Tenth International Conference on Computational Fluid Dynamics (ICCFD10), Barcelona, Spain, July 9-13, 2018

ICCFD10-059

An Efficient Multiobjective Global Optimization Algorithm for Expensive Aerodynamic Shape Optimizations

F. Deng[*] and N. Qin[**]
Corresponding author: fdeng@nuaa.edu.cn

* Nanjing University of Aeronautics & Astronautics, China
** The University of Sheffield, UK.

Abstract: A hybrid improvement function (also known as infill criterion) used for multiobjective efficient global optimization (EGO) algorithm has been proposed to solve the aerodynamic shape optimization problems with multiple objectives based on the high-fidelity CFD simulations, such as RANS-based solutions. The proposed algorithm has combined some of the ideas from two other multiobjective expected improvements (EI) known as statistical multiobjective EI and hypervolume-based EI and avoided the difficulty of high-dimensional integrals required by these two multiobjective EIs. Ten analytical test problems and two aerodynamic shape optimization problems with different characteristics have been tested. In comparison with the other two multiobjective EGO algorithms and a standard evolutionary multiobjective optimization algorithm in the commercial software, the present algorithm gives the best performance in most of the test cases and shows great potential in aerodynamic shape optimizations based on the expensive CFD simulations.

Keywords: Aerodynamic Shape Optimization, Multiobjective Optimization, Efficient Global Optimization, Computational Fluid Dynamics.

1 Introduction

Nowadays the aerodynamic shape optimizations based on the RANS solutions become more and more popular in the aeronautical industries. In comparison with the solutions based on the linear aerodynamic theory, the RANS solutions can give much more accurate results. However, solving the RANS equations for practical aerodynamic problems is still time-consuming and the aerodynamic shape optimizations usually require a large number of CFD solutions. To make the high-fidelity aerodynamic shape optimizations affordable, one way is to build some kinds of surrogate models and then perform the optimizations on these surrogate models. One of the widely used surrogate models is the Kriging model [1]. Since the accuracy of a Kriging model has to be validated before optimizations, lots of resources may be wasted during this validation process. By introducing an infill criterion called the expected improvement (EI), Jones et al. [2] has proposed an efficient global optimization algorithm (EGO). After an initial Kriging model is established, a series of sampling points can be continuously added based on the EI value until the optimum or the maximum iteration is reached. The EGO algorithm can automatically balance the local (exploitation) and global search (exploration) so that no explicit validations are required.

Since the EGO algorithm shows great success in the optimization problems with one objective, this idea was also generalized to deal with the multiobjective optimization problems. By introducing the weighted scalarizing function, Knowles [3] converted a multiobjective optimization problem into a number of single-objective optimization problems, which can be easily solved by the single-objective EGO algorithm. Zhang et al. [4] also proposed a similar algorithm which can generate multiple candidate points during one iteration. In essence, these algorithms do not change the classical single-objective EGO algorithm and the free parameters introduced by these algorithms have to be tuned for each specific problem. In terms of the statistical meaning of EI, Keane [5] has generalized the original EI to the statistical multiobjective EI and derived an analytical expression of two-objective EI. For problems with more than two objectives, so far no analytical expressions of multiobjective EI can be found in literature. Emmerich et al. [6] has generalized the original EI to the multiobjective EI in another way. They proposed to continuously improve the hypervolume constructed by the approximate Pareto front and a hypervolume-based EI is used to infill new sampling points. To calculate the hypervolume-based EI, a high-dimensional integral is required. For two-objective optimization problems, Emmerich et al. [7, 8] have worked out a direct method to calculate the values of the hypervolume-based EIs. However, for the optimization problems with more than two objectives, time-consuming numerical methods have to be employed.

In this paper, a hybrid figure of merit for multiobjective optimization problems is proposed. This hybrid infill criterion has combined some of the ideas from Keane's multiobjective EI [5] and hybervolume-based EI by Emmerich et al. [6]. Its analytical formula can be easily found for the problems with more than two objectives. Furthermore, it can give better results than other multiobjective optimization algorithms in the tests of a number of analytical cases and practical aerodynamic shape optimization cases.

2 Optimization Algorithm

2.1 Multiobjective Optimization Problem

The multiobjective optimization problem considered in this paper is defined as

$$\begin{aligned} &\min(\vec{f}(\vec{x})) \\ s.t.\quad &\vec{b}_l < \vec{x} < \vec{b}_u \end{aligned} \tag{1}$$

where $\vec{f}(\vec{x}) = (f_1, f_2, \cdots, f_k)$ is the objective vector and $\vec{x} = (x_1, x_2, \cdots, x_d)$ is the design vector, with upper and lower bounds represented as $\vec{b}_l$ and $\vec{b}_u$. In this paper $\vec{f}(\vec{x})$ and $\vec{x}$ are assumed to be continuous.

2.2 Single-objective EGO Algorithm

2.2.1 Kriging Model

Assuming there are n sampling points by using some kinds of space-filling sampling method [9], like the most common one called Latin hypercube sampling method, the predictor of the Kriging model at any point $\vec{x}$ can be written as

$$\hat{y}(\vec{x}) = \hat{\mu} + \psi' \Psi^{-1}(\vec{y} - \vec{1}\hat{\mu}) \tag{2}$$

where $\vec{y} = (y_1, y_2, \cdots, y_n)$ is the function vector of the sampling points $(\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n)$, and the mean $\hat{\mu}$ is found from

$$\hat{\mu} = \frac{\vec{1}' \Psi^{-1} \vec{y}}{\vec{1}' \Psi^{-1} \vec{1}} \tag{3}$$

Ψ is the correlation matrix and its corresponding component ψ is given by

$$\psi = \left[Cor(\vec{x}, \vec{x}_1) \quad \cdots \quad Cor(\vec{x}, \vec{x}_n) \right]' \tag{4}$$

where the correlation function is defined as

$$Cor(\vec{x}_i, \vec{x}_j) = \exp\left(-\sum_{l=1}^{d} \theta_l \left| x_{il} - x_{jl} \right|^{p_l} \right) + \lambda \delta_{ij} \tag{5}$$

where θ_l and p_l are called hyperparameters of the Kriging model, and λ is a regularization constant used to govern the degree of regression. In this paper, since the objective functions are relatively smooth, the hyperparameters p_l are fixed to be 2 and λ is set to be 10^{-6} to filter the possible numerical noise. The hyperparameters θ_l are tuned by maximizing the likelihood, which is defined as

$$\frac{1}{(2\pi)^{n/2} (\hat{\sigma}^2)^{n/2} \left|\Psi\right|^{1/2}} \exp\left[\frac{-(\vec{y} - \vec{1}\hat{\mu})' \Psi^{-1} (\vec{y} - \vec{1}\hat{\mu})}{2\hat{\sigma}^2} \right] \tag{6}$$

where the variance $\hat{\sigma}^2$ is given by

$$\hat{\sigma}^2 = \frac{(\vec{y} - \vec{1}\hat{\mu})' \Psi^{-1} (\vec{y} - \vec{1}\hat{\mu})}{n} \tag{7}$$

The accuracy of the model is measured by the mean squared error of the prediction, which is

$$\hat{s}^2(\vec{x}) = \hat{\sigma}^2 [\vec{1} - \psi \Psi^{-1} \psi + \frac{\vec{1} - \vec{1}' \Psi^{-1} \psi}{\vec{1}' \Psi^{-1} 1}] \tag{8}$$

As one can see from the above formulas, unlike other surrogate models, the prediction of a Kriging model consists of a certainty part $\hat{y}(\vec{x})$ as shown in equation (2) and an uncertainty part $\hat{s}^2(\vec{x})$ as shown in equation (8). In other words, the prediction of a Kriging model is a random variable, which forms the basis of the EGO algorithm.

2.2.2 Single-objective EI

Considering k=1 in equation (1), the problem becomes a single-objective optimization problem with bound constraints. In order to solve this optimization problem, Jones [2] has proposed an improve function defined as $I(\vec{x}) = y_{\min} - y(\vec{x})$, where $y_{\min} = \min(\vec{y})$ and $y(\vec{x})$ is the prediction of the Kriging model. Therefore, the search of the optimum can be automatically driven by finding a sampling point which can maximize $I(\vec{x})$. Since $y(\vec{x})$ is a random variable with a mean value of $\hat{y}(\vec{x})$ and a variance of $\hat{s}^2(\vec{x})$, $I(\vec{x})$ is also a random variable. Thus, the probability of improvement can be defined as

$$P(I(\vec{x})) = \Phi(\frac{y_{\min} - \hat{y}(\vec{x})}{\hat{s}(\vec{x})}) \tag{9}$$

and its corresponding EI can be given by

$$E[I(\vec{x})] = \begin{cases} (y_{\min} - \hat{y}(\vec{x}))\Phi(\frac{y_{\min} - \hat{y}(\vec{x})}{\hat{s}(\vec{x})}) + \hat{s}(\vec{x})\phi(\frac{y_{\min} - \hat{y}(\vec{x})}{\hat{s}(\vec{x})}) & \hat{s}(\vec{x}) > 0 \\ 0 & \hat{s}(\vec{x}) = 0 \end{cases} \tag{10}$$

where $\Phi()$ is the normalized Gaussian distribution function and $\phi()$ is the normalized Gaussian density function.

Figure 1 has explained the meanings of the probability of improvement and EI. The shade area represents the probability of improvement, which indicates the probability that a new sampling point $\vec{x}$ will improve the existing sampling points in the sense of searching the minimum, and the EI is the first moment of this shade area about the line through the $y_{\min}$, which measures how much an improvement will be by adding a new point $\vec{x}$ into the existing sampling points.

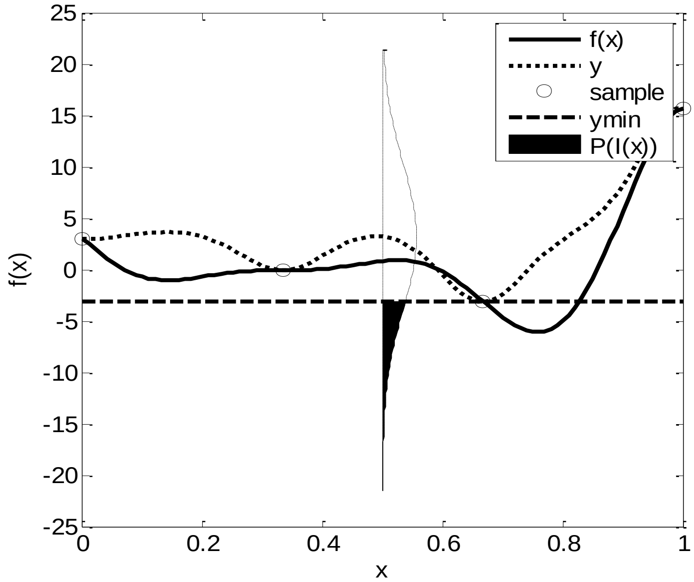

Figure 1: Probability of improvement and expected improvement (EI).

2.2.3 Single-objective EGO Algorithm

Then, the procedure of the EGO algorithm can be described as:

Algorithm I Single-objective EGO Algorithm

1) *Generate initial sampling points by one of the space-filling sampling methods;*
2) *Build a Kriging model by tuning the hyparameters in equation (5);*
3) *Find the optimum of the EI in equation (10) and add a new sampling point;*
4) *Check the convergence; if the converged requirements aren't satisfied, go back to step 2).*

2.3 Multiobjective EGO Algorithm

2.3.1 Statistical Multiobjective EI

Keane [5] has generalized the ideas of EI to address the multiobjective optimization problems. Here we consider the optimization problem with only two objectives given as $f_1(\vec{x})$ and $f_2(\vec{x})$. Given initial M_0 sampling points, we can construct an approximate Pareto front, which is

$$f_{1,2}^{*}=\left\{\left[f_1^{(1)*}(\vec{x}^{(1)}),\, f_2^{(1)*}(\vec{x}^{(1)})\right],\cdots,\left[f_1^{(M_0)*}(\vec{x}^{(M_0)}),\, f_2^{(M_0)*}(\vec{x}^{(M_0)})\right]\right\} \tag{11}$$

We can also build an independent Kriging model for each objectives denoted as $\hat{f}_1(\vec{x})$ and $\hat{f}_2(\vec{x})$. Then the joint probability density function can be written as

$$\phi(\hat{f}_1,\hat{f}_2)=\frac{1}{\hat{s}_1(\vec{x})\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\frac{[\hat{f}_1-\hat{\mu}_1(\vec{x})]^2}{\hat{s}_1^2(\vec{x})}\right\}\times\frac{1}{\hat{s}_2(\vec{x})\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\frac{[\hat{f}_2-\hat{\mu}_2(\vec{x})]^2}{\hat{s}_2^2(\vec{x})}\right\} \tag{12}$$

If we define the set which dominates the Pareto front $f_{1,2}^{*}$ as A_{nd}

$$A_{nd}=\left\{(f_1,f_2):(f_1,f_2)\succ(f_1^{(i)^*},f_2^{(i)^*}),\forall i\in 1,\cdots,M_0\right\} \tag{13}$$

Then the probability of improving the Pareto front is

$$P(I)=\iint_{(\hat{f}_1,\hat{f}_2)\in A_{nd}}\phi(\hat{f}_1,\hat{f}_2)d\hat{f}_1 d\hat{f}_2 \tag{14}$$

According to Keane [5], the EI for multiobjective problems can be written as

$$MOEI_{keane}=E[I]=P[I]\sqrt{\left(\bar{\hat{f}}_1-f_1^{(j)*}\right)^2+\left(\bar{\hat{f}}_2-f_2^{(j)*}\right)^2} \tag{15}$$

where ($\bar{\hat{f}}_1$, $\bar{\hat{f}}_2$) is the centroid of probability integral and ($f_1^{(j)*}$, $f_2^{(j)*}$) is the nearest point to the centroid. The definition of the centroid is

$$\begin{cases} \bar{\hat{f}}_1 = \iint_{(\hat{f}_1, \hat{f}_2) \in A_{nd}} \hat{f}_1 \phi(\hat{f}_1, \hat{f}_2) d\hat{f}_1 d\hat{f}_2 \, / \, P(I) \\ \bar{\hat{f}}_2 = \iint_{(\hat{f}_1, \hat{f}_2) \in A_{nd}} \hat{f}_2 \phi(\hat{f}_1, \hat{f}_2) d\hat{f}_1 d\hat{f}_2 \, / \, P(I) \end{cases} \tag{16}$$

A complicated analytical expression of $MOEI_{keane}$ for two-objective optimization problems has been derived by Keane [5]. However, no analytical expressions of $MOEI_{keane}$ for optimization problems with more than two objectives have been found so far. In this paper, the subroutines provided by Forrester et al. [10] are used to calculate the value of $MOEI_{keane}$.

2.3.2 Hypervolume-based EI

The concept of hypervolume for a Pareto front was proposed by Zitzler et al. [11]. As shown in Figure 2, the hypervolume of a Pareto front is defined as the shade area/volume between a reference point and the points in the Pareto front. It can be used to measure the quality of a Pareto front. Fonseca et al. [12] have figured out a fast method used to calculate the hypervolumes, and the subroutines developed by them have been employed in this study. Based on the concept of the hypervolume, Emmerich et al. [6] has generalized the EI for multiple objectives. If we assume that the current Pareto front is $f_{1,2}^{*}$ and the Pareto front after adding a new sampling point $\vec{x}$ is $f_{1,2}^{**}$, then the improvement based on their hypervolume difference can be defined as

$$I_H(\vec{x}) = I_H(f_{1,2}^{**}) - I_H(f_{1,2}^{*}) \tag{17}$$

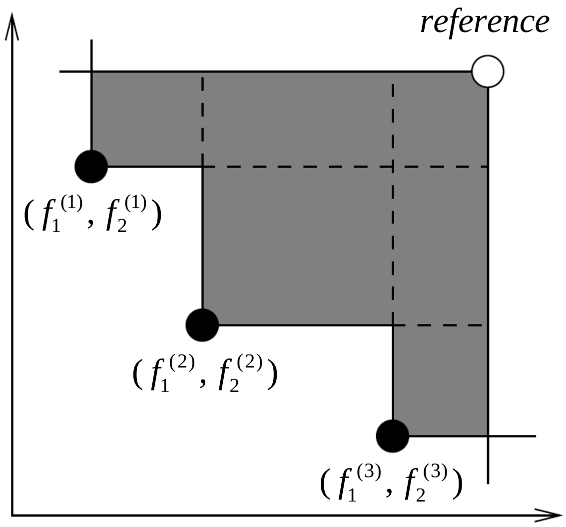

Figure 2: The definition of hypervolume.

Apparently, $(\hat{f}_1, \hat{f}_2)$ are random variables, so does $I_H(\vec{x})$. Naturally, the hypervolume-based EI can be written as

$$MOEI_{hv} = E(I_H(\vec{x})) \tag{18}$$

It is difficult to find an analytical expression of $MOEI_{hv}$, since $I_H(\vec{x})$ is a random variable and a high-dimensional integral is required. However, Emmerich et al. [7, 8] has managed to find a direct method to calculate its value while there are only two objectives. The subroutines developed by them are used in this study. For the optimization problems with more than two objectives, the Monte-Carlo integration method has to be adopted, although its cost is very high in comparison with the direction method. It can be written as

$$MOEI_{hv} \approx \frac{1}{m} \sum_{i=1}^{m} I_i \tag{19}$$

where m is the total number of sampling points. The value of 10000 is used in this study.

2.3.3 Hybrid EI

As seen in equation (18), the difficulty of calculating $E(I_H(\vec{x}))$ is because of the need to perform a high-dimensional integral for a random variable. Now if only the predictors of the Kriging models are used to calculate the hypervolume difference written as $I_H(\hat{f}_{1,2}^{**}) - I_H(\hat{f}_{1,2}^{*})$, the difficulty of calculating $E(I_H(\vec{x}))$ in equation (18) can be greatly reduced. Since the predictor of a Kriging model is the value with the highest probability close to the true value, the hypervolume difference based on the predicators can be used to measure how much improvement there will be if a new sampling point $\vec{x}$ is added.

On the other hand, if we assume the set of the squared points in Figure 3 is $f_{1,2}^{-}$, which contains the corner points used to calculate the hypervolume between a reference point and the current Pareto front represented as circles in Figure 3, the probability improvement to point i (the probability of being in the shade area) in the $f_{1,2}^{-}$ can be given by

$$P_i(\vec{x}) = \Phi(\frac{\mu_1^{(i)} - \hat{\mu}_1(\vec{x})}{\hat{s}_1(\vec{x})})\Phi(\frac{\mu_2^{(i)} - \hat{\mu}_2(\vec{x})}{\hat{s}_2(\vec{x})}) \tag{20}$$

By summing all of the contributions from the points in the set $f_{1,2}^{-}$, written as $\sum_{i=1}^{n} P_i(\vec{x})$, the probability of improvement to the current Pareto front by adding a new sampling point $\vec{x}$ can be measured. For k objectives, it can be easily generalized as

$$P_i(\vec{x}) = \prod_{j=1}^{k} \Phi(\frac{\hat{\mu}_j^{(i)} - \hat{\mu}_j(\vec{x})}{\hat{s}_j(\vec{x})}) \tag{21}$$

By combining the hypervolume difference based on the predicators, corresponding to the local search (exploitation) capability in the context of global optimizations, and the probability of improvement to the corner points used to calculate the hypervolume, corresponding to the global search (exploration) capability, the present hybrid EI can be written as

$$MOEI_{present} = \left(\sum_{i=1}^{n} P_i(\vec{x}) \right) \times \left[I_H(\hat{f}_{1,2}^{**}) - I_H(\hat{f}_{1,2}^{*}) \right] \tag{22}$$

In some sense, this hybrid EI has combined some of the ideas from Keane [5] and Emmerich et al. [6], and it has greatly improved the efficiency and capability of the multiobjective EGO algorithm as shown in the following tests. Furthermore, in comparison with equation (15) and (18), the present hybrid EI is much easier to be calculated and generalized to the optimization problems with more than two objectives.

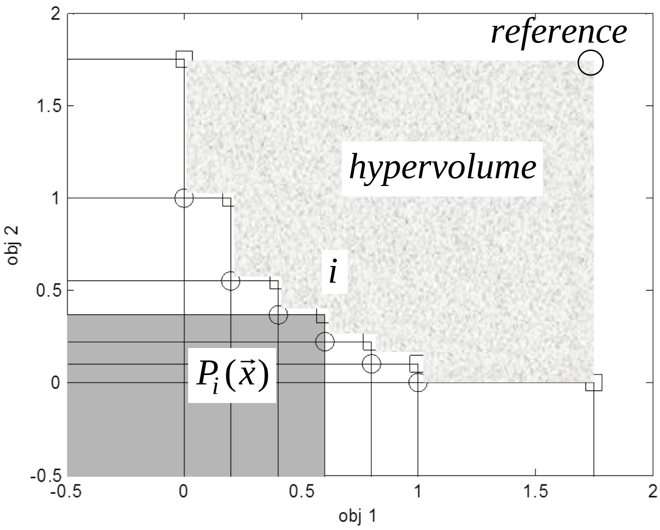

Figure 3: The definition of Hybrid EI.

2.3.4 Multiobjective EGO Algorithm

Similar to Algorithm I, the procedure of the multiobjective EGO algorithm can be described as:

Algorithm II Multiobjective EGO Algorithm

1) *Generate initial sampling points by one of space-filling sampling methods;*
2) *Build Kriging models for all of the objectives by tuning the hyparameters in equation (5);*
3) *Build an approximate Pareto front based on the existing data;*
4) *Find the optimum of the multiobjective improvement in equation (15) or (18)or (22);*
5) *Check the convergence; if the converged requirements aren't satisfied, go back to step 2).*

---

3 Results and Discussions

3.1 Analytical Test Problems

3.1.1 Performance Metrics

In order to compare the performance of different multiobjective algorithms, the inverted generational distance (*IGD*) metric and hypervolume difference (I_H^-) metric as suggested by Zhang [4] are adopted. The inverted generational distance is defined as

$$IGD(P^*, P) = \frac{\sum_{\upsilon \in P^*} d(\upsilon, P)}{\left|P^*\right|} \tag{23}$$

where *P* is the approximate Pareto front and P^* is the set of the exact Pareto front. $d(\upsilon, P)$ is the minimum distance between point υ and the points in *P*. If $\left|P^*\right|$ is large enough to represent the exact Pareto front, the IGD can be used to measure the accuracy of the approximate Pareto front *P*. Similarly, the hypervolume difference is defined as

$$I_H^-(P^*, P) = I_H(P^*) - I_H(P) \tag{24}$$

3.1.2 Experiment Results

In order to test different multiobjective optimization algorithms, ten analytical test problems with different characteristics are chosen as suggested by Ref. [3, 13-17]. The analytical expressions of these test problems can be found in Table 2 in the Appendix. In this paper, the word "gamultiobj" represents the multiobjective optimizer in the commercial software MATLAB [15], which solves the multiobjective optimization problems by an evolutionary algorithm. In the “gamultiobj” optimizer, the size of population for one generation is set as 30 and the maximum generation is set as 101. Thus, the total population used by “gamultiobj” is equal to 3030. For all of the multiobjective EGO algorithms, the size of initial sampling points is set as 11 and the maximum iteration is set as 89. Thus, the total number of sampling points is equal to 100, which is about 30 times less than that of the “gamultiobj” optimizer. Each experiment is repeated 50 times and the mean value and covariance are calculated.

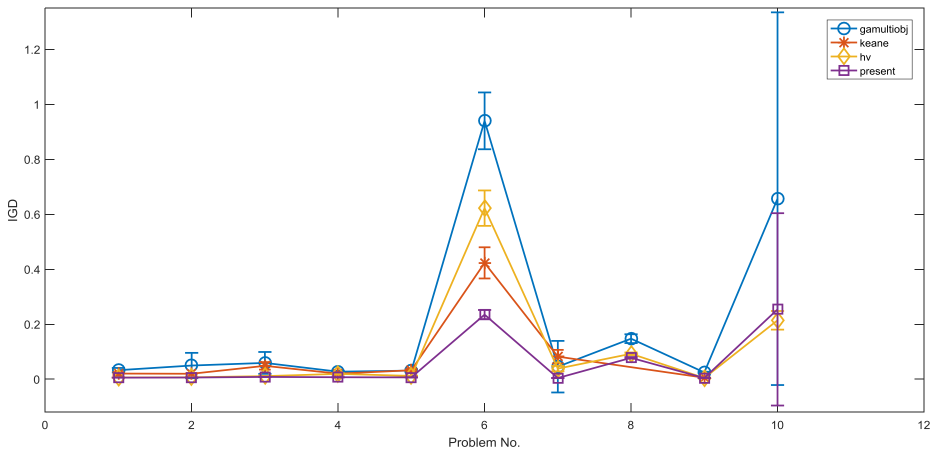

Figure 4: Comparison of inverted generational distances for test problems.

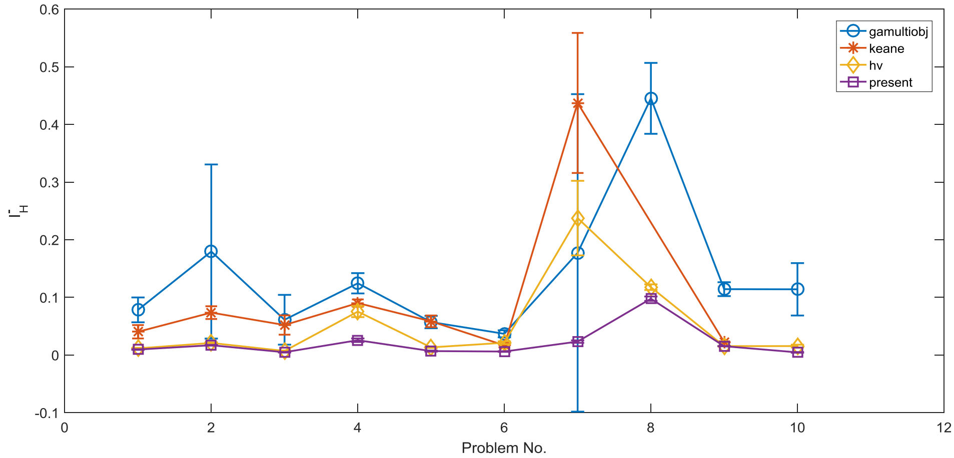

Figure 5: Comparison of hypervolume differences for test problems.

Figures 4 and 5 have compared the inverted generational distances and hypervolume differences of different algorithms. In these Figures and the Figures below, “keane” represents $MOEI_{keane}$ in equation (15), “hv” represents $MOEI_{hv}$ in equation (18) and “present” represents $MOEI_{present}$ in equation (22). For the optimization problems with three objectives (Problem No. 8 and 10), only “gamultiobj”, $MOEI_{hv}$ and $MOEI_{present}$ have been tested. It can be easily seen that the present algorithm is the winner for most of the cases. Figure 6 has compared the Pareto fronts by different algorithms in their last experiments. The fact that the present algorithm has the best performance has been clearly confirmed. Interestingly, although the “gamultiobj” optimizer has used about 30 times more points, its performance rarely matches that of the multiobjective EGO algorithms. This indicates that the costs of optimizing the multiobjective problems can be greatly reduced by the multiobjective EGO algorithms.

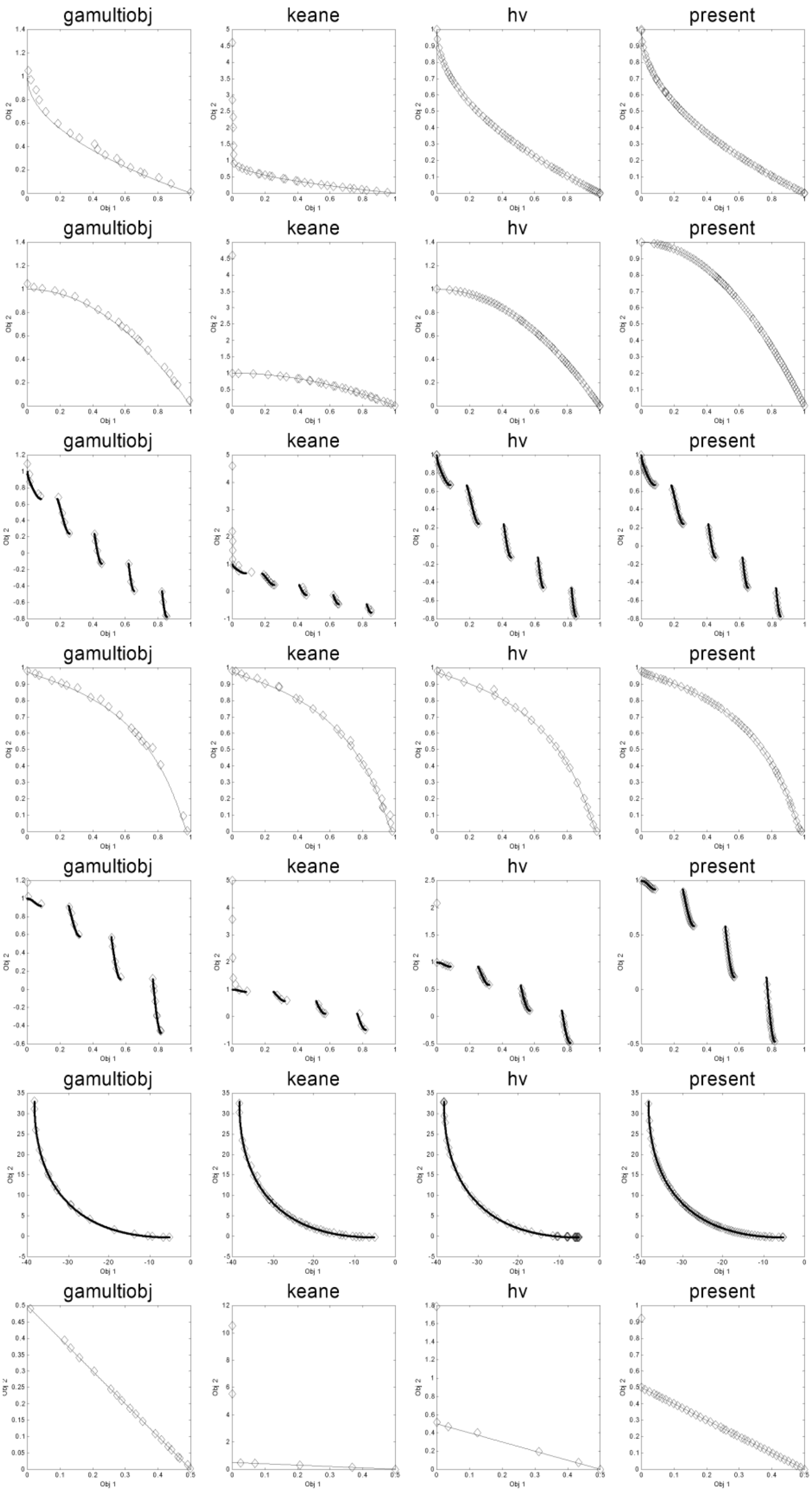

Figure 6: Comparison of Pareto fronts for test problems: The rows from top to bottom correspond to problem No.1-7

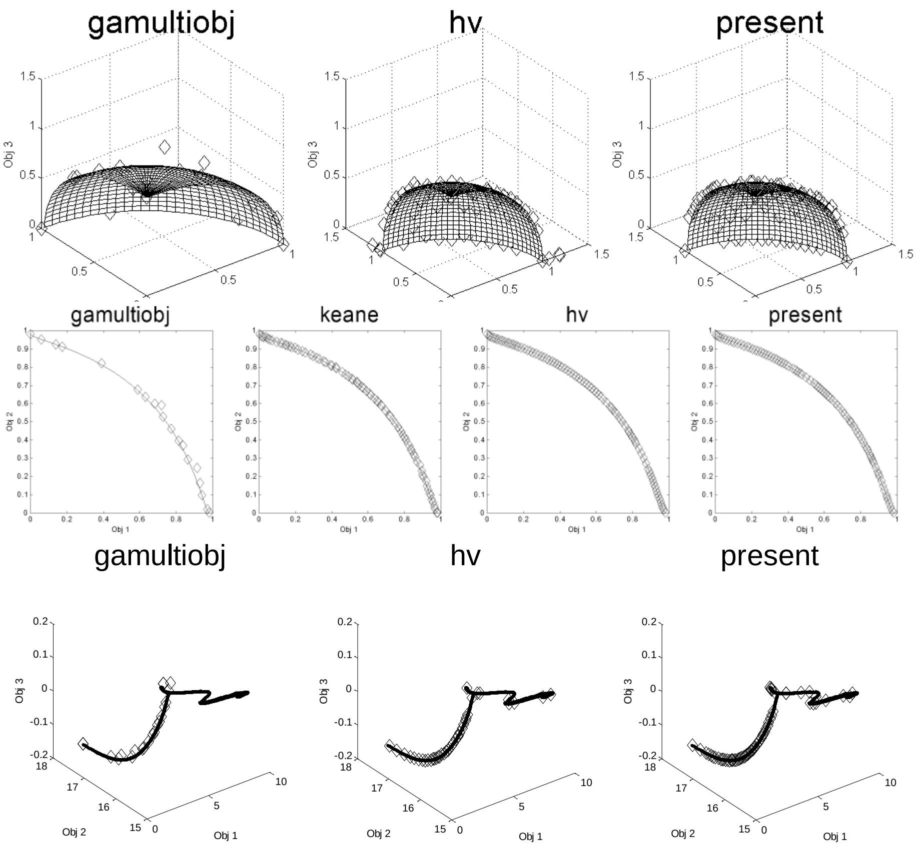

Figure 6 (continued): Comparison of Pareto fronts for test problems: The rows from top to bottom correspond to problem No.8-10

3.2 Aerodynamic Shape Optimizations

3.2.1 Shock Control Bump Optimization with Two Objectives

The first case for testing the performance of different algorithms in the aerodynamic shape optimizations is a shock control bump optimization on an airfoil called RAE5243. The parameterization of 2D shock control bump [18] with four design variables is shown in Figure 7. Based on our previous experiences, the bump length has been fixed as 0.3 of the chord length. Thus the remaining free design variables are bump crest location (x1), relative bump crest location (x2) and bump height (x3). The bumps are directly added on the upper surface of the RAE5243 airfoil.

The CFL3D v6.7 solver [19, 20] has been employed. A C-type mesh with 249×65 points has been generated and the average y^+ value on the first cell near the wall has been adjusted to O(1). Figure 8 and Table 1 has shown the mesh around the airfoil and the comparison of numerical results and experiment data [21, 22]. Different turbulence models have been tested and the Menter's SST turbulence model has been chosen because of its accuracy and robustness. A fast algebraic dynamic method [23] has been adopted to update the mesh during the optimization process.

The drag minimization problem at two different Mach numbers has been considered. The objective 1 is the drag coefficient at Mach 0.66, corresponding to a shockless flow condition. The objective 2 is the drag coefficient at Mach 0.70 and a strong normal shock wave will appear on the upper surface of the RAE5243 airfoil.

For comparisons, all of the multiobjective EGO algorithms including $MOEI_{keane}$, $MOEI_{hv}$ and $MOEI_{present}$ have adopted the same setting with 3 initial sampling points and 200 maximum sampling points. Two different parameter settings have been employed in the "gamultiobj" optimizer. The first setting denoted as "gamultiobj-fine" has 50 populations for each generation and 101 generations in total, and the other setting denoted as "gamultiobj-coarse" has 10 populations for each generation and 21 generations in total. Thus, "gamultiobj-coarse" has almost the same points as the multiobjective EGO algorithms.

The results have been shown in Figures 9, 10 and 11. It is clearly that the present algorithm gives the best results. By using only about 4% of the costs of the "gamultiobj-fine", the present algorithm has produced almost the same good Pareto front. In this case, the results of $MOEI_{keane}$ has better performance than that of $MOEI_{hv}$. The "gamultiobj-coarse" has the same number of points as the multiobjective EGO algorithms, but it produces the worst Pareto front. The star in Figure 9 represents the original location of the RAE5243 airfoil. It can be seen that both of the objectives have been improved and eventually a trade-off between two objectives need to be considered.

Table 1 Comparison of C_d while C_l=0.5154 for RAE5243 at Mach=0.6799, Re=18.68$\times 10^6$

Experiment [22]	BL	BL with DS modification	SA	SST
0.00877	0.00960	0.00911	0.00980	0.00912

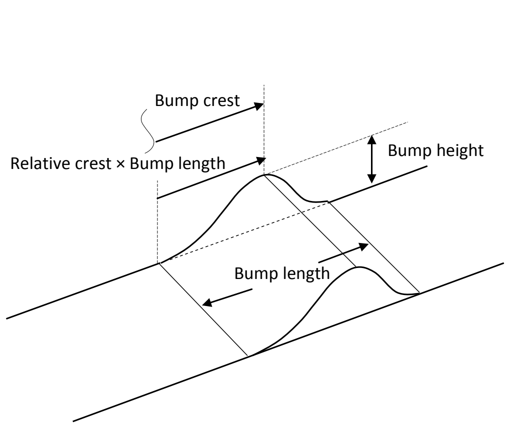

Figure 7: The parameterization of 2D shock control bump [18].

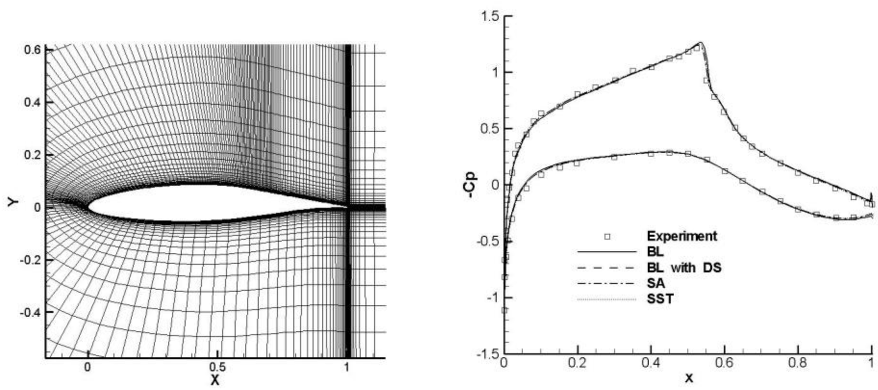

Figure 8: Mesh for RAE5243 and CFD validations: Mach=0.6799, Re=18.68$\times 10^6$, C_l=0.5154.

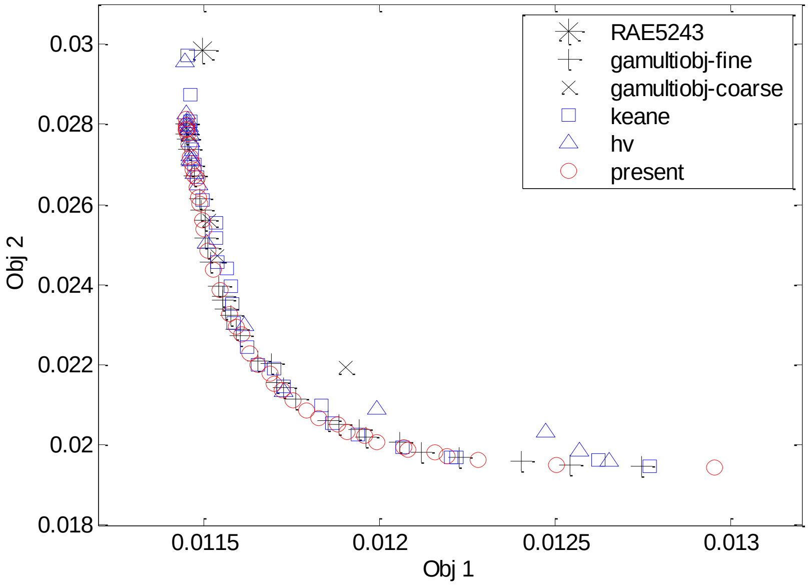

Figure 9: Comparison of different algorithms for shock control bump optimizations on the RAE5243 airfoil.

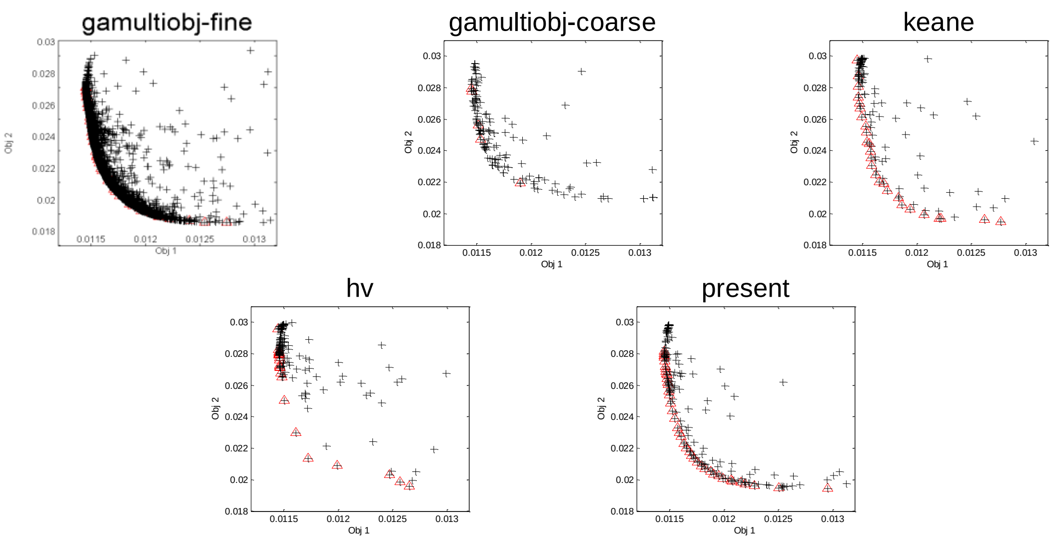

Figure 10: Comparison of Pareto fronts for shock control bump optimizations on the RAE5243 airfoil.

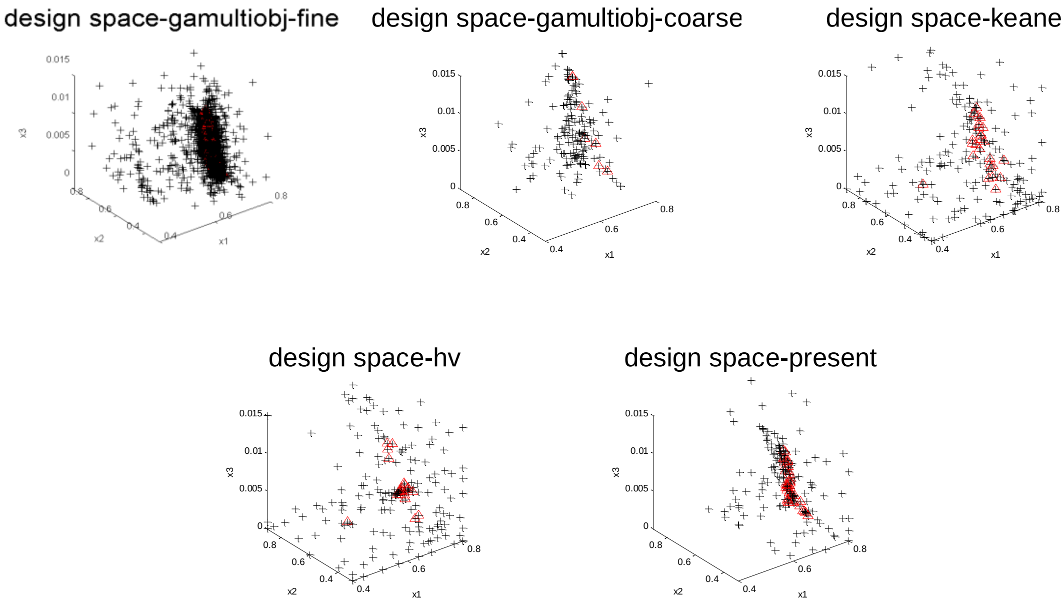

Figure 11: Comparison of design spaces for shock control bump optimizations on the RAE5243 airfoil.

3.2.2 Airfoil Optimization with Three Objectives

The airfoil optimization problem with three objectives has also been tested. The original airfoil is RAE2822, which is a typical supercritical airfoil. The CST parameterization method [24] is adopted and there are 8 design variables for changing the upper surface of this airfoil. In comparison with the previous case, this case has more design variables and objectives, providing more challenges to the multiobjective optimization algorithms. The mesh and CFD settings are the same as the previous case. As suggested by [25], the drag minimization problems at three different flow conditions are considered:

1) Objective 1：C_d when Ma=0.734, Re=6.5×10^6, C_l=0.7866；
2) Objective 2：C_d when Ma=0.754, Re=6.2×10^6, C_l=0.7478；
3) Objective 3：C_d when Ma=0.680, Re=5.7×10^6, C_l=0.5470。

In order to find the trade-offs among these flow conditions, three optimizers including “gamultiobj”, $MOEI_{hv}$ and $MOEI_{present}$ have been tested. The “gamultiobj” optimizer has 30 populations for each generation and 101 generations in total. Both of the multiobjective EGO optimizers have adopted the same setting with 3 initial sampling points and 250 maximum sampling points. The comparison of the Pareto fronts has been shown in Figure 12. Clearly, the good performance of the present algorithm has been confirmed again. It can also be seen that the “gamultiobj” optimizer, which requires large number of CFD calculations, shows no competitiveness in this case. By using the multiobjective optimizations, the trade-offs among three design points have been clearly revealed as shown in Figure 12. The information provided by the multiobjective optimizations can give engineers more insights into the design of airfoils flying at a wide range of flow conditions.

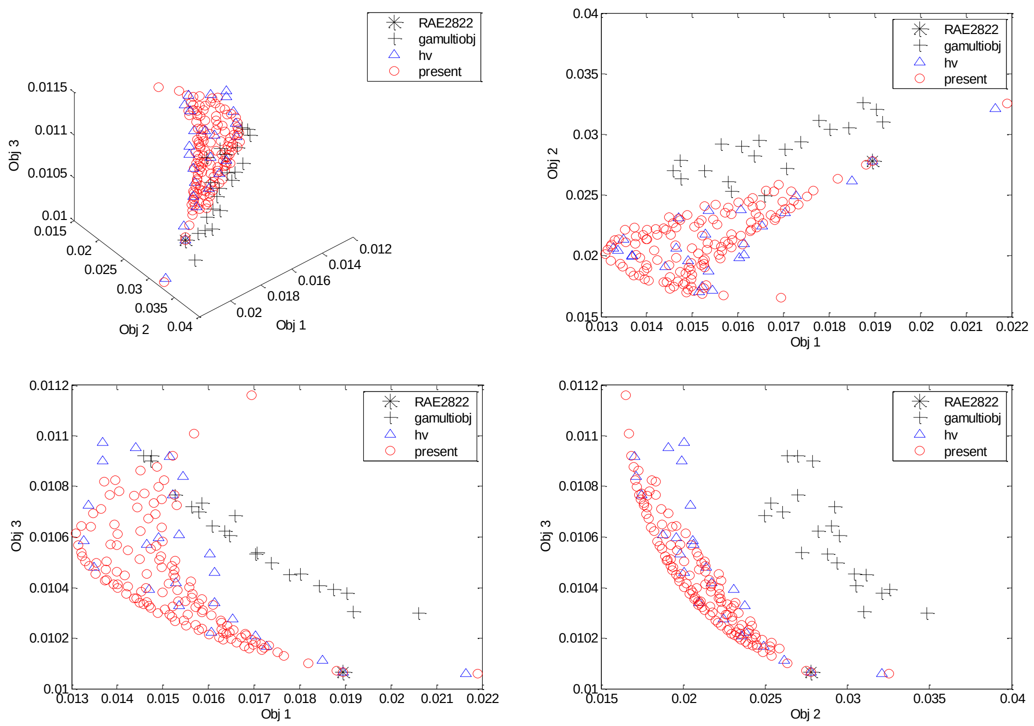

Figure 12: Comparison of Pareto fronts found by different algorithms for airfoil optimizations.

4 Conclusion

An efficient multiobjective global optimization algorithm with a hybrid improvement function has been proposed and tested in this paper. The present algorithm has combined some of the ideas from Keane's statistical multiobjective EI [5] and hypervolume-based EI by Emmerich et al. [6]. The proposed hybrid improvement function has avoided the difficulty of high-dimensional integrals and can be easily generalized to the optimization problems with more than two objectives. By solving ten analytical test problems and two aerodynamic shape optimization problems, the performance of the present algorithm has been compared with that of the other two multiobjective EGO algorithms and a multiobjective optimizer in the commercial software MATLAB, which stands for the state-of-the-art evolutionary multiobjective optimization algorithms. The tests show that all of the multiobjective EGO algorithms can produce better results than that of the evolutionary algorithm even the number of function calculations required by the evolutionary algorithm is much higher in some cases. Among three multiobjective EGO algorithms, the present hybrid improvement criterion gives the best performance, which has great potential in aerodynamic shape optimizations based on the expensive CFD simulations.

Acknowledgments

This work was funded by the project 11502112, 11432007 and 11672132 supported by National Natural Science Foundation of China. This work was also supported by a project funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions of China. The first author would also like to thank the funding from University of Sheffield pump priming research funds to support his research stay at the University of Sheffield.

References

[1] J. Sacks, W. J. Welch, T. J. Mitchell, et al. Design and analysis of computer experiments.

Statistical Science, 4: 409-423, 1989.
[2] D. Jones, M. Schonlau and W. Welch. Efficient global optimization of expensive black-box functions. Journal of Global Optimization, 13: 455-492, 1998.
[3] J. Knowles. ParEGO: A hybrid algorithm with one-line landscape approximation for expensive multiobjective optimization problems. IEEE Transactions on Evolutionary Computation, 10: 50-66, 2006.
[4] Q. Zhang, W. Liu, E. Tsang, et al. Expensive Multiobjective Optimization by MOEA/D with Gaussian Process Model. IEEE Trans on Evolutionary Computation, 14: 456-474, 2010.
[5] A. J. Keane. Statistical improvement criteria for use in multiobjective design optimization. AIAA Journal, 44: 879-891, 2006.
[6] M. Emmerich, K. Giannakoglou and B. Naujoks. Single and multi-objective evolutionary optimization assisted by Gaussian random field metamodels. IEEE Trans. Evol. Comput., 10: 421–439, 2006.
[7] M. Emmerich, A. Deutz and J. Klinkenberg. The computation of the expected improvement in dominated hypervolume of Pareto front approximations. LIACS-TR 9-2008, 2008.
[8] M. Emmerich, A. H. Deutz and J. W. Klinkenberg. The computation of the expected improvement in dominated hypervolume of Pareto front approximations, LIACS TR-4-2008, 2008.
[9] T. W. Simpson, A. J. Booker, D. Ghosh, et al. Approximation methods in multidisciplinary analysis and optimization：A panel discussion. Struct. Multidisc. Optim.，27：302~313，2004.
[10]A. I. J. Forrester, A. Sobester and A. J. Keane. Engineering Design via Surrogate Modelling - A Practical Guide. England: John Wiley & Sons, 2008.
[11]E. Zitzler and L. Thiele. Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study. Conference on Parallel Problem Solving from Nature (PPSN V), Amsterdam: 292~301, 1998.
[12]C. M. Fonseca, L. Paquete and M. López-Ibáñez. An Improved Dimension-Sweep Algorithm for the Hypervolume Indicator. 2006 IEEE Congress on Evolutionary Computation, Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada, 2006.
[13]K. Deb. Multiobjective genetic algorithms: Problem difficulties and construction of test problems. Evol. Comput., 7: 205~230, 1999.
[14]C. M. Fonseca and P. J. Fleming. An Overview of Evolutionary Algorithms in Multiobjective Optimization. Evolutionary Computation, 3: 1~16, 1995.
[15]The MathWorks Inc. Genetic Algorithm and Direct Search Toolbox Use with Matlab®: User's guide (Version 1).
[16]K. Deb, L. Thiele, M. Laumanns, et al. Scalable multiobjective optimization test problems. Proc. IEEE Congr. Evol. Comput., Honolulu, HI, 1: 825~830, 2002.
[17]D. A. VanVeldhuizen and G. B. Lamont. Multiobjective evolutionary algorithm test suites. Proc. Assoc. Comput. Mach. Symp. Appl. Comput., San Antonio, TX, Feb.: 351~357, 1999.
[18]W. S. Wong. Mechanisms and Optimisations of 3D Shock Control Bumps, PhD thesis. Sheffield: The University of Sheffield, 2006.
[19]A. J. Sclafani, M. A. DeHaan, J. C. Vassberg, et al. Drag Prediction for the Common Research Model Using CFL3D and OVERFLOW. Journal of Aircraft, 51: 1101-1117, 2014.
[20]Website: https://cfl3d.larc.nasa.gov/. (accessed on 17 June 2018)
[21]N. Qin, Y. Zhu and S. Shaw. Numerical study of active shock control for transonic aerodynamics. International Journal of Numerical Methods for Heat and Fluid Flow, 14: 444~466, 2004.
[22]J. L. Fulker and M. J. Simmons. An experimental study of shock control methods. DRA/AS/HWA/TR94007/1, 1994.
[23]F. Deng, N. Qin, X. Q. Liu, et al. Shock control bump optimization for a low sweep supercritical wing. Sci China Technol Sci, 56: 2385~2390, 2013.
[24]B. M. Kulfan. Universal Parametric Geometry Representation Method. Journal of Aircraft, 45: 142~158, 2008.
[25]J. Brezillon and N. R. Gauger. 2D and 3D aerodynamic shape optimisation using the adjoint approach. Aerospace Science and Technology, 8: 715~727, 2004.

Appendix

Table 2 Analytical Test Problems

No.	Name	Number of objectives	Number of DVs	Bounds	Expression
1	ZDT1 [13]	2	2	$x_i \in [0,1]$	$f_1 = x_1;\ f_2 = h \bullet g$ $g = 1 + 9\sum_{i=2}^{m} \frac{x_i}{m-1};\ h = 1 - \sqrt{f_1 / g}$
2	ZDT2 [13]	2	2	$x_i \in [0,1]$	$f_1 = x_1;\ f_2 = h \bullet g$ $g = 1 + 9\sum_{i=2}^{m} \frac{x_i}{m-1};\ h = 1 - (f_1 / g)^2$
3	ZDT3 [13]	2	2	$x_i \in [0,1]$	$f_1 = x_1;\ f_2 = h \bullet g$ $g = 1 + 9\sum_{i=2}^{m} \frac{x_i}{m-1}$ $h = 1 - \sqrt{f_1 / g} - \frac{f_1}{g}\sin(10\pi f_1)$
4	Fonseca [14]	2	3	$x_i \in [-4,4]$	$f_2 = 1 - \exp\left(-\sum_{i=1}^{m}\left(x_i + \frac{1}{\sqrt{m}}\right)^2\right)$ $f_1 = 1 - \exp\left(-\sum_{i=1}^{m}\left(x_i - \frac{1}{\sqrt{m}}\right)^2\right)$
5	Coello [3]	2	2	$x_i \in [0,1]$	$f_1 = x_1;\ f_2 = gh$ $g = 1 + 10x_2;\ h = 1 - \left(\frac{x_1}{g}\right)^2 - \frac{x_1}{g}\sin(8\pi x_1)$
6	Mat [15]	2	2	$x_i \in [-5,5]$	$f_1 = x_1^4 - 10x_1^2 + x_1x_2 + x_2^4 - x_1^2x_2^2$ $f_2 = x_2^4 - x_1^2x_2^2 + x_1^4 + x_1x_2$
7	DTLZ1 [16]	2	2	$x_i \in [0,1]$	$f_1 = x_1(1+g)/2;\ f_2 = (1-x_1)(1+g)/2$ $g = 100[1+h];$ $h = \sum_{i \in 2,\cdots,m}\begin{pmatrix}(x_i - 0.5)^2 \\ -\cos(2\pi(x_i - 0.5))\end{pmatrix}$
8	DTLZ2 [16]	3	2	$x_i \in [0,1]$	$f_1 = (1+g)\cos(\theta_1)\cos(\theta_2)$ $f_2 = (1+g)\cos(\theta_1)\sin(\theta_2)$ $f_3 = (1+g)\sin(\theta_1)$ $g = \sum_{i \in 3,\cdots,m}(x_i - 0.5)^2;$ $\theta_1 = x_1\pi/2;\ \theta_2 = x_2\pi/2$
9	VLMOP2 [17]	2	2	$x_i \in [-2,2]$	$f_1 = 1 - \exp(-\sum_{i=1}^{2}(x_i - 1/\sqrt{n})^2)$ $f_2 = 1 - \exp(-\sum_{i=1}^{2}(x_i + 1/\sqrt{n})^2)$
10	VLMOP3 [17]	3	2	$x_i \in [-3,3]$	$f_1 = 0.5(x_1^2 + x_2^2) + \sin(x_1^2 + x_2^2)$ $f_2 = \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15$ $f_3 = \frac{1}{(x_1^2 + x_2^2 + 1)} - 1.1\exp(-x_1^2 - x_2^2)$