

Multi-Objective Aerodynamic Optimization Using Hybrid Evolutionary-Adaptive Local Search Strategy[†]

HyeonWook Lim* and Hyoungjin Kim*
Corresponding author: hyoungjin.kim@khu.ac.kr

* Kyung Hee University, Republic of Korea

Abstract: A novel hybrid evolutionary multi-objective algorithms (EMOAs) - adaptive local search method was applied to multi-objective aerodynamic optimization problems for convergence improvement. The directional operator is comprised of selection of search direction and a one-dimensional search. Probability of the directional operator is adaptively changed based on the relative effectiveness of the directional local search operator and evolutionary operators such as crossover and mutation. The adaptive directional operator is combined with a baseline EMOA, NSGA-II. Multi-objective airfoil design optimization examples are defined as drag minimization or lift maximization and L/D maximization in cruise and high lift conditions. Results show that the present adaptive local search strategy enables significant enhancement of convergence when a local search is effective, while minimizing unnecessary computation for cases that a local search is not well suited for. Statistical test confirms the superiority of the hybrid method.

Keywords: Aerodynamic Design, Multi-Objective Optimization, Evolutionary Algorithms, Directional Search, Adaptive Local Search

1. Introduction

Multi-objective optimization methods are getting more attention than before as single-objective optimization methodologies are maturing, and trade-off between conflicting objectives is becoming more critical in modern engineering problems of Multi-disciplinary Design Optimization (MDO).

Evolutionary algorithms (EAs) are search algorithms based on genetic evolution and natural selection. EAs are well suited to Multi-Objective Optimization Problems (MOOPs) because they are based on population rather than single solution and therefore are naturally adequate to generate distributed solutions on the non-dominated Pareto front.

A well-known drawback of an EA is its slow convergence to the optimum solution, especially in regions near the optimum. This behavior is still true for evolutionary multi-objective algorithms (EMOAs) [1-6]. To cope with this problem, hybrid methods combining EMOAs and a local search method have been reported.[7-13] The hybrid methods are also referred to as memetic algorithms. [12]

Local search methods for the hybrid EMOAs can be put into two categories: neighborhood-based local searches and directional local searches. Neighborhood-based local search schemes generate perturbed solutions around a baseline solution and try to find better solutions than the baseline. The directional local search methods conduct a local search along a search direction, which is usually

[†]This paper is dedicated to the memory of the late Dr. Meng-Sing Liou.

determined by sensitivity information of objective functions with respect to decision variables. So, the effectiveness of the gradient-based directional local searches depends on the cost of the calculation of the gradient information. Refer to Ref.[13] for a comprehensive review on hybrid methods.

One of major issues for the hybrid methods is that we do not know *a priori* if conducting local search would be beneficial for a given specific optimization problem; it would be effective for uni-modal problems or in the last stage of optimization process near the true optimum. However, it might be waste of computational budget for multi-modal problems or in the early stages of the evolution process, where exploratory search is more needed for a successful optimization. Effective use of evolutionary and local directional search operators within available computational budget can be implemented by optimum selection of probability for the local search operator, p_{ls} , which determines the number of individuals going through the local search operator [14].

In our previous work [13], we developed a novel hybrid MOEA - adaptive local search method to enhance the convergence of MOEAs by combining an efficient directional local search method with a baseline MOEA. To manage the computational budget efficiently, we adaptively adjust a probability for the local search operator based on the relative effectiveness of the local search with respect to the evolutionary operators. The probability is increased if the local search is effective and reduced if the local search is not working relatively well compared to evolutionary operators. The adaptive probability depends on the performances of evolutionary and local search operators, which are affected by property of the problem at hand and convergence stages of evolution processes. The proposed algorithm was tested on several analytic test functions with two and three objective functions and 10 to 100 decision variables for quantitative evaluation of performances. In the present study, the hybrid method in Ref.[13] is applied to multi-objective airfoil design optimization problems to show its validity for real engineering applications. Two-objective airfoil design problems are defined at subsonic and transonic regimes. One objective is drag minimization or lift maximization and the other objective is L/D maximization. For subsonic flow conditions, the XFOIL code was used and for transonic flow simulations two-dimensional Reynolds-averaged Navier-Stokes solver was adopted. Statistical test is also performed using the Pareto-front solutions from multiple runs of the optimization.

The remainder of this paper is composed as follows: In Section 2, we briefly describe the baseline MOEA for the hybrid method. Followed are details of the directional operator in Section 3. The adaptive strategy for the hybrid method is summarized in Section 4. Numerical results on multi-objective airfoil design optimization are presented in Section 5. Finally Section 6 presents concluding remarks.

2. Baseline Multi-Objective Evolutionary Algorithms

2.1 Definition of Multi-Objective Optimization Problems

Multi-objective optimization problems (MOOPs) are defined in their general form as follows:

$$\begin{aligned} & \text{Minimize } (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})) \\ & \text{Subject to } g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, J \\ & \quad \quad \quad x_i^L \leq x_i \leq x_i^U, \quad i = 1, 2, \dots, N_{dv}, \end{aligned} \quad (1)$$

where M , J , and N_{dv} are the number of objective functions, inequality constraints, and decision variables, respectively. x^L and x^U are the lower and upper bounds of decision variables. In this study, objective functions, constraint functions and decision variables are all real numbers.

The dominance relation between solutions of MOOPs is defined by the following two conditions:

Solution a dominates solution b in minimization problems if

- 1) $f_j(a) \leq f_j(b)$ for all $j \in \{1, \dots, M\}$
- 2) $f_k(a) < f_k(b)$ for at least one $k \in \{1, \dots, M\}$

In other words, solution a dominates solution b if a is not worse than b in all objectives and a is strictly better than b in at least one objective; this is written symbolically as “ $a < b$.” Similarly, if a is dominated by b , it is represented as “ $a > b$.” A solution that is not dominated by any other solutions is said to be “non-dominated”; the nondominated solutions form the Pareto optimal set.

2.2 Baseline EMO algorithms

In this paper, NSGA-II and MOGA were used as baseline EMO algorithms. Since we are interested in design problems with real-valued decision variables, we employ real-parameter operators such as the simulated binary crossover (SBX) [5] for the crossover and the polynomial mutation [5] as the mutation operator for both algorithms. The evolutionary operators are applied to each design variables of parents with probability parameters p_c and p_m for the crossover and mutation operators respectively.

2.2.1 NSGA-II

NSGA-II (nondominated sorting genetic algorithm II) [2] is used in the first design examples in this study as a baseline EMO algorithm because of its popularity and widespread usage in engineering MDO applications. NSGA-II is an elitism approach utilizing the fast nondominated sorting. The crowded distance sorting is used for diversity preservation. Evolutionary operators such as crossover and mutation are conducted to generate N_{pop} children from the parents. Here N_{pop} is the population size. N_{pop} children and N_{pop} parents are combined to build a set of size $2N_{pop}$, which undergoes the reduction process. Fig. 1(a) shows the main procedure of NSGA-II. In Fig. 1, P stands for parents set of size N_{pop} , Q stands for children set of size N_{pop} , and $R = P \cup Q$ is the combined set of parents and children. t is for generation number.

2.2.2 MOGA

In design examples in this study, a baseline EMO algorithm for the hybrid method is a modified MOGA (Multiobjective Genetic Algorithm) with a new fitness sharing [16], which consists of the Pareto ranking, a novel fitness sharing method and an elite preservation approach. In MOGA, with a population size of N_{pop} , N_{pop} mating solutions are selected from N_{pop} parents by the stochastic universal sampling. For an elite preservation, all nondominated solutions are preserved in a separate archive and dominated solutions are removed from the archive. The overall procedure for MOGA is depicted in Fig.2.

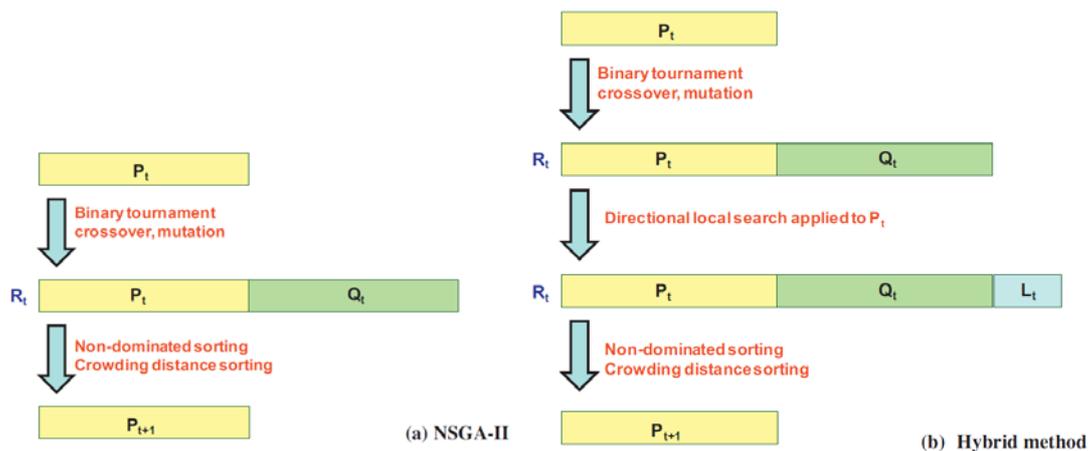


Figure 1: Procedure of solution set management for NSGA-II and its hybrid EMOA

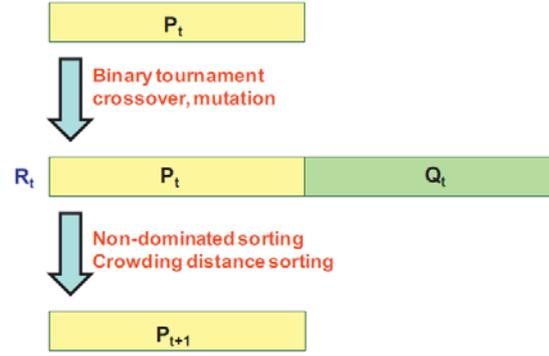


Figure 2: Procedure of solution set management for MOGA.

3. DIRECTIONAL LOCAL SEARCH OPERATOR

For the directional local search in the hybrid EMOA, we define a composite objective function Z , which is a sum of normalized objectives, to be minimized by the local search:

$$Z_i = \sum_{m=1}^M \frac{f_m(x_i) - f_m^{min}}{f_m^{max} - f_m^{min}}, \quad (2)$$

where f_m^{max} and f_m^{min} are the maximum and minimum objective function values of the m^{th} objective function for the current non-dominated solutions in the population, and \mathbf{x}_i is the design vector for solution i .

In multi-objective problems, a composite or aggregated objective function approach may have difficulties in finding solutions on a nonconvex Pareto front [5, pp.55-56]. In general, however, directional search methods do not suffer from the nonconvexity issue: If an initial solution lies off the nonconvex Pareto front and search direction is properly selected, application of a directional search can find a solution on the concave part of the Pareto front.

The directional local search operator proposed in this study comprises two steps: the determination of search direction and the local search along the search direction in order to minimize the composite objective function in Eq. (2).

3.1 Determination of search direction

Gradient calculation of an objective function by using a finite differencing requires N_{dv} function analyses. Therefore computational cost for the brute-force calculation of function gradient easily becomes prohibitively large for design problems with costly objective and/or constraint functions and many number of design variables. Therefore, it would be beneficial to determine search directions without explicit computation of objective function gradients. In this study, the algorithms suggested in Ref.[13] is followed.

Firstly, we build stencils for each starting solution of the local line search making use of information of existing solutions obtained in the optimization procedure. A stencil set S is defined as a set of neighbor solutions to be used for approximation of a search direction for an initial solution \mathbf{x}_0 :

$$S(\mathbf{x}_0, r) = \{ \mathbf{x} | \mathbf{x} \in T \text{ and } (x_{0i} - r \leq x_i \leq x_{0i} + r, \forall i = 1, \dots, n) \text{ and } (\mathbf{x} < \mathbf{x}_0 \text{ or } \mathbf{x} > \mathbf{x}_0) \} \quad (3)$$

where T is a set of all individuals occurred in the evolution history. The smaller the radius r is, the more accurate the resulting search directions would be. However, it is not guaranteed whether one could find sufficient number of solutions within the specified radius to perform the directional search. Specifying a radius small enough for accurate search directions and large enough to have at least one stencil could be difficult and problem dependent.

In this study, stencils are selected from existing solutions in zone of dominance of the objective space. In other words, solutions that dominate or are dominated by the initial solution point are considered to be included in the stencil. The stencil radius r is set as 1.0. Since all the decision variables are normalized into a domain of $[0, 1]$ in the present optimization, radius of 1.0 means the there is no constrained radius for S . In our previous work [13], we conducted numerical experiments with three different values of r : 0.1, 0.3 and 1.0, and found that $r = 1.0$ gives satisfactory results

To select a limited number of stencil solutions, we put the latest solutions in the design history into S until $|S|_{\max}$ is reached. $|S|_{\max}$ is set to be the same as the number of decision variables N_{dv} in this study.

Calculation of a centroid of stencil solutions is performed by a weighted summation of decision variable vectors using the following relation:

$$\mathbf{x}_c = \frac{\sum_{j=1}^{N_{stencil}} (w_{0j} \mathbf{x}_j)}{\sum_{j=1}^{N_{stencil}} w_{0j}}. \quad (4)$$

The weighting factor w_{0j} is defined as the inverse of Euclidean distance in the design space between solutions 0 and j in order to emphasize genotypically closer stencil solutions in the weighted sum.

$$w_{0j} = \frac{1}{|\mathbf{x}_j - \mathbf{x}_0|} \quad (5)$$

When the centroid \mathbf{x}_c is calculated, the closest solution to the centroid is selected from S , and is newly set as \mathbf{x}_c . Once \mathbf{x}_c is determined in S , a line search is conducted for Z in the direction connecting \mathbf{x}_c and \mathbf{x}_0 with a scalar parameter α as depicted in Fig.4.

3.2 Line Search

Once \mathbf{x}_m and \mathbf{x}_l are determined, a new solution \mathbf{x}_r is selected as $\mathbf{x}_r = \mathbf{x}_m + (\mathbf{x}_m - \mathbf{x}_l)$, and function values are calculated at \mathbf{x}_r . If the following condition for convexity is satisfied for the three solutions as illustrated in Fig. 3,

$$\frac{Z(\alpha_m) - Z(\alpha_l)}{\alpha_m - \alpha_l} < \frac{Z(\alpha_r) - Z(\alpha_m)}{\alpha_r - \alpha_m}, \quad (6)$$

the objective function Z is approximated as a quadratic polynomial of α :

$$Z(\alpha) = a\alpha^2 + b\alpha + c. \quad (7)$$

The coefficients a , b , and c in Eq. (7) are calculated by the three function values, and the second function call is made for $Z(\alpha^*)$ with the estimated optimum position $\alpha^* = -b/2a$. (See Fig.3 (c))

If the convexity condition in Eq. (6) is not satisfied, the step size is doubled to find a new \mathbf{x}_r , and the convexity condition is checked again. This procedure is repeated for N_{max} times as shown in Fig.4 (b). In this study, N_{max} is set as 1, which means the total computational cost in a line search is at most two function calls.

When the composite function Z shows a convex behavior, the final output of the local search is determined as follows:

$$Z_{opt} = Z(\alpha^*) \quad (8)$$

If the composite objective is not convex in the line search procedure, the computational cost for the line search is one function call, and the final output function of the local search is determined as the best solution found in the line search procedure:

$$Z_{opt} = \min(Z(\alpha_m), Z(\alpha_r)) \quad (9)$$

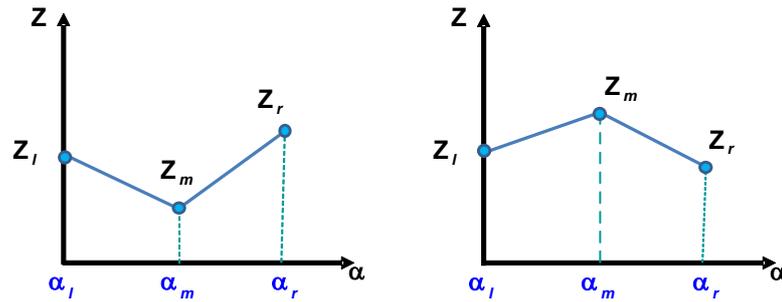
The improvement made by the local search in terms of Z is measured by the difference between initial and final values of Z :

$$imp_{ls} = Z_0 - Z_{opt} \quad (10)$$

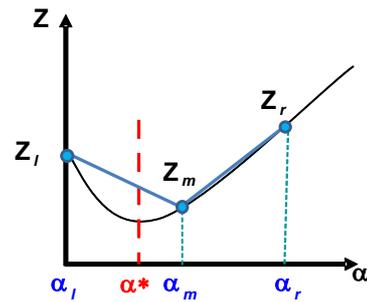
Regardless of the value of imp_{ls} , all the newly computed solutions during the local search are added to the combined set of parents and children solutions to undergo the ranking and selection process. The local search operator is applied to each of the N parents with a probability of p_{ls} . Then roughly $N \times \bar{c}_{ls} \times p_{ls}$ new solutions are added to the combined $2N$ solutions to join the reduction process for best- N solutions as illustrated in Fig. 1, which shows the procedure of solution set management for the hybrid EMOA-local search method. Although Fig.5 is depicted for NSGA-II, the procedure for any other baseline EMOA, e.g. MOGA, is very similar.

A new parameter C_{ls} , the computational cost of a line search, is defined for a later use in an adaptive local search strategy as the number of additional function evaluations needed to conduct a

line search. In most cases, two evaluations of objective functions are required for each line search. Therefore, \bar{C}_{ls} in Table 1 is set equal to two, as a reference number of function calls of a line search.



(a) Convex distribution of three points (b) Concave distribution of three points



(c) Quadratic fitting and optimal point for convex three points

Figure 3: Line search using a three-point quadratic fitting

3.3 Constraint handling in the directional search

For constrained problems, stencils for gradient calculation are sought among feasible solutions only. Also, optimal solutions in the line search procedure are found in the feasible region. If solutions with $\alpha=1$ or $\alpha= \alpha^*$ are infeasible for a geometric constraint, a nearest boundary of the feasible region is found by an iterative procedure. If the flow field for the solution with $\alpha=1$ does not converge, the local line search is stopped. Otherwise, another function call is made for $\alpha= \alpha^*$. If the flow field for the solution with $\alpha= \alpha^*$ does not converge, $Z(\alpha_l)$ is returned as an output of the local search.

```

Algorithm eLS
Input:  $\mathbf{x}_0, r$ 
Output:  $\mathbf{x}_0$ , line search solution set L added to R
1: Build  $S(\mathbf{x}_0, r)$ 
2: If  $|S| > 0$  Then
3:   Find a centroid solution  $\mathbf{x}_c$  in  $S$ 
4:   Conduct LineSearch_eLS
5:    $R := R \cup L$ 
6: Eneif

```

(a) Overall procedure of the present local search method

```

Algorithm LineSearch_eLS
Input:  $\mathbf{x}_0, \mathbf{x}_c, Z_0, Z_c, N_{\max}$ 
Output:  $\mathbf{x}_{\text{opt}}$ , line search solution set L
1: If  $\mathbf{x}_c < \mathbf{x}_0$  Then
2:    $\mathbf{x}_l := \mathbf{x}_0; \mathbf{x}_m := \mathbf{x}_c$ 
3: Else
4:    $\mathbf{x}_l := \mathbf{x}_c; \mathbf{x}_m := \mathbf{x}_0$ 

```

```

5: Endif
6:  $\mathbf{x}_r := \mathbf{x}_m + (\mathbf{x}_m - \mathbf{x}_l)$ 
7:  $\alpha_l := 0; \alpha_m := 1; \alpha_r := 2$ 
8:  $L := \{\mathbf{x}_r\}$ 
9:  $j := 1$ 
10: While  $j \leq N_{\max}$ 
11:    $\mathbf{x}_{\text{opt}} = \mathbf{x}_r$ 
12:   Check convexity of  $Z$  at  $\mathbf{x}_l$ ,  $\mathbf{x}_m$ , and  $\mathbf{x}_r$ 
13:   If  $Z$  is convex then
14:     Find an optimal step size  $\alpha^*$ 
15:      $\mathbf{x}_{\text{opt}} = \mathbf{x}_l + \alpha^*(\mathbf{x}_m - \mathbf{x}_l)$ 
16:      $L := L \cup \{\mathbf{x}_{\text{opt}}\}$ 
17:      $j := N_{\max} + 1$ 
18:   Else
19:      $\alpha_r := 3$ 
20:      $\mathbf{x}_l = \mathbf{x}_m; \mathbf{x}_m = \mathbf{x}_r; \mathbf{x}_r = \mathbf{x}_r + 2(\mathbf{x}_m - \mathbf{x}_l)$ 
21:      $L := L \cup \{\mathbf{x}_r\}$ 
22:      $j = j + 1$ 
23:   Endif
24: End While

```

(b) Procedure of LineSearch_eLS
Figure 4: Present directional local search method eLS

4. ADAPTATION STRATEGY

4.1 Hybrid EMOA-adaptive local search

An adaptation strategy for adjusting local search parameters should be applicable to both unimodal and multimodal problems without any *a priori* knowledge on MOPs at hand. For uni-modal problems, a large p_{ls} would be beneficial to get fast convergence to the real Pareto-optimal front. For multimodal or severely nonlinear problems, a small or even zero p_{ls} would be preferred. Even for a same problem, the optimal value of p_{ls} would change according to the stage of evolution process; in an earlier stage, more global exploration would be preferred than in a later stage of evolution process. However, in general we are not aware of which stage the evolution process is in, because the required number of generation and population for an evolution to fully converge would be problem dependant.

In other words, an adaptation algorithm should be able to detect property of MOPs on the fly and adjust parameters accordingly. For example, if a directional local search is more effective than genetic operators the local search probability should be increased, and vice versa. In this study, and local search probability p_{ls} was used as adaptation parameter. Adaptation of the parameter is based on local and global effectiveness of a local search operator. The local effectiveness is measured by quantitative comparison of improvements made by genetic operators and a local search operator. The global effectiveness is determined by the ratio of numbers of nondominated solutions by local search and genetic search operations.

4.2 Calculation of average improvement by evolutionary operators

Before we proceed to present a strategy for the adaptive local search, a measure of average improvement by genetic operators in terms of the composite objective Z is be defined. For this purpose, we compare a mating pair P_1, P_2 and their two children C_1 and C_2 regarding the dominance relations and differences in F as illustrated in Fig. 5. The improvement $\Delta F_{ij} \geq 0, i, j \in \{1, 2\}$ is defined as

$$\Delta Z_{ij} = \begin{cases} Z_{P_i} - Z_{C_i} & \text{if } C_i \text{ dominates } P_i \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Mutually non-dominating parent and child solutions are not taken into account in the averaging process by increasing N_{nocount} , the number of no-counted solutions by one. After calculating ΔF_{ij} for all mating couples of parents and their children in a generation, the average improvement by evolutionary operators imp_{evol} is calculated by

$$imp_{evol} = \frac{\sum_{j=1}^{N_{pop}/2} \max(\Delta Z_{11} + \Delta Z_{22}, \Delta Z_{12} + \Delta Z_{21})_j}{N_{pop} - N_{nocount}} \quad (12)$$

where N_{nocount} is defined as the number of parent solutions in the mating process of each generation that are mutually nondominating with their two children. For example, in Fig. 5, N_{nocount} is increased by one because P1 is neither dominating or being dominated by C1 or C2 and P2 is dominated by C2. Also, ΔZ_{11} , ΔZ_{21} , and ΔZ_{22} are all zero, and ΔZ_{12} is non-zero from Eq. (11).

The metric of evolutionary improvement imp_{evol} provides a single averaged value of distributed improvements in terms of convergence for a population of each generation. The metric imp_{evol} concerns only convergence aspect of improvement made by evolutionary operators because it is meant to be compared with improvement in Z by local search given in Eq. (10). We are not considering enhancing diversity or sidestepping by eLS.[13]

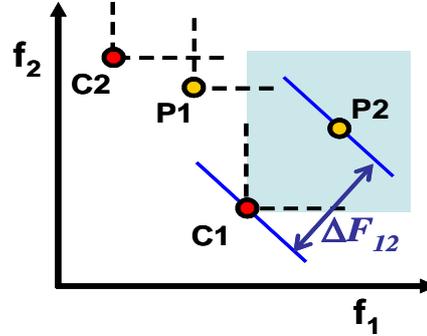


Figure 5: Calculation of improvement by evolutionary operators.

4.3 local effectiveness of a local search

Given the measures of convergence improvement by the directional local search operator and evolutionary operators, defined in Eqs. (10) and (12), respectively, the local effectiveness of a local search operator to evolutionary operators can now be determined. Fig. 6 shows three zones in the objective space representing zones of success, failure and nondeterminacy for a local search operator. For evaluation of local effectiveness (worthiness) of conducting a local search, we introduce the required improvement in Z , $imp_{req} \equiv C_{ls} \times imp_{evol}$, to impose a condition that a local search should make (at least) C_{ls} times more improvement than evolutionary operators since it spends C_{ls} times more computational cost than evolutionary operators. The center point of the dotted cross in Fig. 6 represents the initial solution in the objective space, and the other points are possible locations of a final solution of a directional local search. If the local search achieves at least the required improvement imp_{req} , the local search is defined as successful. If the final solution is dominated by the initial solution, the local search defined as a failure. If the local search is not successful and the final solution is not dominated by the initial solution, the solution lies in the zone of non-determinacy: we cannot tell whether the local search is a success or a failure at this moment. In summary, the success and failure of a local search is determined as follows:

```

if  $imp_{ls} \geq imp_{req}$ 
    a local search is a success;  $N_{success} := N_{success} + 1$ 
otherwise
    if the initial solution dominates the final solution
        a local search is a failure;  $N_{failure} := N_{failure} + 1$ 
    endif
endif

```

The numbers of success (N_{success}) and failure (N_{failure}) are counted for all the local search operation conducted in each generation. The difference of numbers of successes and failures, $N_{\text{success}} - N_{\text{failure}}$,

defines the local effectiveness in a generation and is used as major indicator in our adaptation strategy. If $N_{\text{success}} - N_{\text{failure}} > 0$, we define the local search in the generation as locally effective.

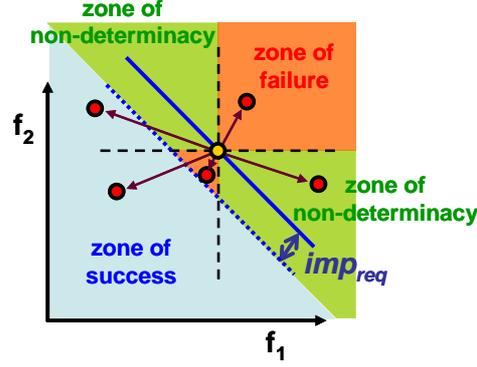


Figure 6: Schematic representation of three zones for defining the effectiveness of a local search over evolutionary operators.

4.4 Global effectiveness of a local search

Global effectiveness is to make an adaptation more conservative even when the local search is seemingly locally effective ($N_{\text{success}} > N_{\text{failure}}$). The global effectiveness can be evaluated by checking the number of local search solutions in the nondominated solutions, which can be considered as final solution outputs of the optimization. The basic idea behind the global effectiveness is that the number ratio of local search solutions to evolutionary search solutions in the set of nondominated solutions should be higher than the computational cost ratio spent for the local search and evolutionary search. If it is not the case, we can tell that the local search is not working properly in a global perspective even if it appears locally effective. A constraint for the global effectiveness can be written as

$$B_{ls} \geq \bar{C}_{ls} \times p_{ls} \quad (12)$$

where B_{ls} is the ratio of the number of local search solutions to the number of evolutionary search solutions contained in the set of nondominated solutions as defined in Table 1. For EMOAs with external archive for non-dominated solutions such as MOGA [16], B_{ls} is defined in the external archive. For NSGA-II, B_{ls} is defined for the nondominated solutions before the selection process. Another constraint related to the global effectiveness is an upper limit constraint on B_{ls} ; It is based on empirical observations that at least half of the nondominated solution set needs to be filled with solutions by evolutionary operators to prevent genetic shift and premature convergence. Thus, an additional constraint is introduced:

$$B_{ls} \leq 1.0 \quad (13)$$

The two constraints for the global effectiveness in Eqs. (12) and (13) can be put together as follows:

$$\bar{C}_{ls} \times p_{ls} \leq B_{ls} \leq 1.0 \quad (14)$$

4.4 Overall Procedure of Hybrid MOGA-adaptive local search approach

In our previous work [13], the stencil radius r and the local search probability p_{ls} were tested as adaptation parameters and the results were compared. For many test functions of various properties, it was found that keeping $r=1$ and adjusting p_{ls} gave satisfactory results. Therefore, in the present study, the local search probability p_{ls} is selected as the adaptation parameter. The adaptation method is based on local and global effectiveness of a local search at each generation. If the local search is globally effective in a generation, an adaptation parameter is increased based on the local effectiveness. Otherwise, the extent of application of a local search is reduced because the local search solutions are not good enough to remain in the nondominated solution set or the local search is too prevailing. The detailed algorithm of the present adaptation strategy is presented in Fig. 7. Performance indices and parameters for the adaptive local search strategy are listed in Table 1. The present adaptive strategy can be combined with any EMOAs and local search methods.

The adaptation formula is as follows: 1) If any of the global effectiveness constraints in Eq. (14) is violated, p_{ls} is reduced by 10%. 2) Otherwise p_{ls} is adjusted by:

$$p_{ls} := p_{ls} + \Delta p_{ls} (N_{\text{success}} - N_{\text{failure}}) \quad (15)$$

```

Algorithm AdaptiveLocalSearch
(Local Search with adaptive probability and/or adaptive neighborhood radius)
1: if  $N_{ls} > 0$  then /* # of local search solutions in the previous generation */
2:   if r_adaptation then
3:     if  $p_{ls} \times \overline{C_{ls}} \leq B_{ls} \leq 1$  then /* globally effective */
4:        $r := r + \Delta r (N_{success} - N_{failure})$ 
5:     else
6:        $r := r - \Delta r$ 
7:     end if
8:      $r := \max(r, r^{\min})$ 
9:      $r := \min(r, r^{\max})$ 
10:  end if
11:  if pls_adaptation then
12:    if  $p_{ls} \times \overline{C_{ls}} \leq B_{ls} \leq 1$  then /* globally effective */
13:       $p_{ls} := p_{ls} + \Delta p_{ls} (N_{success} - N_{failure})$ 
14:    else
15:       $p_{ls} := 0.9 \times p_{ls}$ 
16:    end if
17:     $p_{ls} := \max(p_{ls}, p_{ls}^{\min})$ 
18:     $p_{ls} := \min(p_{ls}, p_{ls}^{\max})$ 
19:  end if
20: end if
21: Initialize  $N_{ls}$ ,  $N_{success}$  and  $N_{failure}$  as zero
22: loop do  $i = 1, \dots, N_{pop}$ 
23:   if random_number  $\in [0,1] < p_{ls}$  then
24:      $\mathbf{x}_i \in P$ , a set of  $N_{pop}$  parents
25:      $S_2 = find\_NeighborSet(H, \mathbf{x}_i, r)$ 
26:     if  $|S_2| > 0$  then
27:       call  $eLS(\mathbf{x}_i, |S_2|, \mathbf{x}_{opt})$ 
28:        $N_{ls} := N_{ls} + 1$ 
29:       if  $imp_{ls} > imp_{req}$  then
30:          $N_{success} := N_{success} + 1$ 
31:       else if  $\mathbf{x}_i < \mathbf{x}_{opt}$  then /*  $\mathbf{x}_{opt}$  is the local search solution */
32:          $N_{failure} := N_{failure} + 1$ 
33:       end if
34:     end if
35:   end if
36: end loop

```

Figure 7: Algorithm for the adaptive directional local search operator, AdaptiveLocalSearch [13]

Table 1. Parameters used or defined in the local search

Name	Definition	Remarks
B_{ls}	$\frac{\text{number of local search solutions in nondominated solutions}}{\text{number of evolutionary search solutions in nondominated solutions}}$	Ratio of local search solutions in the nondominated solutions
\bar{B}_{ls}	average of B_{ls} in the whole optimization process	
C_{ls}	Cost, or the number of function calls for each local search.	In the present study, $C_{ls} = 2$
\bar{C}_{ls}	Reference number of function calls in a local search	set as 2
imp_{ls}	Improvement in the composite objective made by a local search	$imp_{ls} = Z_0 - Z_{opt}$ (Eq.10)
imp_{evol}	Averaged improvement in the composite objective by evolutionary operators at each generation	(Eq.12)
imp_{req}	$imp_{evol} \times C_{ls}$	required improvement for a local search to be more effective than evolutionary operators
N_{ls}	number of local search runs in each generation	
N_{max}	Maximum number of trials for checking local convexity in a line search	1 (input)
p_{ls}	Local search probability	$[p_{ls}^{min}, p_{ls}^{max}]$
\bar{p}_{ls}	$\frac{1}{C_{ls}} \frac{\text{number of function calls by local search}}{\text{number of function calls by evolutionary search}}$	Average of effective local search probability based on actual function calls by local search
$ S _{max}$	Maximum size of stencil set S	Set as $\max(100, N_{dv})$ (input)
\bar{p}_{ls}^{max}	prespecified maximum value for \bar{p}_{ls}	0.20 (input)
Δp_{ls}	increment step for variation of p_{ls}	$1/N_{pop}$ (input)
p_{ls}^{min}	minimum value for p_{ls}	$1/N_{pop}$ (input)
p_{ls}^{max}	maximum value for p_{ls}	1.0 (input)

5. AIRFOIL DESIGN EXAMPLES

5.1 Design Problems

For performance comparison of a baseline EMOA and the hybrid EMOA-adaptive local search method on multi-objective aerodynamic design optimization problems, here conducted are subsonic and transonic airfoil design examples. For subsonic airfoil design problems, the following two cases are considered: [17]

L/D maximization and drag minimization at

- 1) a cruise condition ($M_\infty = 0.417$, $AOA = 0^\circ$, $Re = 2.5 \times 10^6$) and
- 2) a loiter condition ($M_\infty = 0.1$, $AOA = 6^\circ$, $Re = 6.5 \times 10^5$.)

For transonic airfoil design problems, also two cases are considered: [18]

- 3) L/D maximization and drag minimization at a cruise condition ($M_\infty = 0.78$, $AOA = 2^\circ$, $Re = 7.0 \times 10^6$.) and
- 4) Lift maximization at flow condition 1 ($M_\infty = 0.20$, $AOA = 10.8^\circ$, $Re = 5.0 \times 10^6$) and L/D maximization at flow condition 2 ($M_\infty = 0.77$, $AOA = 1.0^\circ$, $Re = 1.0 \times 10^7$)

All the above design problems are two-objective problems. And the first three design examples are single-point design, and design example 4 is a two-point design.

We imposed two constraints on all the design problems: one is that the maximum airfoil thickness should not be less than 10% chord length, and the other is that the flow behavior should be steady at the design condition. The steady flow constraint is assumed to be satisfied if a flow simulation converges for a specified convergence criterion.

The subsonic and transonic design examples are conducted with different flow analysis code, shape parameterization, and baseline EMOA, which will be described below.

5.2 Flow Simulation

For the subsonic airfoil design problems, the XFOIL code [19] was used because of its validated accuracy for subsonic airfoils and rapid turnaround time. The XFOIL code combines a panel method and an integral boundary layer formulation for simulation of potential and viscous boundary layer flow around airfoils. The number of panels on airfoil was selected as 160 after sensitivity study on the number of panels around an airfoil.

5.3 Airfoil Shape Parameterization

For the subsonic airfoil design problems, the airfoil shape is parameterized by the CST method. [20] There are sixteen control points used to define an airfoil shape. Eight control points are arranged at the upper and lower side of airfoil, respectively.

For the transonic design problems, a third order B-spline method is adopted for geometric parameterization. There are nine control points distributed around an airfoil shape. Two control points are fixed at the trailing edge (1, 0) and one control point is fixed at the origin (0, 0), which results in twelve design variables: x and y coordinates for six movable control points. [21] Figure 8 depicts B-spline control points around an airfoil.

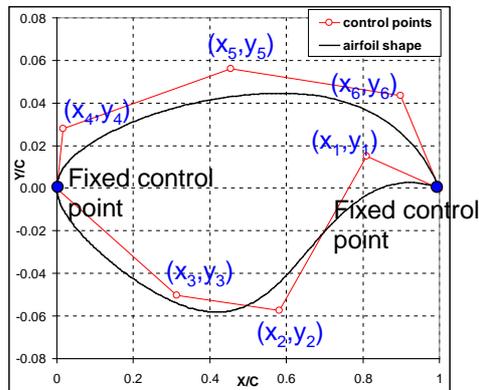


Figure 8: B-spline parameterization of transonic airfoil shape:
12 design variables are used for the parameterization

5.4 Baseline EMO Algorithms

In subsonic airfoil designs, NSGA-II is used as a baseline EMOA, to which the adaptive directional local search operator is added. The hybrid algorithm of NSGA-II + adaptive local search method is denoted as NSGA-II+als. For transonic airfoil design problems, MOGA is adopted as a baseline EMOA. The hybrid algorithm will be denoted as MOGA+als.

For evolutionary operators in both baseline EMOAs, we use the SBX recombination with a distribution parameter $\eta_c = 15$ and probability of crossover, $p_c = 1$, and the polynomial mutation operator with a distribution parameter $\eta_m = 20$ and probability of mutation, $p_m = 1/N_{dv}$. [22] Constraints are treated by a Pareto-based constraint handling technique. [23]

The purpose of using two different baseline EMOAs is not to compare NSGA-II and MOGA, but to show that the present adaptive local search method can be combine with general baseline EMOAs. Also we want to show that the hybrid method improve performance when the design space is uni-

modal and does not do any harm to the performance of the baseline EMOA when the design space is multi-modal by minimizing the use of the function call budget.

5.5 Performance Index and Statistical Test

For comparison of quality of Pareto front solutions obtained by the multi-objective algorithms, we use the hypervolume indicator, which measures both convergence and diversity of solutions on the Pareto front. We used the *hyp_ind* program contained in performance assessment tools of PISA.[24] The *hyp_ind* calculates the hypervolume difference indicator, I_H^- , the difference in hypervolume between a reference set and an approximate set under consideration, and therefore the smaller is the better.

Comparison of performance of stochastic algorithms, statistical test is required for multiple independent runs of each algorithm. In this study, the quick turnaround time of XFOIL enabled to run optimizations multiple times. So, subsonic airfoil design examples using the XFOIL code was run 31 times, and statistical test was performed using a program for the Kruskal Wallis test included in PISA for conducting a nonparametric test for difference between multiple independent samples. The Kruskal Wallis test is adopted with a confidence level of 95% (i.e., p value below 0.05.).

In case of transonic designs, the computational burden for simulation of two-dimensional Navier-Stokes code was prohibitive for running 31 optimizations, and thus just two optimizations was run for comparison.

5.6 Results of Design Examples

5.6.1 Design Example 1: Subsonic Cruise Design

For the baseline EMOA, the population size and number of generation are set to be 96 and 60, respectively. Hence the total number of function evaluation is 5,760. For the hybrid method, the population size is the same, and the generation number is adjusted so that the total number of evaluations is not exceeded. The optimization was run 31 times independently for the algorithms under comparison: the NSGA-II and the NSGA-II+*als*. The multi-objective optimization problem at a cruise condition is defined as follows:

$$\begin{aligned}
 & \text{Maximize } L/D \\
 & \text{Minimize } C_d \\
 & \text{Subject to } t_{\max}/c \geq 0.10 \\
 & \quad \text{steady flow field}
 \end{aligned} \tag{16}$$

The flow condition is defined as: $M_\infty = 0.417$, $AOA = 0^\circ$, $Re = 2.5 \times 10^6$.

In Fig.9, box plots for hypervolume indicator are depicted. A box plot represents minimum, first quartile, third quartile and maximum values of multiple runs. A dot in the box indicates second quartile or a median value. A statistical test shows that the performance of NSGA-II+*als* is better than NSGA-II with a p value of 0.02694.

Figure 10 shows variation of adaptation parameters and p_{ls} for an optimization run corresponding to third quartile of the box plot for NSGA-II+*als*. Imp_{evol} , the average improvement made by the evolutionary operators decreases from O(2) to O(-3) indicating convergence of the evolutionary optimization process. In the run, p_{ls} increases from 1% up to 16% in the middle of the design history because the directional local search is locally effective. After this point, p_{ls} decreases to p_{ls}^{\min} , but the constraints of the global effectiveness, $\bar{p}_{ls} \leq \bar{B}_{ls}$ is always satisfied throughout the optimization history. It is also noted that the solutions found in the local search are of good quality to survive in the reduction process for Best-N solutions, although \bar{B}_{ls} gradually decreases along the optimization process.

Pareto fronts for the NSGA-II and NSGA-II+*als* are depicted in Fig.11. The Pareto front by the NSGA-II+*als* is superior to the results of the baseline algorithm due to the effectiveness of the adaptive local search.

Airfoil shapes and surface pressure distributions are shown in Fig.12 for two extreme solutions and one compromising solution on the Pareto front obtained by NSGA-II+*als*.

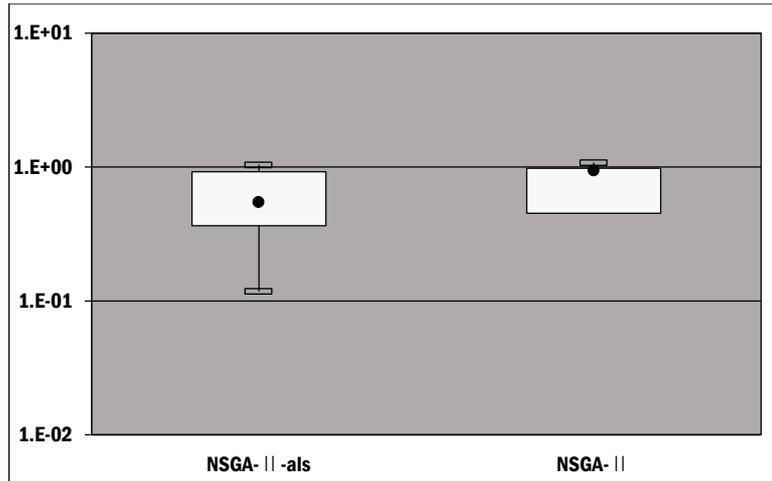


Figure 9: Box plots of hypervolume indicator for the design example 1 (smaller value is better).

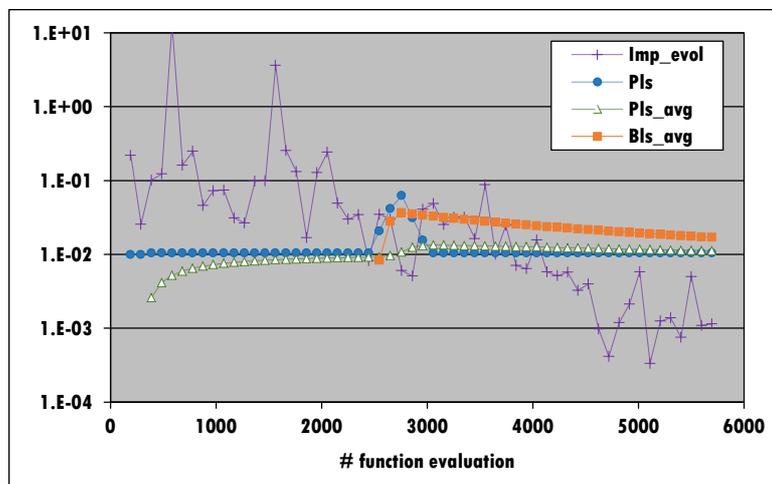


Figure 10: History of adaptive local search parameters for design example 1

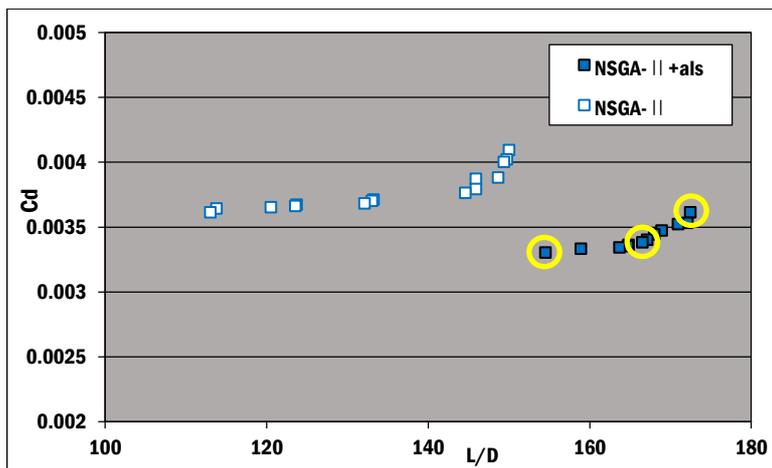


Figure 11: Comparison of Pareto fronts for design example I. Yellow circles indicate two extreme and one compromising solutions to be presented in the following figure.

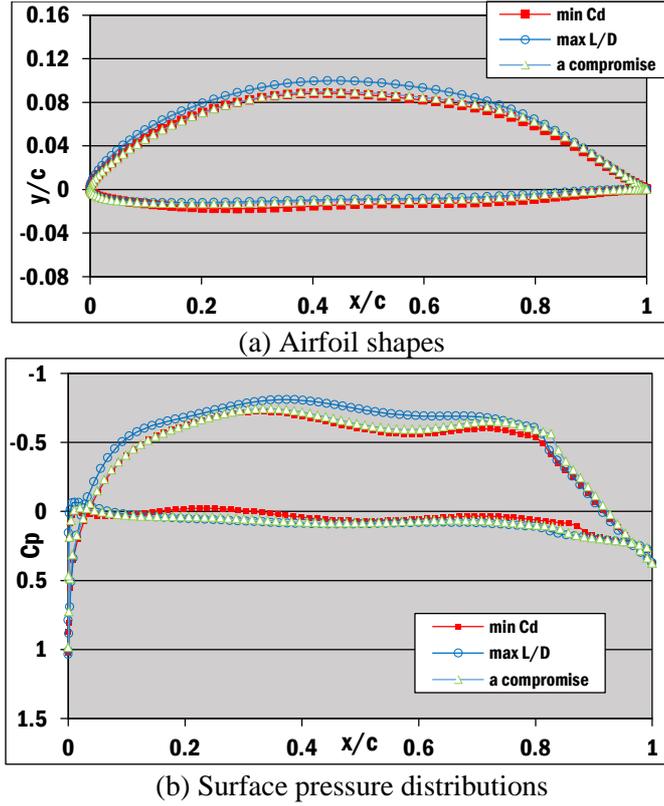


Figure 12: Results of the NSGA-II+als run for design example 1

5.6.2 Design Example 2: Subsonic Loiter Design

The same multi-objective optimization problem in Eq.(16) is also used for a loiter design with flow conditions for loitering: $M_\infty = 0.1$, $AOA = 6^\circ$, Reynolds number = 6.5×10^5 .

In Fig.13, box plots for hypervolume indicator of example 2 are depicted. For the multi-objective optimization problems, it is show that the performance of NSGA-II+als is almost same as NSGA-II, and the p value is 0.3743, which does not satisfy the significance level.

Figure14 show that variation of adaptation parameters and p_{ls} which corresponds to middle point of the box plot for NSGA-II+als. Imp_{evol} , the average improvement made by the evolutionary operators decreases from $O(2)$ to $O(-3)$ indicating convergence of the evolutionary optimization process. In this run, p_{ls} is hardly perturbed in the design history until the number of function evaluation is about 35,000, because the constraint of global effectiveness $\bar{p}_{ls} \leq \bar{B}_{ls}$ is not satisfied.

Since \bar{B}_{ls} is lower than \bar{p}_{ls} , it is noted that the local search solutions by the minimum magnitude of p_{ls} are having difficulties to survive in the reduction process for Best-N solutions. p_{ls} is slightly perturbed when the number of function evaluation is about 40,000. However, it seems like the local search solutions does not make a significant difference in the optimization process, presumably because the loitering condition with a relatively high AOA makes the design space multi-modal.

Pareto fronts for the middle points of the box plots by NSGA-II and the NSGA-II+als are depicted in Fig.15. In the present case, Pareto front by the NSGA-II+als is basically supposed to be very similar to the result of the NSGA-II algorithm due to fact that the adaptive local search is ineffective.

Airfoil shapes and surface pressure distributions are compared in Fig.16 for two extreme solutions and one compromising solution on the Pareto front of NSGA-II+als. The minimum Cd solution has the smallest thickness and camber, while the maximum L/D solution has the largest thickness and camber.

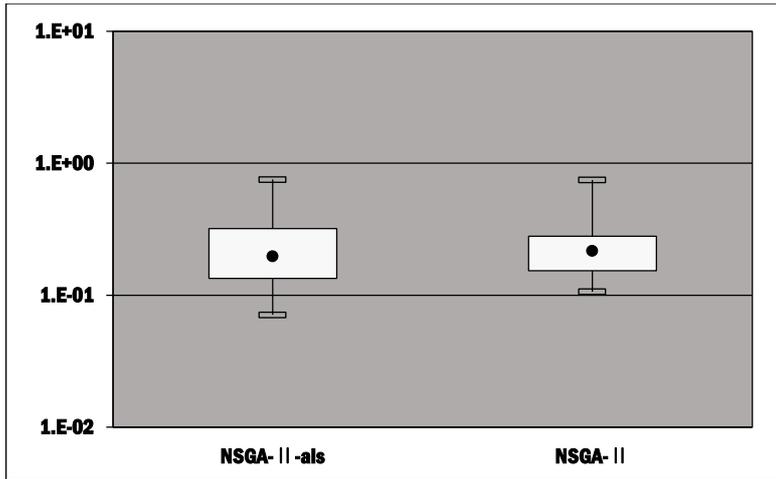


Figure 13: Box plots of hypervolume indicator for the design example 2 (smaller value is better).

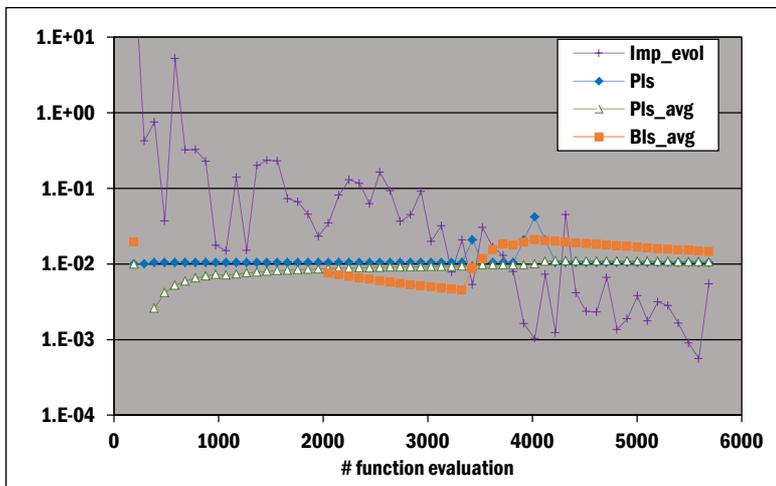


Figure 14: History of adaptive local search parameters for design example 2

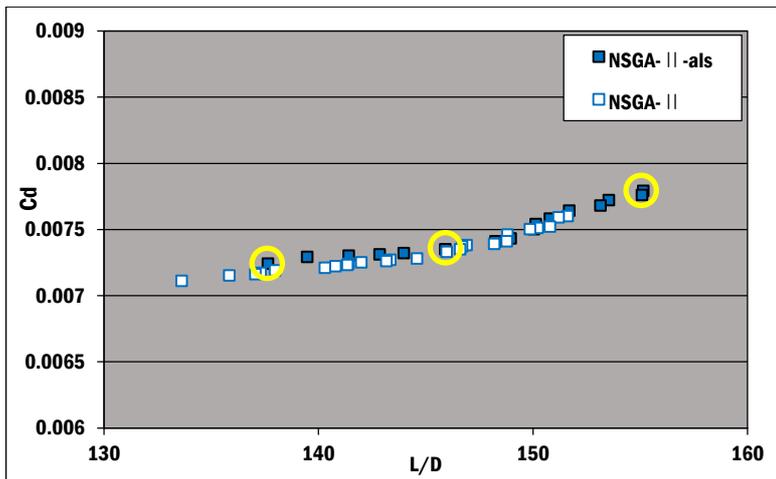
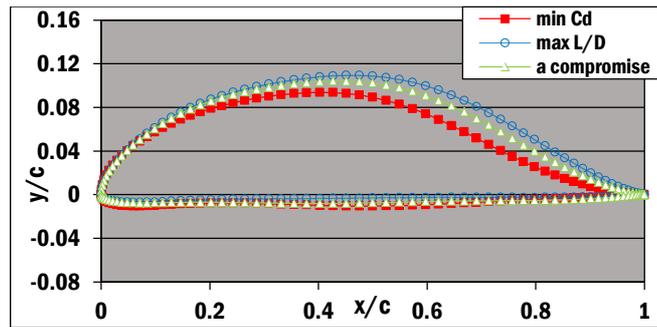
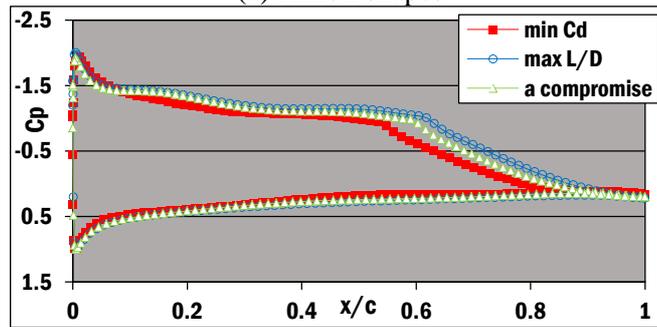


Figure 15: Comparison of Pareto fronts for design example 2. Yellow circles indicate two extreme and one compromising solutions to be presented in the following figure



(a) Airfoil shapes



(b) Distributions of surface pressure coefficients

Figure 16. Results of the NSGA-II+als for design example 2

6. Conclusions

A novel adaptive hybrid strategy for EMOA and a directional local search method has been applied to multi-objective aerodynamic optimization problems. The adaptive strategy adjusts probability for a local search operator depending relative effectiveness of evolutionary operators and a local search operator considering solution improvement, required computational cost and survival rate of local search solutions through the reduction process. Optimization examples are defined for multi-objective airfoil design in subsonic and transonic flow regimes with constraint on maximum thickness and flow steadiness. Results show that the suggested method successfully improves convergence for cases where the local search works well and minimizes waste of computational cost when local search is not effective. Statistical test confirms superiority of the hybrid method to the baseline EMOA. The local search is found to be effective for cruise designs and ineffective for lift maximization or high angle of attack conditions, which are prone to flow separations

References

- [1] Fonseca, C.M. and Fleming, P.J., "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion, and Generalization," in Proceedings of the Fifth International Conference on Genetic Algorithms, pp.416 – 423, 1993.
- [2] Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T. "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II," IEEE Trans. Evol. Comput. Vol. 6, No.2, 2002, pp.182–197.
- [3] Zitzler, E., Laumanns, M., and Thiele, L. "SPEA2: Improving the Strength Pareto Evolutionary Algorithms," TIK report 103, Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH) Zurich, May 2001.
- [4] Sasaki, D. and Obayashi, S., "Efficient Search for Trade-Offs by Adaptive Range Multi-Objective Genetic Algorithms," Journal of Aerospace Computing, Information, and Communication, Vol. 2, No. 1, 2005, pp. 44 – 64.

- [5] Deb, K., "Multi-Objective Optimization Using Evolutionary Algorithms," Wiley, 2001.
- [6] Goldberg, D. E. and Richardson, J., "Genetic Algorithms with Sharing for Multimodal Function Optimization," Proceedings of the First International Conference on Genetic Algorithms and Their Applications, pp.41-49, 1987.
- [7] Poloni, C., "Hybrid GA for Multi Objective Aerodynamic Shape Optimization," Genetic Algorithms in Engineering and Computer Science, John Wiley and Sons, Vol. 33, pp.397-415, 1995.
- [8] Vicini, A. and Quagliarella, D.: Airfoil and Wing Design through Hybrid Optimization Strategies. *AIAA Journal* Vol. 37, no. 5, pp. 634 – 641. May 1999
- [9] Espinoza, F., Minsker, B. and Goldberg, D.E., "A Self Adaptive Hybrid Genetic Algorithm," Proceedings of GECCO 2001, Morgan Kaufmann Publishers, 2001.
- [10] Dumas L., Muyl F., Herbert V., "Hybrid Method for Aerodynamic Shape Optimization in Automotive Industry," Proceedings of AMIF 2002, special issue of Computers and Fluids, Vol. 33, pp. 849 – 858, 2004.
- [11] Song, W., "Multiobjective Memetic Algorithm and Its Application in Robust Airfoil Shape Optimization," *Multiobjective Memetic Algorithms*, Springer Berlin / Heidelberg, 2009, pp. 389-402.
- [12] Knowles, J. and Corne, D., "Memetic Algorithms for Multiobjective Optimization: Issues, Methods and Prospects," *Recent Advances in Memetic Algorithms*, Springer Berlin / Heidelberg, 2005, pp. 313-352
- [13] Kim, H. and Liou, M.-S., "Adaptive Directional Local Search Strategy for Hybrid Evolutionary Multiobjective Optimization," *Applied Soft Computing*, Vol.19 June 2014, pp.290-311.
- [14] Synhya, K., Deb, K. and Miettinen, K., "A Local Search Based Evolutionary MultiObjective Approach for Fast and Accurate Convergence," in Parallel Problem Solving from Nature-PPSN X, ser. Lecture Notes in Computer Science, vol. 5199/2008. Springer, 2008, pp. 815 – 824.
- [15] Schutze, O., Sanchez, G., Lara, A., and Coello, G. A. C. "HCS: A New Local Search Strategy for Memetic Multi-Objective Evolutionary Algorithms," CINVESTAV report TR-OS-2008-05, 2008.
- [16] Kim, H. and Liou, M.-S., "New fitness sharing approach for multiobjective genetic algorithms", *J. Glob. Optim.* Vol. 55, No.3, 2013, 579-595.
- [17] Fincham, J. H. S., and Friswell, M. I., "Aerodynamic Optimisation of a Camber Morphing Aerofoil," *Aerospace Science and technology*, Vol 43, 2015, pp: 245 – 255.
- [18] Naujoks, B., Willmes, L., Haase, W., Back, T. and Schutz, M., "Multi-point Airfoil Optimization Using Evolution Strategies," In *ECCOMAS 2000, European Congress on Computational Methods in Applied Sciences and Engineering*, 2000.
- [19] Drela, M., "XFOIL – an analysis and design system for low Reynolds number airfoils," T.J. Mueller (Ed.), *Low Reynolds Number Aerodynamics*, Springer-Verlag, Berlin, 1989, pp. 1-12.
- [20] Kulfan, Brenda M., "Universal parametric geometry representation method." *Journal of Aircraft*, Vol. 45, No. 1, 2008, pp. 142-158.
- [21] Oyama, A. Verburg, P., Nonomura, T. and H. Hoeijmaker, "Flow Field Mining of Pareto-Optimal Airfoils Using Proper Orthogonal Decomposition," AIAA 2010-1140, Orlando, FL, Jan. 2010.
- [22] Chiba, K., Kanazaki, M., Shimada, T., "Simple control of oxidizer flux for efficient extinction-reignition on a single-stage hybrid rocket," *Aerospace Science and Technology*, Vol. 71, Dec. 2017, pp.109-118.
- [23] Oyama, A., Shimoyama, K., and Fujii, K., "New Constraint-Handling Method for Multi-objective Multi-Constraint Evolutionary Optimization," *Transactions of the Japanese Society for Aeronautical and Space Sciences*, Vol. 50, No. 167, 2007, pp. 56-62.
- [24] Zitzler, E., Thiele, L., Laumanns, M. Fonseca, C. M., and Fonseca, V., "Performance Assessment of Multiobjective Optimizers: An Analysis and Review," *IEEE Transactions on Evolutionary Computation*, Vol. 7, Apr. 2003, pp. 117-132,