discussed later in the paper, to Kriging Believer for contour estimation and found the performance to be quite similar. Parr et al. (2012) extended Kriging Believer to deal

with constraints. They compared several approaches, and concluded that a multi-objective treatment of the objective and constraints (also known as the filter method) works best. Zhu et al. (2015) extended Kriging Believer to robust design with expensive constraints Li et al. (2016) proposed a method of adding multiple points by combining maximization of EI and minimization of mutual information between the points to be added.

3.1.2 EGO – probability of improvement

The other infill criterion used with EGO, Probability of Improvement (PI), has the capability of adding multiple points per optimization cycle in a much easier way. Jones (2001) pointed out that PI with multiple targets as a highly promising approach and it also overcomes the shortcomings of target setting. The advantage of using PI is that, under the assumption of the local optima being far enough apart, it is easy and cheap to find the joint probability of improvement for all the added points to be used as the objective of the optimization problem. Viana and Haftka (2010) proposed a multi-point PI method which aimed at finding multiple new samples using a joint PI function. Chaudhuri and Haftka (2014) use an adaptive target setting approach to find reasonable target setting. They then implemented multi-point PI by finding the point with maximum PI and then constraining its nearby region by putting a hypersphere around it and running the optimizer to find the next best point far enough apart.

3.1.3 Other gaussian process methods

An important recent development is seeking theoretical convergence rates that quantify the benefit of parallel computation. For Gaussian Process surrogates these can be derived for appropriately structured algorithms. The measure of performance of the algorithm is the regret, defined as the difference or gap between the best result and the actual minimum, or the cumulative regret, which is the sum of the regrets over all the function evaluations. Several papers explore the bounds on the convergence of the regret or the cumulative regret.

Srinivas et al. (2010, 2012) developed an algorithm for maximization of noisy functions that was based on selecting points that maximize the upper confidence bound (UCB) on the maximum, which is the mean plus a multiple of the standard deviation. This balances exploration and exploitation. However, the multiple is adjusted so that bounds on the regret can be obtained, showing that it is a 'no-regret' algorithm (i.e., will almost surely converge to the maximum).

Desautels et al. (2014) extended Srinivas's work as well as earlier work by Krause and Ong (2011) to a parallel algorithm called Batch UCB or GP-BUCB. The parallelized version takes advantage of updating the prediction variance much in the same way as in the Kriging Believer algorithm. With the batch approach there are two multipliers of the standard deviation, and those are again chosen to allow theoretical estimates of the convergence of the regret. The paper proves that when the number of iterations is substantially larger than the batch size K, the regret bounds are reduced by the square root of K.

Contal et al. (2013) developed a similar parallel Gaussian Process algorithm that is based on upper and lower confidence bounds for the maximum (they treated a maximization rather than a minimization problem). Both are based on the mean of the Gaussian process plus or minus a fixed number of standard deviations. The location of the upper confidence bound is one point to be sampled, and the region where the surrogate is above the lower confidence bound defines a "Relevant Region", or the region where there is good probability to find the maximum. All the other points are taken from that relevant region in a pure exploration approach in that region. They too prove that with K parallel simulations, the regret decreases by the square root of K compared to a purely sequential approach.

These papers do not claim to have superior performance to algorithms without proof of rate of convergence. Rather they show by numerical experiments that they have comparable performance with such algorithms, but with the additional advantage of the estimates of the bounds on the regret.

Bischl et al. (2014) formalized the desirability of searching near low predictions of the surrogate (exploitation) and high uncertainty (exploration) by using bi-objective evolutionary algorithm in order to find multiple points on the Pareto front of the two objectives. They also introduced a distance objective in order to obtain points that are well separated.

3.1.4 Methods without uncertainty models

Gutmann's (2001) minimum bumpiness algorithm can be similarly parallelized by simultaneously optimizing for multiple targets. This was implemented by Regis and Shoemaker (2007a) and by Holmstrom (2008). The latter applied Jones approach (2001) for setting multiple targets to the RBF algorithm. Regis and Shoemaker (2007a) also parallelized their own CORS-RBF algorithm by optimizing with a series of maximum distances in parallel. Regis and Shoemaker (2009) also parallelized their SRS algorithm and showed superior performance for a suite of test functions both compared to their parallel CORS-RBF algorithm as well as compared to non-surrogate based algorithms, such as parallel pattern search.

The boundary and best neighbor search (BBNS) method of Hu et al. (2008) was applied in parallel fashion using support vector regression surrogate. The approach appears to be applicable to any other surrogate without using an uncertainty structure. It was applied also by Wang et al. (2010, same Hu Wang of the previous reference, who may have changed the order of his name).

3.2 Multiple surrogates

Fitting multiple surrogates to the same data provides an easy way of adding parallelism to surrogate based optimization. This method is very straight forward in approaches that zoom on promising regions of the design space to restrict the search space, because it does not require an uncertainty model for the surrogates. For example, Zerpa et al. (2005) used multiple surrogates to optimize alkaline-surfactant-polymer flooding processes for oil extraction, combined with the DIRECT global optimizer. Glaz et al. (2009) used four different surrogates as well as two different weighted combinations of these surrogates to perform six optimizations using a genetic algorithm to produce 6 different optima that could be zoomed upon to refine the design.

Besides individual surrogates it is common also to consider a weighted combination of surrogates (e.g., Goel et al. 2007). Müller and Piché (2011) introduced one such combination based on Dempster-Shafer theory, and Müller and Shoemaker (2014) parallelized it by adding random sampling similar to that of Regis and Shoemaker (2007b). Rosales-Perez et al. (2013) have experimented with an ensemble of SVM surrogates each with a different hyperparameters, and found good performance for the NSGA-II evolutionary multi-objective algorithm.

For use with EGO, Viana et al. (2013) suggested using the uncertainty model of Kriging for other surrogates that lack an uncertainty model. This allowed them to use 10 different surrogates and produce up to 10 sampling points for each EGO cycle. The use of multiple surrogates always reduced the number of cycles needed for the optimization, and for one of the test functions (Hartman 6) it even reduced the number of function evaluations compared to the use of only Kriging. This was due to the fact that some of the other surrogates were more accurate than Kriging. A similar approach was used by Viana et al. (2012) for contour estimation, needed to define constraint boundaries for constrained optimization. The approach proved to be comparable to the Kriging Believer approach, as discussed earlier.

Additionally, Chaudhuri et al. (2015) used multiple surrogates (Gaussian process surrogates and polynomial response surfaces), multiple infill criteria (EGO-EI and EGO-PI) and multiple points per EGO-PI for the same experimental data set to add multiple points in a cycle for optimization of flapping wing micro air vehicles for maximum thrust in hover. In this study, parallelization helped take advantage of batch manufacturing and testing of designs. As the multiple surrogates and multiple criteria provided many designs, a distance-based criterion was used to prevent fabrication and testing of designs that were too similar.

3.3 Global–local approaches

A common strategy for exploitation in surrogate-based optimization is to construct a coarse surrogate in the entire design space, and then use it to zoom on promising regions. Then a more refined surrogate may be constructed in the smaller region with a new DOE, possibly utilizing points from the initial DOE that fall in the zoomed region. It is then possible to explore several basins simultaneously, an idea already mentioned in Booker et al. (1999), and van Keulen and Toropov (1999).

For example, Wang et al. (2001, 2003) developed the Adaptive Response Surface Method (ARSM), which disregarded regions with large function values as predicted by the surrogate, and built a new DOE using central composite design or Latin Hypercube sampling (LHS) in the reduced region. The mode-pursuing sampling method (Wang et al. 2004) creates local quadratic surrogates for promising regions, where the sampling approach tends to generate dense samples. Hu et al. (2008) pursued this idea further, using particle swarm optimization to refine the sampling. Peri and Tinti (2012) used a global surrogate of the objective function combined with second order approximations at each DOE (based on the surrogate) point to find an approximate local optimum near the starting point. Sun et al. (2015) combined global and local surrogates for particle swarm optimization, as discussed in the next section.

4 Parallel surrogate-assisted evolutionary algorithms

Most evolutionary optimization methods, such as genetic algorithms or particle swarm optimization are population based algorithms, and as such have built-in parallelism that has been exploited by many (e.g., Schutte et al. 2004). There are many ways the surrogates may be used along with exact function evaluations and Jin (2005 and 2011) provides a good survey of different methods. The idea of using surrogates to improve the efficiency of evolutionary methods continues with recent interest in application to particle swarm optimization (e.g., Parno et al. 2012; Regis 2014; Sun et al. 2015) and evolutionary programming (Regis 2015).

Though Jin's paper on surrogate-assisted evolutionary algorithms as well as more recent papers on particle swarm optimization do not specifically mention parallelization, as many have pointed out, the mechanisms behind nature-inspired algorithms easily lend themselves to parallelization. Alba (Alba and Tomassini 2002; Alba and Troya 1999) provides a survey of parallelization techniques for genetic algorithms and provides a vision for future efforts to parallelize GAs. It mainly discusses restructuring or dividing the search space (e.g., different islands or neighborhoods) without the use of surrogates.

4.1 Global surrogates

With a generational approach, a simple strategy is to have some generations evaluated by surrogates (e.g., Harrison et al. 1995). However, Syberfeldt et al. (2008) point out that generational evolutionary algorithms have a disadvantage with respect to parallelization compared to steady state evolutionary algorithms. First the number of processors may not be a good match for the ideal population size, and also time may be wasted waiting for individuals with slow simulations. They propose instead a steady state parallel surrogate based multi-objective optimization. The algorithm generates large number of child design from a given set of parents and uses the surrogate to select candidate among them. However, the errors manifest between the surrogate and the exact simulation for the parents is used to adjust the surrogate prediction for the child designs.

Similarly, Asouti et al. (2009) developed a steady state evolutionary algorithm that generates local RBF metamodels once a user defined number of exact evaluations are completed. Then as in Syberfeldt, multiple candidates are generated from the metamodels and the top one is chosen for exact evaluation. The parallelization is carried out with grid-assisted asynchronous approach, and the method is applied to aerodynamic shape optimization.

There has been also investigation of which surrogate are most suitable for evolutionary algorithms. Dıaz-Manrıquez et al. (2011) compared quadratic polynomials, kriging, RBF, and support vector regression (SVR) for a Differential Evolution optimizer. They found that for low dimensional problems kriging and SVR performed best, while for high dimensional problems RBF was best. Akhtar and Shoemaker (2015) proposed an RBF-assisted multi-objective evolutionary algorithm framework for achieving a balance between exploration and exploitation through different metrics as objectives, which can all be computed in parallel.

4.2 Local searches

Some use surrogates for improving the local search rather than the global search. Kogiso et al. (1994) created derivative-based local approximation near each member of the population for accelerating the convergence of genetic algorithms. The class of algorithms that combine evolutionary algorithms with local search are now known as memetic algorithms (Hart et al. 2005, Zhou et al. 2007a, b). Sun et al. (2015) combine a global surrogate with local surrogates near particles for particle swarm optimization.

Multiple surrogates are also used to distribute the search space to conduct multiple local searches. Ong et al. (2003) used local surrogates to perform local search at trust regions around individuals in the population. Parallelization is straightforward as the many local searches are performed in parallel. Zhou et al. (2007a, b) developed the Multiple-Surrogate Assisted Memetic Algorithm, in which multiple local surrogates are constructed to perform many local searches in parallel. Shao and Krishnamurthy (2008) proposed the Clustering-Based Multi-Location Search algorithm, using a genetic algorithm using surrogate predictions to find clusters with potentially local optimal points. As mentioned previously, Glaz et al. (2009) used multiple surrogates to find different optima that could be zoomed upon to refine the design.

5 Concluding remarks and guidelines

The field of parallel surrogate based global optimization is relatively new, and we do not feel that it has reached the point where one can draw definite conclusions on the relative merits of the different options. Yet, the reader of this paper may need to make decisions on choosing a method, and so we provide below guidelines, with the warning that they probably reflect our biases, so that we tend to favor methods with which we have first-hand experience.

The first of two of questions that we would suggest a user to ask is whether the objective function is differentiable with readily available derivatives. For such differentiable problems, the user may want to ignore all the methods described in this paper and consider instead multi-start local searches, because of their ease of application as well as their advantage in using derivatives. In particular, the use of derivatives permits their application even when the number of design variables is in the thousands, while surrogate based optimization is typically limited to under one hundred design variables. From our experience in reviewing a large number of papers on global optimization, it appears that authors rarely compare their methods to this simple approach even when the problem is differentiable. Many commercial software have built-in multi-start capabilities, and many use this strategy in conjunction with global optimization algorithms that require multiple starts. For example, in Villanueva et al. (2013), one method to locate the global optimum was to use many local searches that locate local optima using MATLAB's built-in "fmincon" function. In order to actually find the global optimum, multiple starts were necessary for multi-modal problems and, at times, had comparable success to other methods presented in the paper. We feel that this should be a standard practice in such problems.

The second question asked by the user is whether they are committed to use a particular surrogate or particular optimization method, or they are open to select one or the other based on parallelization considerations. EGO users can choose between several methods described in the review. We have a strong preference for using multiple EGO flavors and multiple surrogates, but this is based on our own experience. EGO is typically used with kriging (or Gaussian process surrogate),

which is most commonly used with smooth functions. For noisy functions, it is possible to use kriging with a nugget, but our experience with such a surrogate is mixed, so we have also used polynomial response surfaces for noisy data (e.g., Chaudhuri et al. 2015). Similarly, Wang et al. (2011) have proposed least squares support vector regression surrogate for fitting noisy functions with mode-pursuing sampling.

On the other hand, if the user preferences were toward nature-inspired algorithms, they would be directed to the algorithms that use those optimization methods. User preference again dictates whether a global or local approach is used, and the question of a single surrogate vs multiple surrogates. Users of single surrogates that do not have an uncertainty structure are directed to use the distance based refinements. Users of multiple surrogates that do not have uncertainty structure can use the multiplicity of the surrogates, either by itself or in combination with the distance based approaches.

When it comes to computational costs, the main driver for computational cost is often if surrogate(s) are fit locally in subregions of the design space or globally. For the former, this first requires the decision on where to fit the local surrogates based on an initial global surrogate and followed by fitting surrogates in the refined space (e.g., Booker et al. 1999). In this article, it is assumed that the cost of training a surrogate and using it for prediction is negligible compared to the cost of evaluating the objective function without distinction for different surrogate types. However, though it is simple to generate multiple designs in parallel using multiple surrogates, cost can vary with the number and type of surrogates fit over the space. Additionally, the computational cost of evaluating the surrogate prediction can vary. The overhead cost of surrogate fitting and prediction hasn't been considered in any literature to the knowledge of the authors but it may become considerable when the number of samples increases.

When using population based algorithms such as GA or PSO, the available parallelization is important for a decision on whether to carry multiple runs. With very large number of available processors, multiple parallel runs may be more efficient than a single run with very large populations. However, for methods that use surrogates, there may be more of an advantage of a single run versus multiple runs because the accuracy of the surrogate will be better when all the function evaluations are used to construct it. More research may be needed on intermediate approaches, where there is some sharing of function evaluations between surrogates. Additionally, runs may be in parallel but still asynchronous as explored by Asouti et al. (2009). A fundamental question is under what conditions should information be shared and what criteria call for information to be kept private. This is an issue when communication between surrogates or processors is a source of computational overhead, an issue often brought up in cooperative distributed problem solving in computer science (Durfee et al. 1989).

Acknowledgments Part of this work was supported by the U.S. Department of Energy, National Nuclear Security Administration, Advanced Simulation and Computing Program, as a Cooperative Agreement under the Predictive Science Academic Alliance Program (DE-NA0002378).

References

Akhtar T, and Shoemaker CA (2015) Multi objective optimization of computationally expensive multi-modal functions with RBF surrogates and multi-rule selection. J Global Optimization, 1–16

Alba E, Tomassini M (2002) Parallelism and evolutionary algorithms. IEEE Trans Evol Comput 6(5):443–462

Alba E, Troya JM (1999) A survey of parallel distributed genetic algorithms. Complexity 4(4):31–52

Asouti VG, Kampolis IC, Giannakoglou KC (2009) A grid-enabled asynchronous metamodel-assisted evolutionary algorithm for aerodynamic optimization. Genet Program Evolvable Mach 10(4):373–389

Basudhar A, Dribusch C, Lacaze S, Missoum S (2012) Constrained efficient global optimization with support vector machines. Struct Multidiscip Optim 46(2):201–221

Bichon BJ, Eldred MS, Swiler LP, Mahadevan S, McFarland J (2008) Efficient global reliability analysis for nonlinear implicit performance functions. AIAA J 46(10):2459–2468

Bischl B, Wessing S, Bauer N, Friedrichs K, Weihs C et al (2014) MOI-MBO: Multiobjective infill for parallel model-based optimization. In: Pardalos PM (ed) LION 2014, LNCS, vol 8426., pp 173–186. doi:10.1007/978-3-319-09584-4

Booker AJ, Dennis JE, Frank PD, Serafini DB, Trosset MW (1999) A rigorous framework for optimization of expensive functions by surrogates. Struct Multidiscip Optim 17(1):1–13

Chaudhuri A, Haftka RT (2014) Efficient global optimization with adaptive target setting. AIAA J 52(7):1573–1578. doi:10.2514/1.J052930

Chaudhuri A, Haftka RT, Ifju P, Chang K, Tyler C, Schmitz T (2015) Experimental flapping wing optimization and uncertainty quantification with limited samples. Struct Multidiscip Optim 51(4):957–970. doi:10.1007/s00158-014-1184-x

Chevalier C, and Ginsbourger D (2013) Fast computation of the multipoints expected improvement with applications in batch selection, In Learning and Intelligent Optimization, Springer, pp. 59–69

Contal E, Buffoni D, Robicquet A, and Vayatis N (2013) Parallel Gaussian Process Optimization with Upper Confidence Bound and Pure Exploration, Proceedings of Machine Learning and Knowledge Discovery in Databases, European Conference ECML PKDD 2013, Part I, 225–240

Desautels T, Krause A, Burdick JW (2014) Parallelizing exploration-exploitation tradeoffs with gaussian process bandit optimization. J Mach Learn Res 15(1):3873–3923

Dıaz-Manrıquez A, Toscano-Pulido G, and Gomez-Flores W (2011) On the Selection of Surrogate Models in Evolutionary Optimization Algorithms, IEEE Congress on Evolutionary Computation, 2155–2162

Durfee EH, Lesser VR, Corkill DD (1989) Trends in cooperative distributed problem solving. IEEE Trans Knowl Data Eng 1(1):63–83

Epitropakis MG, Plagianakos VP and Vrahatis MN (2011) Finding multiple global optima exploiting differential evolution's niching capability, In IEEE Symposium on Differential Evolution (SDE), pp. 1–8.

Frazier PI (2012) Parallel global optimization using an improved multipoints expected improvement criterion. In INFORMS Optimization Society Conference, Miami

Ginsbourger D, Le Riche R, and Carraro L (2007), A multi-points criterion for deterministic parallel global optimization based on kriging, International Conference on Nonconvex Programming, Rouen, France

Ginsbourger D, Le Riche R and Carraro L (2010) Computational Intelligence in Expensive Optimization Problems, chapter "Kriging is well-suited to parallelize optimization", Studies in Evolutionary Learning and Optimization, Springer-Verlag

Glaz B, Goel T, Liu L, Friedmann PP, Haftka RT (2009) Multiple-surrogate approach to helicopter rotor blade vibration reduction. AIAA J 47(1):271–282

Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of multiple surrogates. Struct Multidiscip Optim 33(3):199–216

Gutmann HM (2001) A radial basis function method for global optimization. J Glob Optim 19(3):201–227

Harrison PN, Le Riche R and Haftka RT (1995) Design of Stiffened Composite Panels by Genetic Algorithm and Response Surface Approximations, AIAA Paper 95–1163, Proceedings, 36th AiAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, April 10–12, New Orleans, LA, Part 1, pp. 58–68

Hart WE, Krasnogor N, Smith JE (2005) Memetic evolutionary algorithms, in recent advances in memetic algorithms. Springer, Berlin, pp 3–27

Hennig P and Schuler C J (2012) Entropy search for information-efficient global optimization. The Journal of Machine Learning Research 13(1): 1809-1837

Hernández-Lobato J M, Hoffman M W and Ghahramani Z (2014) Predictive entropy search for efficient global optimization of black-box functions, In: Advances in Neural Information Processing Systems,pp 918-926

Holmstrom K (2008) An adaptive radial basis algorithm (ARBF) for expensive black-box global optimization. J Glob Optim 41(3): 447–464

Horn J, Nafpliotis N, and Goldberg, DE (1994) A niched Pareto genetic algorithm for multiobjective optimization, In Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 82–87

Hu W, Yao LG, Hua ZZ (2008) Optimization of sheet metal forming processes by adaptive response surface based on intelligence sampling method. J Mater Process Technol 197(1–3):77–88

Hu W, Li E, Yao G (2009) The least square support vector regression with parallel sampling scheme metamodeling technique and application in sheet forming optimization. Mater Des 30:1468–1479

Janusevskis J, Le Riche R, Ginsbourger D, Girdziusas R (2012) Expected improvements for the asynchronous parallel global optimization of expensive functions: potentials and challenges, in learning and intelligent optimization. Springer, Berlin, pp 413–418

Jin Y (2005) A comprehensive survey of fitness approximation in evolutionary computation. Soft Comput 9(1):3–12

Jin Y (2011) Surrogate-assisted evolutionary computation: recent advances and future challenges. Swarm Evol Comput 1(2):61–70

Jones DR (2001) A taxonomy of global optimization methods based on response surfaces. J Glob Optim 21(4):345–383

Jones DR, Perttunen CD, Stuckman BE (1993) Lipschitzan optimization without the lipschitz constant. J Optim Theory Appl 79(1):157–181

Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-Box functions. J Glob Optim 13(4):455–492

Kogiso N, Watson LT, Gürdal Z, Haftka RT (1994) Genetic algorithms with local improvement for composite laminate design. Struct Optimiz 7(4):207–218

Krause A and Ong CS (2011) Contextual gaussian process bandit optimization, In Advances in Neural Information Processing Systems, 2447–2455

Kushner HJ (1964) A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J Basic Eng 86(1):97–106

Le Riche R, Haftka RT (1993) Optimization of laminate stacking sequence for buckling load maximization by genetic algorithm. AIAA J 31(5):951–956

Li Z, Ruan S, Gu J, Wang X, and Shen C (2016) Investigation on parallel algorithms in efficient global optimization based on multiple points infill criterion and domain decomposition. Structural and Multidisciplinary Optimization. doi:10.1007/s00158-016-1441-2

M.Gorges-Schleuter (1989) "ASPARAGOS An Asynchronous Parallel Genetic Optimisation Strategy". Procs. of the 3rd ICGA, J. D. Schaffer (ed.), Morgan Kaufmann, pp. 422–427

Mockus J, Tiesis V and Zilinskas A (1978) The application of Bayesian methods for seeking the extremum, in L.C.W. Dixon and G.P. Szego (eds.), Towards Global Optimisation, Vol.2, pp. 117–129. North Holland, Amsterdam

Müller J, Piché R (2011) Mixture surrogate models based on Dempster-Shafer theory for global optimization. J Glob Optim 60:123–144

Müller J, Shoemaker CA (2014) Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems. J Glob Optim 51:79–104

Ong YS, Nair PB, Keane AJ (2003) Evolutionary optimization of computationally expensive problems via surrogate modeling. AIAA J 41(4):687–696

Parno MD, Hemker T, Fowler KR (2012) Applicability of surrogates toimprove efficiency of particle swarm optimization for simulation-based problems. Eng Optim 44(5):521–535

Parr JM, Keane AJ, Forrester AIJ, Holden CME (2012) Infill sampling criteria for surrogate-based optimization with constraint handling. Eng Optim 44(10):1147–1166

Peri D and Tinti F (2012) A multistart gradient-based algorithm with surrogate model for global optimization, Communications in Applied and Industrial Mathematics, 3(1)

Pettey C, Leuze MR, Grefenstette J (1987) "A Parallel Genetic Algorithm". Proceedings of the 2nd ICGA, J. Grefenstette (ed.), Lawrence Erlbraum Associates, pp. 155–161

Picheny V, Ginsbourger D, Roustant O, Haftka RT and Kim N (2010) Adaptive design of experiments for accurate approximation of a target region, J Mech Des 132 (7)

Ranjan P, Bingham D, Michailidis G (2008) Sequential experiment design for contour estimation from complex computer codes. Technometrics 50(4):527–541

Ranjan P, Bingham D, Michailidis G (2011) Errata. Technometrics 53(1): 109–110. doi:10.1198/TECH.2011.10192

Regis RG (2014) Particle swarm with radial basis function surrogates for expensive black-box optimization. J Comput Sci 5(1):12–23

Regis RG (2015) Trust Regions in Surrogate-Assisted Evolutionary Programming for Constrained Expensive Black-Box Optimization, Evolutionary Constrained Optimization, S Infosys Science Foundation Series, 51–94

Regis RG, Shoemaker CA (2005) Constrained global optimization of expensive black box functions using radial basis functions. J Glob Optim 31(1):153–171

Regis RG, Shoemaker CA (2007a) Parallel radial basis function methods for the global optimization of expensive functions. Eur J Oper Res 182(2):514–535

Regis RG, Shoemaker CA (2007b) Stochastic radial basis function method for the global optimization of expensive functions. INFORMS J Comput 19(4):497–509

Regis RG, Shoemaker CA (2009) Parallel stochastic global optimization using radial basis functions. INFORMS J Comput 21(3):411–426

Rosales-Perez A, Coello Coello CA, Gonzales JA, Reyes-Garcia CA, and Escalante HJ (2013) A Hybrid Surrogate-Based Approach for

Evolutionary Multi-Objective Optimization, IEEE Congress on Evolutionary Computation, 2548–2555

Sareni B, Krähenbühl L (1998) Fitness sharing and niching methods revisited. IEEE Trans Evol Comput 2(3):97–106

Sasena M (2002) Flexibility and Efficiency Enhancements for Constrained Global Design Optimization With Kriging Approximations, Ph.D. thesis, University of Michigan, Ann Arbor, MI

Schutte JF, Reinbolt JA, Fregly BJ, Haftka RT, George AD (2004) Parallel global optimization with the particle swarm algorithm. Int J Numer Methods Eng 61(13):2296–2315

Schutte JF, Haftka RT, Fregly BJ (2007) Improved global convergence probability using multiple independent optimizations. Int J Numer Methods Eng 71(6):678–702

Shao T, Krishnamurthy S (2008) A clustering-based surrogate model updating approach to simulation-based engineering design. J Mech Des 130(4):041101

Sobester A, Leary S, Keane AJ (2004) A parallel updating scheme for approximating and optimizing high fidelity computer simulations. Struct Multidiscip Optim 27(5):371–383

Spiessens P, Manderick B (1991) "A Massively Parallel Genetic Algorithm". Proceedings of the 4th International Conference on Genetic Algorithms, R. K. Belew, L. B. Booker (eds.), Morgan Kaufmann, pp. 279–286

Srinivas N, Krause A, Kakade SM, and Seeger MW (2010) Gaussian process optimization in the bandit setting: No regret and experimental design, In Proceedings of the 27th International Conference on Machine Learning (ICML), 1015–1022

Srinivas N, Krause A, Kakade SM, Seeger MW (2012) Information theoretic regret bounds for gaussian process optimization in the bandit setting. Inf Theory IEEE Trans on 58(5):3250–3265

Sun C, Jin Y, Zeng J, Yu Y (2015) A two-layer surrogate-assisted particle swarm optimization algorithm. Soft Comput 19:1461–1475

Syberfeldt S, Grimm H, Ng A and John RI (2008) A parallel surrogate-assisted multi-objective evolutionary algorithm for computationally expensive optimization problems, In Proceedings of Evolutionary Computation (IEEE World Congress on Computational Intelligence)

Van Keulen F, Toropov VV (1999) The multi-point approximation methods in a parallel computing environment. ZAMM-J Appl Math Mech 79(S1):67–70

Venkataraman S, Haftka RT (2004) Structural optimization complexity: what has Moore's law done for us? Struct Multidiscip Optim 28(6): 375–387

Viana FAC and Haftka RT (2010) Surrogate-based global optimization with parallel simulations using the probability of improvement, 13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Fort Worth, USA, September 13–15, AIAA 2010–9392

Viana FAC, Haftka RT, Watson LT (2012) Sequential sampling for contour estimation with concurrent function evaluation. Struct Multidiscip Optim 45(4):615–618

Viana FAC, Haftka RT, Watson LT (2013) Efficient global optimization algorithm assisted by multiple surrogate techniques. J Glob Optim 56(2):669–689

Villanueva D, Haftka RT, Le Riche R and Picard G (2013) Locating Multiple Candidate Designs with Surrogate-Based Optimization, 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, FL, USA

Villemonteix J, Vazquez E and Walter E (2009) An informational approach to the global optimization of expensive-to-evaluate functions. Journal Of Global Optimization 44(4): 509-534

Wang GG (2003) Adaptive response surface method using inherited latin hypercube design points. J Mech Des 125(2):210–220

Wang GG, Dong Z, Aitchison P (2001) Adaptive response surface method-a global optimization scheme for approximation-based design problems. Eng Optim 33(6):707–734

Wang L, Shan S, Wang GG (2004) Mode-pursuing sampling method for global optimization on expensive black-box functions. Eng Optim 36(4):419–438

Wang H, Li E, Li GY (2010) Parallel boundary and best neighbor searching sampling algorithms for drawbead design optimization in sheet metal forming. Struct Multidiscip Optim 41:309–324

Wang H, Shan S, Wang GG, Li G (2011) Integrating least square support vector regression and mode pursuing sampling optimization for crashworthiness design. J Mech Des 133(4):041002. doi:10.1115/1.4003840

Watson LT, Baker CA (2001) A fully-distributed parallel global search algorithm. Eng Comput 18(1/2):155–169

Yang XS (2010) Nature-inspired metaheuristic algorithms. Luniver press, India

Zerpa LE, Queipo NV, Pintos S, Salager J-L (2005) An optimization methodology of alkaline–surfactant–polymer flooding processes using field scale numerical simulation and multiple surrogates. J Pet Sci Eng 47(3–4):197–208

Zhao D, Xue D (2011) A multi-surrogate approximation method for metamodeling. Eng Comput 27(2):139–153

Zhou Z, Ong YS, Nair PB, Keane AJ and Lum KY (2007a) Combining Global and Local Surrogate Models to Accelerate Evolutionary Optimization, IEEE Transactions on Systems man, and Cybernetics MAN, —PART C: APPLICATIONS AND REVIEWS, VOL. 37, NO. 1

Zhou Z, Ong YS, Lim MH, Lee BS (2007b) Memetic algorithm using multi-surrogates for computationally expensive optimization problems. Soft Comput 11(10):957–971

Zhu P, Zhang S, Chen W (2015) Multi-point objective-oriented sequential sampling strategy for constrained robust design, engineering optimization strategy for constrained robust design. Eng Optim 47(3):287–307. doi:10.1080/0305215X.2014.887705