Ensemble Selection in Expensive Optimization Problems

YOEL TENNE
Ariel University
Department of Mechanical and Mechatronic Engineering
Ariel 40700
ISRAEL
y.tenne@ariel.ac.il

Abstract: Computationally intensive simulations are being extensively used across engineering and science in various design optimization problems. To alleviate the high computational load associated with each simulation run metamodels are used, as they provide predicted objective values at a lower computational cost. However, the optimal metamodel variant is typically unknown and is problem-dependant. In an attempt to alleviate this ensembles use multiple metamodels concurrently and aggregate their predictions. However the optimal ensemble configuration is also problem-dependant and typically unknown. To address this issue, this paper proposes an approach in an optimal ensemble configuration is selected during the search out of a family of candidate ensembles, without a need for user intervention or a-priori domain knowledge. Performance analysis shows that the proposed approach improved the search effectiveness over a range of test problems.

Key–Words: expensive optimization problems, metamodels, ensembles, computational intelligence

1 Introduction

The current availability of high performance computing allows engineers and researchers to evaluate candidate designs with computer simulations instead of using laboratory experiments, thereby reducing the duration and cost of the design process. In this setup, a candidate design is parameterized as a vector of design of variables, and is sent to the simulation for evaluation. Such computer simulations, which still need to be validated with laboratory experiments, transform the design process into an optimization problem having several distinct features (Tenne and Goh, 2010):

- The simulation acts as the objective function as it assigns objective values to candidate designs (input vectors), but it is a 'black-box', namely, the analytic expression of this mapping is unknown. This can occur since the simulation involves intricate calculations, or the simulation's code might be inaccessible to the user. In any case, the lack of an analytic expression presents an optimization challenge.

- Each simulation run is often computationally expensive, and hence only a small number of designs can be evaluated.

- Both the real-world physics being modelled, and the numerical simulation process itself, can yield a black-box function with complicated features, such as multiple optima or discontinuities, which add an additional optimization challenge.

An established solution methodology in such scenarios is to incorporate a *metamodel* into the optimization search. The latter is a mathematical approximation of the true expensive function which provides predicted objective values at a much lower computational cost (Tenne and Goh, 2010). A variety of metamodels have been proposed, but the optimal type is problem-dependant and is typically not known a-priori. To alleviate this issue, *ensembles* use several metamodels concurrently and aggregate their predictions into a single one (Muller and Shoemaker, 2014; Goel et al., 2007; Muller and Piché, 2011). However, the effectiveness of ensembles depends on their topology, namely, which metamodels they incorporate, but again, the optimal topology is typically unknown. To address this issue, this paper proposes an approach in which an optimal ensemble configuration is selected during the search, such that an optimal configuration is continuously being used without the need for user intervention or a-priori domain knowledge. Also, since metamodels are inherently inaccurate, the proposed algorithm operates within a Trust Region (TR) approach to ensure convergence to an optimum of the true expensive function. Performance analysis using both mathematical test functions and a simulation-driven engineering problem shows the effectiveness of the proposed algorithm, and highlights the merit of the proposed dynamic topology adaptation.

The remainder of this paper is organized as follows: Section 2 provides the pertinent background information, Section 3 describes in detail the proposed algorithm, and Section 4 provides an extensive performance evaluation. Lastly, Section 5 concludes this paper.

2 Background

As mentioned above, metamodels (also termed in the literature as *response surfaces* or *surrogates*) are used as computationally cheaper approximations of an intensive numerical simulation. Metamodels are trained with previously evaluated vectors, and some variants include Artificial Neural Networks (ANNs), Kriging, polynomials, and radial basis functions (RBFs) Muller and Shoemaker (2014); Tenne (2012). A typical metamodel-assisted optimization search begins by sampling an initial set of vectors, followed by the main iterative loop in which a metamodel is trained based on the vectors evaluated so far, then an optimization search to locate an optimum of the metamodel, and lastly evaluation of the obtained solution and possibly additional vectors. The process repeats until convergence or when the number of allowed simulation calls is reached. Fig. 1 gives a pseudocode of a typical metamodel-assisted algorithm, while more involved frameworks have also been proposed (Regis and Shoemaker, 2013; Tenne, 2013a).

Figure 1: A typical metamodel-assisted algorithm.

```
sample an initial set of vectors;
while stopping criterion not met do
    train a metamodel based on the cached
    vectors;
    seek an optimum of the metamodel;
    evaluate with the intensive simulation the
    solution found and possibly additional
    vectors;
return the best solution found;
```

While metamodels offer several merits, they also introduce new optimization challenges:

- *Prediction inaccuracy*: Due to the computationally intensive simulation runs only a small number of vectors can be evaluated by it, and hence the resultant metamodel will inherently be inaccurate. Therefore it is necessary to manage the metamodel so as to ensure convergence to a valid solution (Jin et al., 2002). An established approach to safeguard the optimization search is provided by the TR framework (Conn et al., 1997; Powell, 2003), in which a series of trial optimization searches is performed, where each search is confined to a region in which the metamodel is assumed to be adequate. Based on the success or failure of these trials the TR is updated, which in turns ensures asymptotic convergence to a solution of the true expensive objective function Conn et al. (2000). Section 3 gives a detailed description of the TR approach implemented in this study.

- *Metamodel suitability*: An assorted number of metamodels variants have been proposed in the literature, but the optimal variant is problem-dependant and is typically unknown prior to the actual optimization search (Gorissen et al., 2009; Tenne, 2013a). In an attempt to alleviate this, *ensembles* use multiple metamodels and aggregate the individual predictions into a single combined response (Muller and Piché, 2011; Tenne, 2013b). However, the ensemble configuration or *topology* is also problem dependant, and an inadequate topology can decrease the search effectiveness. To demonstrate this aspect, ensembles were generated based on three metamodels: RBFs, radial basis functions neural network (RBFN), and Kriging, as shown in Table 1. The respective prediction accuracy of each metamodel was approximated based on the Root Mean Square Error (RMSE) measure across four test functions in dimensions ranging from 5 to 30. From the test results in follows that the optimal ensemble, namely, that having the smallest Root Mean Square Error (RMSE) varied between the functions, and no single configuration was the overall best. Therefore a-priori fixing a configuration may result in an unsuitable ensemble being used and accordingly diminished performance. To address this issue the following section proposes a dedicated approach.

TABLE 1. THE ROOT MEAN SQUARE ERROR (RMSE) OF DIFFERENT ENSEMBLES CONFIGURATIONS

Function	Ensemble topology			
	R+RN	R+K	RN+K	R+RN+K
Ackley-5D	4.258e-01	3.702e-01	4.151e-01	**2.967e-01**
Rastrigin-10D	1.223e+02	**8.198e+01**	1.312e+02	1.097e+02
Rosenbrock-20D	1.791e+06	1.666e+06	**1.648e+06**	1.693e+06
Schwefel 2.13-30D	**1.882e+06**	2.179e+06	2.343e+06	2.079e+06

R:RBF, RN:RBF neural network, K:Kriging.

3 Proposed approach

Leveraging on the preceding discussion, the approach proposed in this study relies on continuous selection of the ensemble configuration throughout the optimization search so that an optimal configuration is maintained. The proposed approach operates in five main steps, as follows:

Step 1) Initialization: An initial sample of vectors is generated to enable construction of an initial metamodel. In this study the Optimal Latin Hypercube Sampling (OLHS) sampling method was used to ensure an adequate space-filling sample is obtained, as this in turn improves the prediction accuracy of the metamodels (Viana et al., 2009).

Step 2) The vectors which have been sampled and cached so far are split into a training and testing set, and for each prospective metamodel variant a metamodel is trained and tested, thereby yielding an estimated RMSE score. The RMSE of the jth metamodel is calculated as

$$e_j = \sqrt{\frac{1}{l}\sum_{i=1}^{l}\big(m_j(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\big)^2}\,, \qquad (1)$$

where $m_j(\boldsymbol{x})$ is the metamodel generated based on the training set, and $\boldsymbol{x}_i\,, i = 1\ldots l$ are the testing vectors.

Step 3) The sampled vectors are re-split again into training and testing sets, each metamodel variant is re-trained with the new training set, but now each ensemble topology is evaluated. The aggregated prediction of the kth ensemble configuration is then

$$\hat{f}_k(\boldsymbol{x}) = \sum_{j=1}^{n_k} u_j \hat{m}_j(\boldsymbol{x})\,, \quad u_j = \frac{e_j^{-1}}{\sum_{j=1}^{n_k} e_j^{-1}}\,, \qquad (2a)$$

where $\hat{m}_j\,, j = 1\ldots n_k$ are the metamodels incorporated in the ensemble and which have been trained with the training set. The weight assigned to each member metamodel is inversely proportional to its estimated RMSE error obtained earlier. Now the estimated RMSE of the complete ensemble is estimated as

$$\epsilon_k = \sqrt{\frac{1}{l}\sum_{i=1}^{l}\big(\hat{f}(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\big)^2} \qquad (2b)$$

where $\boldsymbol{x}_i\,, i = 1\ldots l$ are the testing vectors in the current testing set, and ϵ_k is the estimated RMSE of the kth ensemble configuration.

Step 4) Amongst all the candidate ensemble configurations the one having the lowest estimated RMSE is chosen as the ‘active’ configuration for the current iteration. Based on this configuration a new ensemble is trained but using all the evaluated vectors so far, namely, without any splitting.

Step 5) Following the Trust Region (TR) framework, a TR is defined around the current best vector ($\boldsymbol{x}_{\mathrm{b}}$), and an optimization search is invoked to find the optimal vector based on the ensemble prediction in the TR. To obtain both an efficient global and local search the search is performed by an evolutionary algorithm (EA) followed by a deterministic SQP solver. During each trial search predicted objective values are obtained only from the ensemble, and no runs of the intensive computer simulation are performed.

Step 6) The optimal vector obtained ($\boldsymbol{x}^\star$) is sent for evaluation with the true expensive function, that is, the simulation, and based on this evaluation the following actions are performed:

- If $f(\boldsymbol{x}^\star) < f(\boldsymbol{x}_{\mathrm{b}})$: The trial step is considered successful since the vector found is indeed better than the current best one. This implies that the ensemble is accurate and the TR radius is doubled.
- If $f(\boldsymbol{x}^\star) \geqslant f(\boldsymbol{x}_{\mathrm{b}})$ *and* there are sufficient vectors inside the TR: The trial step was unsuccessful since the solution predicted by the ensemble to optimal was not in fact so. This implies that the ensemble prediction is inaccurate and since the number of vectors in the TR is deemed as sufficient to support a valid approximation, the failure is attributed to the TR being too large. Accordingly, the TR radius is halved.
- If $f(\boldsymbol{x}^\star) \geqslant f(\boldsymbol{x}_{\mathrm{b}})$ *and* there is an insufficient number of vectors in the TR: as above the trial step was unsuccessful but now the failure is attributed to a poor approximation quality resulting from too few vectors in the TR. Accordingly, a new vector is add in the TR and is evaluated with the true expensive function.

As a change from the classical TR approach, in the approach proposed here the TR radius is reduced only if the number of vectors in the TR is sufficient, to ensure the search is not terminated prematurely. Also, the approach proposed is completely flexible with the metamodels and ensemble configurations. While in this study the metamodels RBF, RBFN, and Kriging were used any other variants or ensemble configurations can equally be incorporated. To complete this section, Fig. 2 presents the pseudocode of an implemented algorithm.

4 Performance analysis

To evaluate the performance gains of the proposed approach the implemented algorithm was applied to both mathematical test functions and a simulation-driven engineering problem, as described in the following text.

4.1 Tests with on mathematical functions

In the first round of tests the well-established set of functions by (Suganthan et al., 2005) was used and which are shown in Table 2, in dimensions in the range of 5 to 40.

For a thorough evaluation the implemented algorithm was benchmarked against four reference algorithms:

- *V1*: A variant of the implemented algorithm which is identical in operation but used only a single metamodel (RBF) and no ensembles. This variant was used to highlight the impact of using ensembles versus using a single metamodel.

- *V2*: A variant of the implemented algorithm which is identical in operation but used a single fixed ensemble of RBF, RBFN, and Kriging metamodels. This variant was used to highlight the impact of the ensemble selection versus using a fixed configuration.

- *EA with Periodic Sampling* (EA–PS): A metamodel-assisted algorithm which leverages on the concepts in (Ratle, 1999; de Jong, 2006). A metamodel-assisted algorithm which uses a Kriging metamodel and an EA, and manages the metamodel by periodically evaluating a small subset of the EA population with the true objective function and thereby refreshing the metamodel. This algorithm is representative of a large variety of metamodel-assisted algorithms in the literature.

Figure 2: An implemented algorithm.

```
/* initialization */
generate an initial sample of vectors and evaluate them with the true objective function;
/* main loop */
repeat
    /* metamodel evaluation */
    split the cached vectors into training and testing sets;
    for each candidate metamodel do
        train a metamodel with the training set;
        estimate its prediction accuracy based on the testing set;
    /* ensemble evaluation */
    re-split the cached vectors into training and testing sets;
    re-train the metamodels using the new training set;
    for each candidate ensemble do
        calculate the ensemble weight of each metamodel incorporated;
        estimate the ensemble prediction accuracy based on the testing set;
    select the most accurate ensemble as the active one;
    re-train the ensemble using all cached vectors;
    centre the TR on the best vector found so far;
    perform an optimization search in the TR using EA+SQP;
    evaluate the vector found with the true expensive function;
    /* TR updates */
    if the new vector is better than the current best then
        double the TR radius
    else if the new vector is not better than the current best and there the number of vectors in the TR is sufficient then
        halve the TR radius;
    else if the new solution is not better than the current best and the number of vectors in the TR is insufficient then
        add a new vector in the TR;
until maximum number of simulation calls;
```

Table 2. Mathematical Test Functions

Function	Definition, $f(\boldsymbol{x}) =$	Domain
Ackley	$-20\exp\left(-0.2\sqrt{\sum_{i=1}^{d} x_i^2/d}\right) - \exp\left(\sum_{i=1}^{d}\cos(2\pi x_i)/d\right) + 20 + e$	$[-32, 32]^d$
Rastrigin	$\sum_{i=1}^{d}\left\{x_i^2 - 10\cos(2\pi x_i) + 10\right\}$	$[-5, 5]^d$
Rosenbrock	$\sum_{i=1}^{d-1}\left\{100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2\right\}$	$[-10, 10]^d$
Weierstrass	$\sum_{i=1}^{d}\sum_{k=0}^{20} 0.5^k\cos\left(2\pi 3^k(x_i + 0.5)\right) - d\sum_{k=0}^{20} 0.5^k\cos(\pi 3^k)$	$[-0.5, 0.5]^d$

- *Expected Improvement with Covariance Matrix Adaptation Evolutionary Strategy* (CMA-ES) (EI–CMA-ES): An involved metamodel-assisted algorithm which uses the CMA-ES optimization algorithm within the expected improvement framework (Büche et al., 2005). This algorithm represents more advanced metamodel-assisted implementations.

This set of algorithms was used to allow allow evaluation of the dynamic ensemble selection and to allow comparison to existing algorithms from the literature. For each algorithm–test function combination 30 runs were repeated so that there were sufficient runs on which a valid statistical analysis could be made. The number of simulations calls, namely, evaluations of the expensive function, was limited to 200, to represent a tight limit on the number of evaluations of the true objective function. Table 3 gives the resultant test statistics of mean, standard deviation (SD), median, minimum (best) and maximum (worst) objective value in each optimization test case. It also gives the statistic α which indicates the significance level (either 0.05, 0.01) at which the performance of the proposed algorithm was better than that of the other algorithms, where an empty entry indicates that there was no statistically significant performance advantage. The α statistic was determined with the Mann–Whitney nonparametric test (Sheskin, 2007).

From the test results in follows that the proposed algorithm performed well as it obtained the best mean and median statistics in three out of four test functions, while achieving statistics which were very near best in other cases. Also, its performance had a statistically significant advantage in 13 out of 24 comparisons, namely over 50% of the cases, which further demonstrates its performance advantage. The proposed algorithm also achieved good SD scores which shows that its performance was robust across different test problems.

The test results highlight the performance gains of selecting an optimal ensemble during the search, as evident from the comparisons to the V1 and V2 variants, which used a fixed metamodel and a fixed ensemble, respectively.

Beyond the test statistics also the pattern of ensembles updates was examined to gauge if a single ensemble was dominant or whether multiple configurations were used. Accordingly, Fig. 3 shows plots of the different ensembles configurations which were selected during an optimization test with the Ackley-10D function and with the Rosenbrock-20D function. It follows that the Kriging metamodel was chosen relatively more often than others, but other ensembles were also used throughout, which shows that adaptively selecting the ensemble during the search improved the search effectiveness.

4.2 Engineering test problem

Beyond the tests with mathematical test functions also included was a test with an engineering simulation-driven problem, as it more closely represents a real-world problem. Here the goal was to generate an airfoil shape which maximizes its lift while minimizing its aerodynamic friction, namely, the drag, at the prescribed operating conditions. Candidate airfoils were represented with the method of Hicks and Henne (1978) so that an airfoil profile was defined as

$$y = y_b + \sum_{i=1}^{h} \alpha_i b_i(x)\,, \tag{3a}$$

$$b_i(x) = \left[\sin\left(\pi x^{\frac{\log(0.5)}{\log(i/(h+1))}}\right)\right]^4\,, \tag{3b}$$

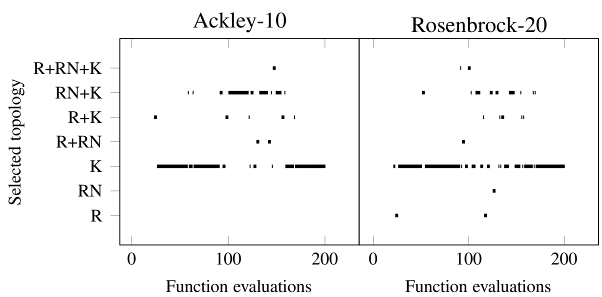

Figure 3. Selected ensemble topologies (R:RBF, RN:RBFN, K:Kriging) .

TABLE 3. TEST STATISTICS–MATHEMATICAL TEST FUNCTIONS

		Proposed	V1	V2	EA–PS	EI–CMA-ES
Ackley-10	Mean	**7.523e+00**	1.398e+01	1.198e+01	5.271e+00	1.781e+01
	SD	8.352e+00	5.037e+00	8.301e+00	5.943e-01	1.617e+00
	Median	**2.314e+00**	1.585e+01	1.249e+01	5.408e+00	1.793e+01
	Min(best)	1.894e-01	2.383e+00	3.457e+00	4.098e+00	1.443e+01
	Max(worst)	1.836e+01	1.780e+01	2.044e+01	6.010e+00	1.988e+01
	α			0.05		0.05
Rastrigin-5	Mean	**6.923e+00**	1.110e+01	9.172e+00	7.490e+00	2.371e+01
	SD	3.739e+00	9.495e+00	1.143e+01	5.677e+00	6.053e+00
	Median	6.629e+00	9.535e+00	**4.731e+00**	6.172e+00	2.390e+01
	Min(best)	2.028e+00	1.148e+00	3.647e+00	1.827e+00	1.459e+01
	Max(worst)	1.166e+01	2.975e+01	3.690e+01	1.703e+01	3.501e+01
	α					0.01
Rosenbrock-20	Mean	5.791e+02	1.011e+03	6.857e+02	**7.578e+02**	4.024e+03
	SD	2.228e+02	6.461e+02	2.805e+02	2.375e+02	1.049e+03
	Median	**5.956e+02**	7.944e+02	7.429e+02	7.012e+02	3.685e+03
	Min(best)	2.143e+02	5.483e+02	3.078e+02	4.676e+02	3.141e+03
	Max(worst)	8.905e+02	2.517e+03	1.184e+03	1.186e+03	6.144e+03
	α		0.05			0.01
Weierstrass-40	Mean	**2.946e+01**	4.222e+01	4.541e+01	3.224e+01	3.763e+01
	SD	4.740e+00	4.556e+00	3.357e+00	1.699e+00	1.542e+01
	Median	**2.651e+01**	4.341e+01	4.567e+01	3.221e+01	2.708e+01
	Min(best)	2.533e+01	3.470e+01	4.042e+01	3.030e+01	2.207e+01
	Max(worst)	3.586e+01	4.913e+01	4.970e+01	3.486e+01	5.935e+01
	α		0.01	0.01		

where y_b is a baseline profile, taken here to be the NACA0012 symmetric profile, b_i are geometric basis functions (Wu et al., 2003), and $\alpha_i \in [-0.01, 0.01]$ are weights to be calibrated, namely, the problem variables. To visualize the problem formulation, Fig. 4 shows the layout of the airfoil problem.

Two optimization scenarios were examined: i) a low dimensional case where each of the upper and lower airfoil profiles were defined by three basis functions, thereby resulting in a total of six design variables, and ii) a high dimensional case where 10 basis were used per profile, thereby resulting in a total of 20 design variables. The lift and drag coefficients of candidate airfoils were obtained by using *XFoil*, a computational fluid dynamics simulation for analysis of subsonic isolated airfoils (Drela and Youngren, 2001). Also incorporated was structural integrity requirement that the minimum airfoil thickness (t) between 20% to 80% of its chord line had to be at above a critical value of $t^{\star} = 0.1$. Accordingly, the objective function used was

$$f = -\frac{c_l}{c_d} + p\,, \quad p = \begin{cases} \dfrac{t^{\star}}{t} \cdot \left|\dfrac{c_l}{c_d}\right| & \text{if } t < \ t^{\star} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where p is a penalty for violation of the thickness constraint. The aircraft operating conditions were an altitude of 30,000 ft, a speed of Mach 0.7, namely 70% of the speed of sound, and an angle of attack (AOA) of 2° , which is roughly the angle between the airfoil

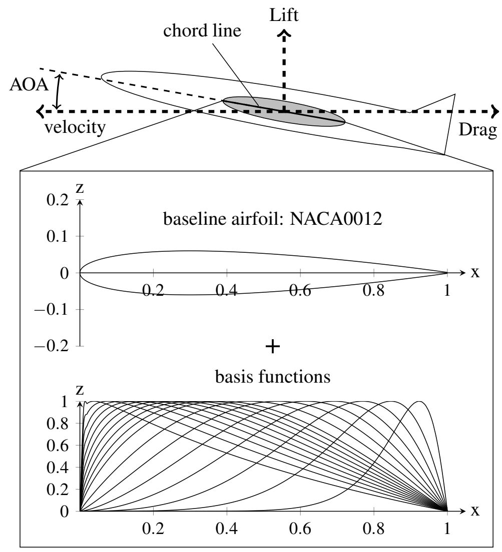

Figure 4. The layout of the airfoil optimization problem.

chord line and the aircraft velocity. These represent typical and common operating conditions.

The evaluation was performed following the setup of Section 4.1, and Table 4 gives the obtained test statistics. It follows that the trends observed with the mathematical functions persist also here as the implemented algorithm again performed well in comparison to the reference algorithms.

Also inline with Section 4.1, Fig. 5 shows the ensembles configurations which were selected during one run from the six dimensional case and one from the 20 dimensional case, respectively. As before, the configurations varied throughout the search. Overall the test statistics and the ensemble updates show that the optimal ensemble configuration is strongly problem dependant also varies during the search itself, and that selecting the optimal configuration improved the search effectiveness.

5 Conclusion

The extensive use of computer simulations in engineering and science has motivated the development of dedicated optimization techniques which perform more effectively in such settings. Specifically, metamodels are used to alleviate the high computational load associated with each simulation run. Since a variety of metamodel variants have been proposed and the optimal metamodel is problem dependant, ensembles try to address this by incorporating multiple metamodels simultaneously. Still, the optimal ensemble configuration is also problem dependant and is typically unknown prior to the actual optimization search. To address this issue this study has proposed an approach in which the ensemble configuration is selected continuously throughout the optimization search such that an optimal configuration is consistently used. The approach also operates within the TR framework to manage the metamodel and to ensure convergence to a valid solution in face of inherent metamodel inaccuracies. In an extensive performance analysis based on both mathematical test function and

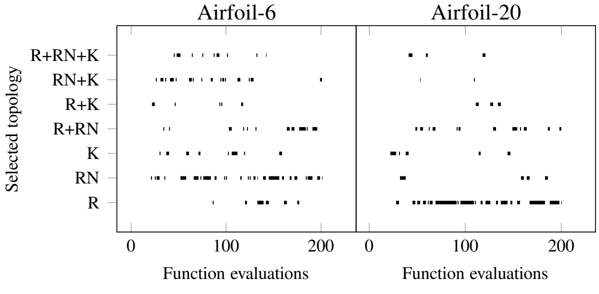

Figure 5. Ensemble configurations selected in two runs of the airfoil problem (R:RBF, RN:RBFN, K:Kriging).

a simulation-driven engineering problem the proposed approach was found to improve the search effectiveness, while the ensemble configuration continuously varied during the search, which highlight the necessity of this procedure. Overall, analysis shows that the proposed approach of continuously selecting an optimal ensemble configuration improved the search performance.

References:

D. Büche, N. N. Schraudolph, and P. Koumoutsakos, “Accelerating evolutionary algorithms with Gaussian process fitness function models,” *IEEE Transactions on Systems, Man, and Cybernetics–Part C*, vol. 35, no. 2, pp. 183–194, 2005.

A. R. Conn, K. Scheinberg, and P. L. Toint, “On the convergence of derivative-free methods for unconstrained optimization,” in *Approximation Theory and Optimization: Tributes to M.J.D. Powell*, A. Iserles and M. D. Buhmann, Eds. Cambridge; New York: Cambridge University Press, 1997, pp. 83–108.

A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust Region Methods*. Philadelphia, Pennsylvania: SIAM, 2000.

K. A. de Jong, *Evolutionary Computation:A Unified Approach*. Cambridge, Massachusetts: MIT Press, 2006.

M. Drela and H. Youngren, *XFOIL 6.9 User Primer*, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 2001.

T. Goel, R. T. Haftka, W. Shyy, and N. V. Queipo, “Ensembles of surrogates,” *Structural and Multidisciplinary Optimization*, vol. 33, pp. 199–216, 2007.

D. Gorissen, T. Dhaene, and F. De Turck, “Evolutionary model type selection for global surrogate modeling,” *The Journal of Machine Learning Research*, vol. 10, pp. 2039–2078, 2009.

R. M. Hicks and P. A. Henne, “Wing design by numerical optimization,” *Journal of Aircraft*, vol. 15, no. 7, pp. 407–412, 1978.

Y. Jin, M. Olhofer, and B. Sendhoff, “A framework for evolutionary optimization with approximate fitness functions,” *IEEE Transactions on evolutionary computation*, vol. 6, no. 5, pp. 481–494, 2002.

TABLE 4. TEST STATISTICS–AIRFOIL PROBLEM

		Proposed	V1	V2	EA–PS	EI–CMA-ES
6D	mean	**-8.297e+01**	-7.434e+01	-8.209e+01	-7.789e+01	-7.241e+01
	SD	1.373e+01	1.756e+00	2.514e+01	2.532e+00	6.341e-01
	median	-7.567e+01	-7.490e+01	**-7.586e+01**	-7.231e+01	-7.264e+01
	min(best)	-1.068e+02	-7.629e+01	-1.436e+02	-8.036e+01	-7.290e+01
	max(worst)	-7.488e+01	-7.174e+01	-6.405e+01	-7.238e+01	-7.099e+01
	α		0.05			0.01
20D	Mean	**-3.439e+00**	-3.387e+00	-3.417e+00	-3.428e+00	-3.409e+00
	SD	1.503e-01	1.035e-01	1.214e-01	1.393e-01	1.150e-01
	Median	**-3.434e+00**	-3.397e+00	-3.424e+00	-3.411e+00	-3.401e+00
	Min(best)	-3.685e+00	-3.512e+00	-3.744e+00	-3.619e+00	-3.558e+00
	Max(worst)	-3.157e+00	-3.254e+00	-3.344e+00	-3.243e+00	-3.264e+00
	α					

J. Muller and R. Piché, "Mixture surrogate models based on dempster-shafer theory for global optimization problems," *Journal of Global Optimization*, vol. 51, no. 1, pp. 79–104, 2011.

J. Muller and C. A. Shoemaker, "Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems," *Journal of Global Optimization*, vol. 60, pp. 123–144, 2014.

M. J. D. Powell, "On trust region methods for unconstrained minimization without derivatives," *Mathematical Programming, Series B*, vol. 97, pp. 605–623, 2003.

A. Ratle, "Optimal sampling strategies for learning a fitness model," in *The 1999 IEEE Congress on Evolutionary Computation–CEC 1999.* Piscataway, New Jersey: IEEE, 1999, pp. 2078–2085.

R. G. Regis and C. A. Shoemaker, "A quasi-multistart framework for global optimization of expensive functions using response surface models," *Journal of Global Optimization*, vol. 56, pp. 1719–1753, 2013.

D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, 4th ed. Boca Raton, Florida: Chapman and Hall, 2007.

P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," Nanyang Technological University, Singapore and Kanpur Genetic Algorithms Laboratory, Indian Institute of Technology Kanpur, India, Technical Report KanGAL 2005005, 2005.

Y. Tenne, "A computational intelligence algorithm for simulation-driven optimization problems," *Advances in Engineering Software*, vol. 47, pp. 62–71, 2012.

——, "An optimization algorithm employing multiple metamodels and optimizers," *International Journal of Automation and Computing*, vol. 10, no. 3, pp. 227–241, 2013.

——, "An algorithm for computationally expensive engineering optimization problems," *International Journal of General Systems*, vol. 42, no. 5, pp. 458–488, 2013.

Y. Tenne and C. K. Goh, Eds., *Computational Intelligence in Expensive Optimization Problems*, ser. Evolutionary Learning and Optimization. Berlin: Springer, 2010, vol. 2.

F. A. C. Viana, G. Venter, and V. Balabanov, "An algorithm for fast optimal Latin hypercube design of experiments," *International Journal of Numerical Methods in Engineering*, vol. 82, no. 2, pp. 135–156, 2009.

H.-Y. Wu, S. Yang, F. Liu, and H.-M. Tsai, "Comparison of three geometric representations of airfoils for aerodynamic optimization," in *Proceedings of the 16th AIAA Computational Fluid Dynamics Conference.* Reston, Virginia: American Institute of Aeronautics and Astronautics, 2003, AIAA 2003-4095.