Optimal sampling strategies for learning a fitness model

Alain Ratle
Département de génie mécanique,
Université de Sherbrooke,
Sherbrooke, Québec, J1K 2R1 Canada
alain.ratle@gme.usherb.ca

Abstract- This paper investigates the use of kriging interpolation and estimation as a function approximation tool for the optimization of computationally complex functions. A model of the fitness function is built from a small number of samples of this function. This model is utilized in a model-based learning strategy as an auxiliary fitness function. The kriging approach represents a compromise between global models and local models. The model is initially a global approximation of the entire domain, and successive updates during the optimization process transform it into a more precise local approximation. In this paper, several approaches for the sampling of the true fitness function are investigated in order to update a fitness model efficiently and at a low computational cost.

1 Introduction

Fitness landscape modeling is a new approach to the optimization of computationally expensive functions by evolutionary algorithms. In a previous paper [1], kriging interpolation has been presented as a general function approximation tool for a model-based learning approach to the optimization of computationally expensive functions. The approach consists of estimating a statistical model of the fitness function from the information given by the first generation of an evolutionary algorithm. This model is subsequently exploited as an auxiliary function to optimize instead of the true fitness function. The model is expected to be, from a computational point of view, much simpler than the true fitness function. This modeling approach is aimed by the way at problems for which the fitness function is very expensive to calculate, such as in engineering design problems.

Function approximation has often been utilized in the past for solving optimal structural design problems [2]. Models are traditionally either global or local, but not both. A global model usually gives a rough estimate of a function over a large domain, while a local model gives a more precise estimate over a smaller subdomain. The approach presented in this paper is somehow a compromise between long range global models and local models: the kriging approach builds up a global model that includes local perturbations as well as global influences. Unless the fitness function is very simple, models built from a small number of random samples obtained during the first generation of an evolutionary algorithm cannot yield anything better than a very rough global approximation. However, by periodically updating the model in order to introduce new information and refine it around the promising areas of the search space, the model is expected to become more precise in these areas, regardless of the others. This is precisely what one needs for optimization purposes. Since no a priori assumptions can be stated, in the general case, on the location of interesting regions, an initially global model is needed. In the later stages of optimization, it is better to have locally optimal models in the regions of higher fitness, regardless of the modeling quality in the regions of lower fitness

This paper investigates various strategies for the model update process in order to minimize the total number of true fitness function evaluations required to obtain an equivalent fitness level. These strategies are experimented over two theoretical test-problems, and conclusions from these problems are used to solve a simple structural design problem, which consists of positioning a certain number of point-masses on a vibrating rectangular thin plate in order to reduce its global vibration level over a certain frequency range.

2 Fitness function approximation

2.1 Kriging theory

Kriging is a general-purpose tool for the modeling and estimation of non-uniformly distributed experimental data points in multi-dimensional spaces. The original method was formulated for 1, 2 or 3 dimensional problems that are found in the area of mining geostatistics [3]. For the purpose of function optimization, the dimensionality of the search space is clearly a problem-dependent factor, and a general L-dimensional version of this tool is utilized [1]. Denoting by $\mathbf{x}$ a vector representing a point in the search space, kriging creates a model $U(\mathbf{x})$ of the function $F(\mathbf{x})$ defined as:

$$U(\mathbf{x}) = a(\mathbf{x}) + b(\mathbf{x}) \cong F(\mathbf{x}) \qquad \text{with} \quad \mathbf{x} = \{x_1, x_2, \dots, x_L\} \tag{1}$$

The drift function, noted $a(\mathbf{x})$, represents the long range expected value of the true fitness function. This is the global part of the kriging model. The drift can be modeled in various ways, such as polynomial or trigonometric series. Even though almost any representation might gives interesting results, the former is retained in the present study. A sum of M basis functions $f_j(\mathbf{x})$ is considered with M undetermined coefficients a_j :

$$a(\mathbf{x}) = \sum_{j=1}^{M} a_j f_j(\mathbf{x}) \tag{2}$$

The second part of $U(\mathbf{x})$ is the covariance function $b(\mathbf{x})$. This function represents a short-distance influence of every data point on the model, hence the local aspect of the kriging approach. The general formulation for $b(\mathbf{x})$ is a weighted sum of N functions, $K_n(\mathbf{x})$, $n = 1 \dots N$. These functions are the generalized covariance functions between the n^{th} data point $\mathbf{x}_n$ and any point $\mathbf{x}$. The fundamental hypothesis is that no data point has any particularly significant value compared to another. This allows a standard covariance function $K(h)$ to be considered, depending only on the distance h between two points. In a way similar to the drift function, any set of basis functions is suitable for the covariance function, as long as the expected continuity conditions are respected and the model has a unique solution for the coefficients. However, the shape of $K(h)$ has a strong influence on the resulting aspect of the statistical model. For example, as shown by Matheron [4], kriging is exactly equivalent to spline interpolation for some particular cases of $K(h)$. This paper considers two categories of covariance functions. The first one is an arbitrarily defined polynomial function. Kriging under this case is considered to be an *interpolator* [5]. Otherwise, an experimental covariance can be estimated from the data points, as described in [6] and kriging is then considered to be an *estimator*. Building up the kriging function in the format described above requires N covariance coefficients and M drift coefficients. There are by the way $N + M$ independent equations that must be stated. A first set of N equations is given by the application of Eq. (1) to each of the sample points:

$$\begin{aligned} &\sum_{i=1}^{N} b_i K(h(\mathbf{x}_i, \mathbf{x}_n)) + \sum_{j=1}^{M} a_j f_j(\mathbf{x}_n) \\ &= F(\mathbf{x}_n), \quad \text{for } n = 1 \dots N \end{aligned} \tag{3}$$

where $h(\mathbf{x}_i, \mathbf{x}_n)$ is the distance between the points $\mathbf{x}_i$ and $\mathbf{x}_n$. The next equations are found using the statement that the drift function represents the expected value of the true fitness landscape. The covariance function must therefore have a null expected value over the domain. This is called the *no-bias* condition of the estimator. Applying this condition to the covariance gives the next M equations:

$$\sum_{i=1}^{N} b_i f_j(\mathbf{x}_i) = 0, \; j = 1 \dots M \tag{4}$$

These equations ensure that the net contribution of the covariance function over all the sample points is zero. The $M + N$ coefficients are found by solving the matrix system defined by Eq. (3) and (4).

2.2 Optimization algorithm

A modified class of optimization algorithms is proposed using a fitness landscape approximation. These algorithms might be founded on any base class of generational evolutionary algorithm, such as genetic algorithms, evolutionary programming or evolution strategies. The question of steady-state evolutionary algorithms has not yet been addressed, although it might be interesting in the context of a model-based learning with continuous update of the model. For the present study, the base class is a genetic algorithm using a real-valued encoding, as described in Bäck [7] or Michalewicz [8].

Algorithm 1 shows the integration of a fitness landscape approximation into the base algorithm. The first generation is classically initiated using random values. Fitness is then evaluated using the true fitness function. Solutions found in this first generation are taken for the construction of a first fitness landscape model by kriging interpolation or estimation. This model is exploited for a fixed number of generations k. During the next generation, that is, the $(k+1)^{\text{th}}$ generation, some points are evaluated using the true fitness function and the kriging model is updated using some of these new points, depending on the sampling strategy. The process is repeated until the global stopping criterion is satisfied.

Algorithm 1 Evolutionary optimization algorithm with fitness landscape approximation

```
Begin
  Initialize μ individuals (μ = pop. size)
  Evaluate fitness with the original function
  Build a first fitness model
  fitness function ← model
  while Stopping criterion not satisfied
    if k generations with the same model
      fitness function ← original function
    end if
    Select individuals according to fitness
    Recombine and mutate
    Evaluate fitness, using current function
    if fitness function = original function
      Update the model with new samples
      fitness function ←new model
    end if
  end while
End Algorithm
```

3 Sampling strategies

Using this optimization algorithm, the model has to be updated after k generations have been performed using the function approximation. In this model, no true function evaluation take place between two model updates. There might be an optimal value for this number of generations, but it is taken as a constant in this study. This paper address the problem of how many new samples should be taken from the true fitness function at every update, and how should these new samples replace the old ones. This task might be handled in a large number of different ways. The 6 strategies presented in Table 1 are considered for updating the model. A population size of μ is assumed and the model is also assumed to include a number of samples N equal to μ, at least in the first generation. Each strategy consists of evaluating a certain number of points using the true fitness function and incorporating some of these new points in the model.

Strategy 1 consists of evaluating μ new individuals, that is, the complete generation, and replacing all the old samples with them. The model is therefore completely rebuilt from scratch every time. It is expected that the first model is best interpreted as a global approximation, while the subsequent ones shall be more precise in some regions and less in others. The second sampling strategy still requires the evaluation of μ points, but only the ξ best ones among them are selected, with $\xi \leq \mu$. These new points replace the ξ worse points in the current population. This strategy represents what is called an elitist replacement of the samples, since the best samples are assured to survive. The third strategy is similar to the second, except that the ξ selected points replace ξ random points instead of the worse points, giving an universal rather than elitist replacement.

The strategy 4 is the only one that does not keep constant the number of samples contained in the model. After every update, μ points are evaluated, and the best ξ ones are selected and added up to the previous model. Hence the quantity ξ should be kept small in order to avoid having a too large kriging system. Otherwise, solving the kriging system would become rapidly a non- negligible task, and the basic assumption on the computational complexity of the model compared to the true fitness function would be no more valid. The fifth approach consider the creation and evaluation of only ξ new points, instead of creating μ points and discarding $\mu - \xi$ of them. These points replace the ξ worse points in the population used to update the model. Finally the sixth and last approach does not creates any new point, but evaluates with the true fitness function the ξ best points in the model's population. The last two strategies are more economical in function evaluations, since only a fraction ξ of the population size μ is evaluated. The evaluation of the whole population using the strategies 1 to 4 might however gives more choice and more information for the model rebuilding. The main features of the six sampling strategies are summarized in Table 1.

4 Computational experiments

4.1 Theoretical functions

The six sampling approaches presented above have been tested on two theoretical problems in a 10-dimensional space. Preliminary conclusions and insights from these problems have been utilized to solve a simple "real-world" problem: the vibration-level reduction of a rectangular plate by the optimal positioning of a certain number of point-masses. The first theoretical problem is the minimization of the Rastrigin's multimodal function. For the purposes of this study, the problem takes the following form:

$$\begin{aligned} &\text{Minimize} \quad f_1(\mathbf{x}) = 10L + \sum_{i=1}^{L} \left(x_i^2 - 10\cos(2\pi x_i)\right) \\ &\text{subject to} \quad -2 \leq x_i \leq 2 \end{aligned} \tag{5}$$

Table 1: Description of the 6 sampling strategies, with the number of solutions being created (C), evaluated (E) with the true fitness function, and included (I) in the new model for each strategy.

No	C	E	I	Description
1	μ	μ	μ	Evaluate μ points with the current generation and build a new model from scratch.
2	μ	μ	ξ	Replace the ξ worst points in the model with the ξ best ones among the μ new points.
3	μ	μ	ξ	Replace ξ randomly chosen points in the model with the ξ best ones among the μ new points.
4	μ	μ	ξ	Add to the current model the ξ best points among the μ new points.
5	ξ	ξ	ξ	Create ξ new points, evaluate them and replace the ξ worst points in the model.
6	0	ξ	ξ	Simply re-evaluate with the true fitness function the ξ best points in the current model's population.

The second test problem is the Fletcher & Powell function [9], stated as:

$$\begin{aligned}\text{Minimize}\quad f_2(\mathbf{x}) &= \sum_{i=1}^{L}(A_i - B_i)^2 \\ A_i &= \sum_{j=1}^{L}(a_{ij}\sin\alpha_j + b_{ij}\cos\alpha_j) \\ B_i &= \sum_{j=1}^{L}(a_{ij}\sin x_j + b_{ij}\cos x_j) \\ \alpha_j, x_j \in [-\pi,\pi] \quad & a_{ij}, b_{ij} \in [-100, 100]\end{aligned} \tag{6}$$

The vector α and the matrices a and b contains randomly chosen values. However, these values have been chosen only once and held constant for all the study. Both problems are considered in a 10-dimensional space. The problems have been solved using an evolutionary algorithm together with a kriging approximation model. Relevant parameters for the evolutionary algorithm and function reconstruction are given on Table 2.

Table 2: Parameters settings for the evolutionary algorithm and kriging model.

Parameter	Value
Evolutionary algorithm	
Number of variables	10
Population size	200
Maximum fitness function evals.	4000
Selection operator	Tournament, size=3
Mutation type	Gaussian noise
Mutation rate	1 mut. / 40 variables
Kriging parameters	
Drift function	cubic
Covariance function	$K(h) = 1-3(h/h^*)^2 + 2(h/h^*)^3$ for $h < h^*$
Distance of influence	$h^* = 0.15$
Generations between updates	10

The theoretical covariance model presented on Table 2 have been used for the Fletcher-Powell problem. This cubic model ensures a continuity to degree C^1. For the Rastrigin's function, an experimental estimation of the covariance have been performed, based on a least-square regression over a Bezier-Bernstein polynomial [6]. The allowed number of generations in both cases have been calculated in order to terminate the algorithm after a fixed number of true fitness function evaluations have been performed. This depends on the sampling strategy considered, since the number of true fitness function evaluation is not constant from one strategy to another.

4.2 Summary of results

The principal results are summarized on Table 3, for the Fletcher-Powell's function, and on Table 4, for the Rastrigin's function. These tables presents the mean fitness values obtained in 5 runs of the same algorithm on the same problem, for the basic algorithm and the model-based algorithm with the 6 sampling strategies and several different values of the parameter ξ. Although this number of run is rather small, compared to accepted practices, the statistical analysis presented in the next section has shown that it is enough to get significant results. The two last columns give the average results for the two problems at hand, after 1000 and 3000 true function evaluations. Algorithms are compared on the basis of true function evaluations, rather than on actual CPU time, since the basic assumption is that the fitness function evaluation requires most of the CPU time, although this is obviously not the case for simple theoretical problems. Boldfaces in Tables 3 and 4 indicate the best and second best results for each of the four columns.

Comparison of strategies 2 and 3 cannot really tells whether an elitist replacement is better or not than an universal replacement, since the results cannot be clearly separated. These two strategy does not seem to be clearly better than the basic evolutionary algorithm. The strategy 4 is also unable to give any significant improvement over all the cases. It is observed that on the average, strategy 6 gives better results than the other strategies, while the differences are sometime subtles. This strategy is therefore the chosen for investigations on the structural design problem.

Table 3: Summary of optimization results for the 6 sampling strategies with the Fletcher-Powell function

		Fitness value ($\times 10^4$)	
Strategy	ξ	1000 evals.	3000 evals.
Basic EA	N/A	6.04	2.41
1	N/A	7.69	3.77
2	10	8.30	**1.22**
	50	5.90	4.48
	100	9.42	6.03
3	10	7.05	3.45
	50	5.08	2.54
	100	5.95	2.41
4	10	9.40	4.51
	20	9.66	3.86
5	10	6.21	2.89
	50	**3.36**	2.56
	100	6.77	3.19
6	10	3.81	3.26
	50	**1.43**	**0.791**
	100	4.15	1.58

Table 4: Summary of optimization results for the 6 sampling strategies with the Rastrigin function

		Fitness value	
Strategy	ξ	1000 evals.	3000 evals.
Basic EA		29.5	**1.98**
1	N/A	11.5	10.7
2	10	5.65	5.65
	50	6.96	6.72
	100	6.46	6.05
3	10	6.72	4.40
	50	9.40	6.50
	100	4.66	4.04
4	10	5.78	5.17
	20	5.26	5.03
5	10	8.42	6.69
	50	12.0	11.5
	100	7.93	5.45
6	10	5.49	5.40
	50	**3.19**	2.89
	100	**3.12**	**2.70**

4.3 Significance of results

Evolutionary algorithms rely heavily on random processes. The results summarized on Table 3 and 4 are inevitably dependent both on the effect of the parameters under study and the effect of random noise. A total of five repetitions have been done for each case, this might or might not be sufficient to conclude on the significance of the results. For that reason, a statistical analysis of the variance has been performed. The method is shortly described in appendix, and the results are summarized on Table 5. The following null hypothesis is stated:

$$H_0: \text{The difference between treatments is due to random noise only.} \tag{7}$$

A treatment is considered in this case to be a particular algorithm applied on a given problem. The confidence levels given on Table 5 represents the degree of certainty in the rejection of the null hypothesis H_0. The analysis of Table 5 have been performed globally on the results of the strategies 5 and 6, with ξ = 10, 50, and 100. These results indicate that for the two problems, the observed differences in mean values after 1000 function evaluations is significant with a 99.5% confidence, while the confidence drops to 95% after 3000 evaluations for the Fletcher-Powell problem. Table 5 also presents the percentage of variance that can be attributed to the effect of the treatments compared to the total variance (treatments + random noise). Results are presented only for the 6 cases using strategies 5 and 6, since these are the most promising ones.

A similar analysis is also presented on Table 6 using only the two most interesting cases: strategy 6 with 10 and 50 new points per model reconstruction. Confidence levels obtained in this more specific analysis are similar to those of the global analysis. The only questionable hypothesis that should be assumed in such an analysis is that the sum of the significant results plus the random noise inherent to the experiments presumably follow a normal distribution. This hypothesis is often assumed in statistical analysis of experiments, but can hardly be proven with certainty.

It is interesting to note that this statistical analysis may be utilized as a convergence criteria: the drop in confidence level and in the percentage of variance due to treatments indicates that no matter which algorithm is taken, no more significant improvement in fitness value is obtained after 3000 fitness function evaluations. Convergence behavior is illustrated on Fig. 1 for the Fletcher-Powell problem and Fig. 2 for the Rastrigin problem. It is observed in both cases that up to about 1000 true function evaluations, the reconstruction approach gives significant increases in convergence velocity. However, after a longer time, the "basic evolutionary algorithm" tends to outperforms most of the other approaches.

Table 5: Statistical analysis of global results obtained with strategies 5 and 6.

Function	Evaluations	Confidence	Significance
F.-P.	1000	0.995	56.0
	3000	0.950	24.4
Rastrigin	1000	0.995	41.82
	3000	0.995	44.6

Table 6: Statistical analysis of the results obtained in strategie 6 with ξ = 10 and 50 points.

Function	Evaluations	Confidence	Significance
F.-P.	1000	0.995	84.2
	3000	0.995	79.2
Rastrigin	1000	0.990	56.1
	3000	0.975	48.7

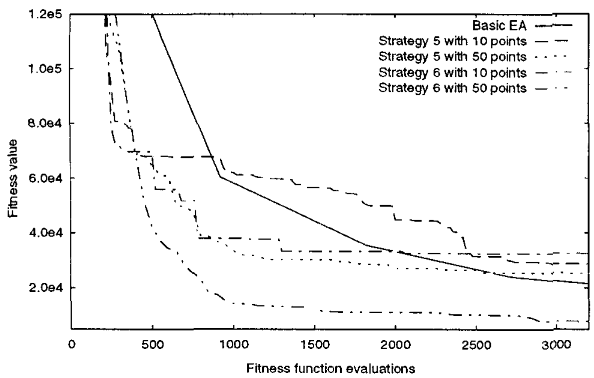

Figure 1: Learning curve for the 10-D Fletcher-Powell function with sampling strategies no 5 & 6.

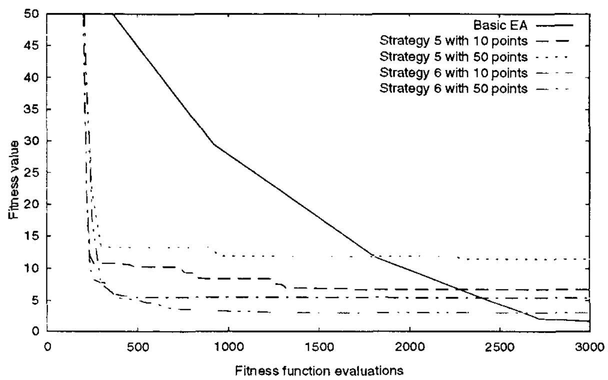

Figure 2: Learning curve for the 10-D Rastrigin function with sampling strategies no 5 & 6.

4.4 A simple vibroacoustic design problem

This section shows an application of the results presented in this paper toward the design of a simple mechanical structure with a noise or vibration level reduction criteria. The mechanical model is described in more details in [10]. Figure 3 shows the general layout and significant parameters.

The system is composed of a thin rectangular simply supported on its four edges. The plate is carrying five point-masses and is excited either by a harmonic point force F of unit amplitude. or by an acoustic plane wave of amplitude P_0 and incidence angles θ and ϕ. The system is modeled using a Rayleigh-Ritz modal decomposition [11]. The fitness function to be minimized is the mean squared velocity of the plate averaged over a frequency band comprising 2 natural frequencies of the plate. Due to the presence of more than one natural frequency, the problem is expected to have many locally optimal solutions. The optimization variables are the two Cartesian coordinates of the masses, giving 10 real parameters to be optimized. Optimization results for the point-force excitation are presented on Fig. 4 using a basic evolutionary algorithm similar to that presented on Table 2, and using the reconstruction algorithm with the sampling strategy number 6 with 10 points evaluated after every 10 generations on the same model. The covariance function was represented either as a theoretical model, or as an experimental polynomial estimated from the actual samples. The fitness value indicated on Fig. 4 represents the mean square velocity of the plate in decibels.

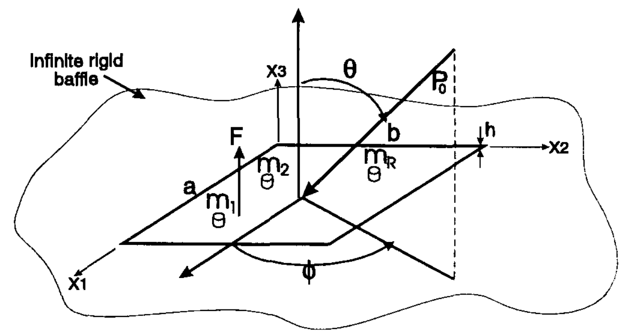

Figure 3: Description of the plate-mass system considered in the present application.

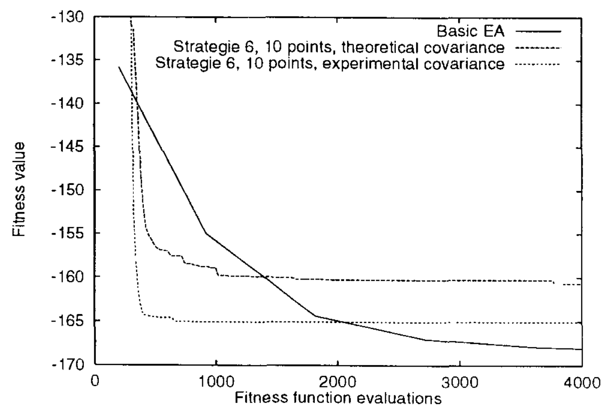

Figure 4: Optimization results for the structural design problem.

The results of Fig. 4 show that although the function approximation algorithms allow a rapid identification of moderately good solutions, during the first 2000 evaluations, the quality of these solutions is rapidly outperformed by a simple evolutionary using only the true fitness functions. It is also observed that the algorithm using an experimental covariance model settles to a lower fitness value than with the theoretical model. This suggests that some more information is gained on the problem by the use of an experimentally estimated covariance. The lack of performance of the reconstruction algorithms might indicates that although the kriging model utilized in this paper was able to produce satisfactory models of the relatively simple theoretical problems, the potentially complex and unknown structure of this kind of real-world problem can not be adequately modeled using the same tools. It also indicates that results and insights from one problem can not be easily transposed to another problem.

5 Conclusions

This paper has presented various approaches for the reconstruction of an unknown fitness function from a small number of samples, in order to gain informations on an optimization problem with a reduced number of function evaluations. these gains should not however be interpreted as a creation of information, but rather as the preservation of informations that would otherwise be lost. Results obtained from theoretical problems have shown that significant gains can be obtained compared to a basic evolutionary algorithms. Better results have been obtained by periodically correcting about 5 to 25% of the points in a model using the true fitness function, rather than replacing a fraction of the points with new ones and thus changing the structure of the model.

However, the translation of these results to a real-world problem of unknown structure has shown to be very difficult. It might be argued that the theoretical problems present a relatively "smooth" and easily characterized surface, in spite of their large number of local minima, while real-world problems may have arbitrarily large surface roughness and unpredictable periodicity. This suggests that for model estimation tools, there are "no free lunches", in a similar way as for optimization algorithms [13]. This latter assertion means that it would be unrealistic to think that one sort of model estimation tool can be suitable for any kind of phenomena to be estimated. Research should rather be oriented toward adaptive modeling tools that can adapt their structure to any problem in a very flexible way. One of these promising approaches is the use of neural networks as a function estimation tools. Since it has been suggested that feedforward neural networks can be used as general modeling tools [14, 15, 16], the combination of an evolutionary algorithm with a neural learning tool is likely to open interesting prospects.

It might also be argued that the use of a model-based optimizer is a hopeless issue, since the No Free Lunch theorems states that no information can be created by an optimization algorithm. This is referred to as the conservation of information principle. Although it is generally clear that no statistical model can create any new information, it can prevent this information from being lost. Most generation-based evolutionary algorithms actually throw away a lot of significant information by restarting from scratch after every generation. This is where a model-based approach can help increasing the efficiency of an optimization algorithm with no violation of the No Free Lunch theorems.

Acknowledgments

The author would like to acknowledge the anonymous referees for their helpful suggestions. The financial support from the IRSST is greatly acknowledged.

Bibliography

[1] A. Ratle. Accelerating the convergence of evolutionary algorithms by fitness landscape approximation. In M. Schoenauer, H.-P. Schwefel, A. E. Eiben & T. Bäck, editors, *Parallel Problem Solving from Nature V*, pages 87–96, 1998.

[2] J.-F.M. Barthelemy & R.T. Haftka. Approximation concepts for optimum structural design - a review. *Structural Optimization*, 5:129–144, 1993.

[3] G. Matheron. The intrinsic random functions and their applications. *Adv. Appl. Prob.*, 5:439–468, 1973.

[4] G. Matheron. Splines et krigeage: leur équivalence formelle. Technical Report N–667, Centre de Géostatistique, École des Mines de Paris, 1980.

[5] F. Trochu. A contouring program based on dual kriging interpolation. *Engineering with Computers*, 9:160–177, 1993.

[6] A. Ratle. Kriging as a surrogate fitness landscape in evolutionary optimization. submitted to *Artificial Intelligence in Engineering Design, Analysis and Manufacturing*, 1998.

[7] T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1996.

[8] Z. Michalewicz. *Genetic Algorithms+Data Structures=Evolution Programs.* Springer-Verlag, 1994.

[9] R. Fletcher & M.J.D. Powell. A rapidly convergent descent method for minimization. *Computer Journal*, 6:163–168, 1963.

[10] A. Ratle & A. Berry. Use of genetic algorithms for the vibroacoustic optimization of a plate carrying point-masses. *Journal of the Acoustical Society of America*, 1043385–3397, 1998.

[11] M. C. Junger & D. Feit. *Sound, Structures and Their Interaction.* Acoustical Society of America, 1993.

[12] C. R. Hicks. *Fundamental Concepts in the Design of Experiments.* Saunders College Publishing, 1993.

[13] D.H. Wolpert & W.G. MacReady. No-free-lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*,1:1 pages 67-82, 1998.

[14] K.Funahashi. On the Approximate Realization of Continuous Mappings by Neural Networks, *Neural Networks*, 2 pages 183–192, 1989.

[15] K. Hornik, M. Stinchcombe & H. White. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2 pages 359–366, 1989.

[16] H. White. Connectionnist Nonparametric Regression: Multilayer Feedforward Networks Can Learn Arbitrary Mappings. *Neural Networks*, 3 pages 535–549, 1990.

A Statistical analysis

This section presents a brief summary of the analysis of variance technique commonly used in the design of experiments [12]. In a general one-factor experiment, k different treatments, or factor levels, are considered, and n_j observations are done over each treatment j, $j = 1, \ldots, k$. The factor levels might be qualitatives or quantitatives. The goal of the analysis is to assert whether the observed differences between the mean values of the treatments is due to experimental noise alone or not. The analysis assumes that all the experimental results are of the form $Y_{ij} = \mu + \tau_j + \epsilon_{ij}$ where Y_{ij} is the value of the i^{th} observation of the j^{th} treatment, μ is the mean value of all observations τ_j is the effect of the j^{th} treatment, and ϵ_{ij} is the noise over each observation. Introducing statistics for μ and τ_j, into the original model, the variance can be broken in two parts as follow:

$$\sum_{j=1}^{k}\sum_{i=1}^{n_j}(Y_{ij} - \bar{Y}_{..})^2 = \sum_{j=1}^{k}\sum_{i=1}^{n_j}(\bar{Y}_{.j} - \bar{Y}_{..})^2 + \sum_{j}^{k}\sum_{i=1}^{n_j}(Y_{ij} - \bar{Y}_{.j})^2 \tag{8}$$

The left hand term is the total sum of squared errors, noted SS_{total}, the first right hand term, noted SS_{tr} is the fraction of the variance due to difference between treatments means, and the second term, noted SS_{er}, is the variance due to differences between observations. The analysis of variance states the null hypothesis: $H_0 : \tau_j = 0 \ \forall j$. This hypothesis should be rejected with a confidence level α if

$$F = \frac{SS_{tr}/(k-1)}{SS_{er}/(N-k)} \geq F_{\alpha,k-1,N-k} \tag{9}$$

where $F_{\alpha,k-1,N-k}$ is the cumulative value of a Fisher distribution of confidence level α with $k-1$ and $N-k$ degrees of freedom, and $N = \sum_{j=1}^{k} n_j$, or $N = kn$, if an equal number of experiments n are performed for each of the k treatments. In this paper, analysis of variance is performed over 2 or 6 treatments, with a constant value of 5 repetitions in each case.

Table 7: Cumulative Fisher distribution

	$F_{\alpha,k-1,N-k}$	
α	$k = 2$ & $N = 10$	$k = 6$ & $N = 30$
0.900	3.46	2.10
0,950	5.32	2.62
0.975	7.57	3.15
0.990	11.3	3.90
0.995	14.7	4.49

After the rejection of the null hypothesis by analysis of variance, components of variance due to the treatments are calculated as followed. An estimate s_ϵ^2 of the variance due to noise σ_ϵ^2 is given by the quantity $MS_{er} = SS_{er}/(N-k)$. The estimate s_τ^2 of the variance due to treatments, σ_τ^2 is obtained indirectly since $MS_{tr} = SS_{tr}/(k-1)$ is an estimator of $\sigma_\epsilon^2 + n\sigma_\tau^2$, where n is the number of observations per treatment. The component of variance due to the treatments, C_{tr} is finally estimated by:

$$C_{tr} = \frac{\sigma_\tau^2}{\sigma_\tau^2 + \sigma_\epsilon^2} \cong \frac{s_\tau^2}{s_\tau^2 + s_\epsilon^2} \tag{10}$$