MULTIOBJECTIVE PARTICLE SWARM OPTIMIZATION TECHNIQUE AS AN EFFECTIVE TOOL FOR AIRCRAFT REQUIREMENTS ANALYSIS

L. Blasi, S. Barbato, L. Iuspa
Department of Aerospace and Mechanical Engineering – Second University of Naples
Via Roma 29, 81031 Aversa
Italy

SUMMARY

In this paper, a multi-objective particle swarm optimization (MOPSO) procedure has been developed and applied in the field of aircraft requirement analysis. In order to identify useful set-up schemes for algorithm control parameters, the optimisation procedure has been preliminarily verified with test-case functions. Moreover, specific tools have been implemented to improve MOPSO effectiveness in finding Pareto front as wide and uniform as possible. The optimization procedure has been subsequently applied to the preliminary definition of a civil transport aircraft configuration. Both maximum takeoff weight and block time have been selected as objective functions to be minimized. At the end of optimization process, useful sensitivity curves, showing cruise speed requirement effects on aircraft main characteristics, have been obtained. Finally, a comparison with a similar task driven by a genetic algorithm has been performed in order to highlight some advantages offered by MOPSO procedure.

1. INTRODUCTION

In most engineering practice, the design process itself can be commonly regarded as a true multi-disciplinary, multi-objective optimization task[1], whose solutions represent a collection of best responses that meet different, often conflicting, requirements. The complex relationships involving design variables, objective functions and constraints (non-linear functions, discrete or discontinuous design domain etc.) encouraged, in the last years, the development of non-conventional, nature-inspired optimization methods like Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO). Both methods are population-based optimizers that find solution through a probabilistic search process, only guided by a fitness function value. However, PSO follows a different approach than GAs, as the evolution is obtained through cooperation among individuals (renamed particles, in PSO context) rather than competition[2]. The effectiveness shown by PSO-based technique in many single-objective optimization problems[3,4], combined with its capability to keep information about evolution of all the particles at the same time, has made the extension of PSO technique to multi-objective optimization problems (MOPSO) almost a natural progression. In particular, this work deals with the application of a MOPSO technique in the field of aircraft requirement analysis. By using the classical Pareto criterion, we can identify a set of optimal, non-dominated solutions. Following the Pareto statement, a solution is said to be *Pareto optimal* or *non-dominated*, if it is not possible to improve any objective function without deteriorating at least another one. The concept of non-dominated solution can be easily used to define a fitness function driving the optimization process. When we extend standard single-objective PSO formulation to multi-objective problem, particular care must be taken to the meaning of "swarm best position" and "single particle best position", as we obtain a set of different optimum solutions with the same level of goodness[5-8]. Therefore, the proposed MOPSO procedure has been applied first to test-case functions to analyze different strategies concerning the selection of the global best position as well as the local best position. Specific tools have been developed with the aim to improve MOPSO effectiveness in finding a Pareto front as wide and uniform as possible. Subsequently, we have investigated the MOPSO applicability in the field of aircraft requirements analysis. This problem has been deemed of great interest by authors as the availability of effective tools providing aircraft overall characteristics sensitivity for different figures of merit it is a very important factor during the conceptual design phase. A multi-objective optimization technique can be usefully utilized to understand how optimum configurations change as different objectives are chosen. In particular, maximum takeoff weight and cruise speed have been selected as objective functions to be minimized. Sensitivity curves obtained at the end of the optimization process can provide the designer a deeper understanding of the speed effect on the aircraft configuration, increasing the degree of confidence about the proposed speed requirement.

Finally, a comparison with results, previously obtained by a genetic-based multi-objective optimization procedure, has been performed.

2. MOPSO METHODOLOGY

2.1. PSO technique

The optimization technique based on Swarm Theory is a method which takes inspiration from the social behaviour of groups of simple creatures as swarms of bees, colonies of ants, flocks of birds etc, that exhibit some form of collective intelligence based on information exchange.

The searching for optimal solutions performed with PSO is obtained defining a population of particles, each one exploring the search space and communicating results to the rest of group. Each particle *i* at time *t* has two state variables:

- current position $\mathbf{x}_i(t)$
- current velocity $\mathbf{v}_i(t)$

as well as a small memory containing:

- previous best position $\mathbf{p}_i(t)$ (*personal best position*)
- swarm best position $\mathbf{g}_i(t)$ (*global best position*)

At time-step t+1 of the optimization process, the particle position is updated according to the relation

(1) $$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1)$$

where $\mathbf{v}$(t+1) is particle velocity, which is calculated as

(2) $$\mathbf{v}_i(t+1) = \chi\{\omega\,\mathbf{v}_i(t) + c_1 r_1[\mathbf{p}_i(t) - \mathbf{x}_i(t)] + \\ + c_2 r_2[\mathbf{g}_i(t) - \mathbf{x}_i(t)]\}$$

In relation (2) r_1 and r_2 are random numbers uniformly distributed over the range [0,1]; c_1 and c_2, named *cognitive* and *social parameter* respectively, are quantities that define the influence that $\mathbf{p}_i(t)$ and $\mathbf{g}_i(t)$ have on the particle velocity; parameter ω (*inertia weight*) is a reducing factor for $\mathbf{v}_i(t)$, while parameter χ (*constriction factor*) is used to limit the particle velocity. Both inertia weight and constriction factor control swarm exploitation as well as exploration capability, heavily affecting convergence speed and effectiveness of the optimization task[9].

The implementation of Pareto Optimality criterion makes PSO a helpful method for solving multi-objective optimization problems. This technique is called Multi-Objective Particle Swarm Optimization (MOPSO).

2.2. Pareto Optimality

In a constrained multi-objective optimisation, we seek to simultaneously extremise D objectives $f_i(\mathbf{x})$, i = 1,...,D all depending upon a vector $\mathbf{x}$ of K decision variables, subject to J constraints $c_j(\mathbf{x}) \geq 0$, j=1,...,J.

Assuming, for a sake of simplicity, all these objectives are to be minimised; the problem can be stated as[10]:

(3) $$\min!\quad f_i(\mathbf{x}) \qquad i=1,\ldots,D$$

subject to:

(4) $$c_j(\mathbf{x}) \geq 0 \quad j=1,\ldots,J$$

A decision vector $\mathbf{u}$ is said to dominate a decision vector $\mathbf{v}$ (denoted $\mathbf{u} \prec \mathbf{v}$) if

(5) $$f_i(\mathbf{u}) \leq f_i(\mathbf{v}) \;\; \forall\, i \in \{1,\ldots,D\}$$

and

(6) $$\exists\, j \in \{1,...,D\} : f_j(\mathbf{u}) < f_j(\mathbf{v})$$

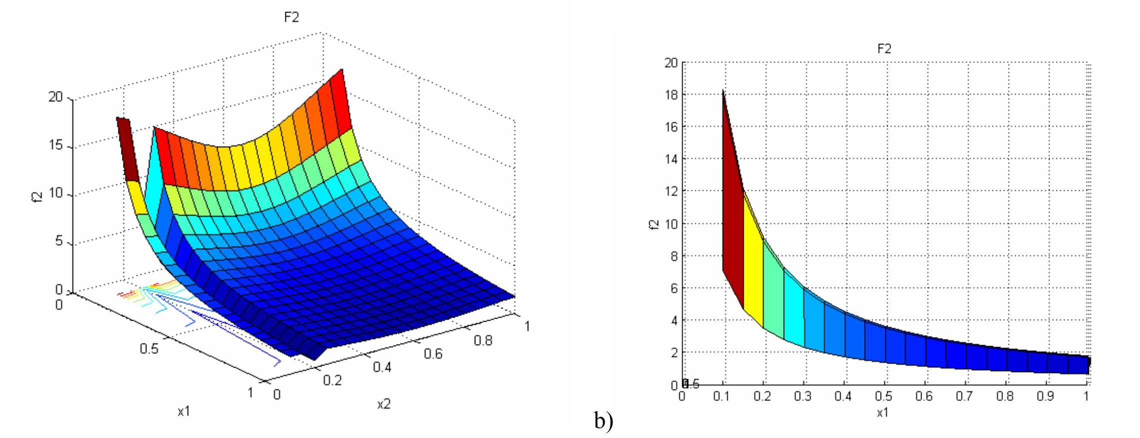
The scope, of a multi-objective optimization problem, is therefore to identify a set of non-dominated solutions. The corresponding objective vectors in the objective space are referred to as the *Pareto front*. Pareto optimality concept can be easily used to define a MOPSO fitness function that takes in account the degree of dominance of each solution among the population.

3. ADVANCES IN MOPSO METHODOLOGY

The preliminary application of MOPSO methodology to classical multi-objective test problems is mainly focused on the proper set-up of algorithm control parameters, as well as the implementation of specific tools for Pareto front improvement.

3.1. Advances in Pareto front finding: the sw*itch* tool

Analysing some problems with many sub-optimal solutions, the cognitive and social sub-parts of the velocity vector can exhibit conflicting responses.



a) b)

FIG. 1 DEB-Bimodal 3-D diagram (a) and objective functions domain (b)

This circumstance causes a "suspension" of particles far from good solutions; the quality of result is seriously compromised if the amount of suspended particles is a conspicuous part of the whole swarm.
Such a situation occurs, for example, when we try to solve the two-objective, unconstrained test-case DEB-Bimodal[11], defined as follows:

$$(7) \qquad \min! \quad \begin{array}{ll} f_1 = x_1 & x_1 \in [0.1 \quad 1] \\ f_2 = g(x_2)/x_1 & x_2 \in [0.1 \quad 1] \end{array}$$

Figures 1-a and 1-b show function f_2 and the objective space respectively. The morphology analysis of function f_2 can be helpful for a better understanding of the swarm behaviour in the attempt to reach the Pareto front. The bottom of the narrow gorge, visible in Figure 1-a, represents the Pareto front. The limited extent of this region and its position, very close to an edge of design variables space, make it almost invisible when compared to the bottom of the large concavity forming the second, sub-optimal front.
Figure 2 shows the result obtained with 250 particles per 100 iterations. In this plot the particles swarm, the Pareto front and the objective space are reported respectively in black, blue and yellow colour.
Very few particles reach the optimal region, whereas most of them stay restrained in the sub-optimal region.
In order to allow particles to reach the optimal region, a simple criterion, acting as a switch that inhibits personal knowledge in the velocity vector, has been developed.
Personal knowledge contribution is neglected (zeroed) when the following condition is met:

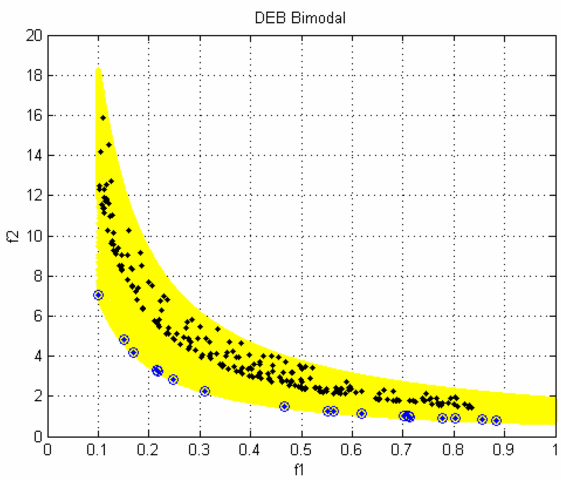


FIG. 2 MOPSO application to DEB-Bimodal problem (SWITCH OFF)

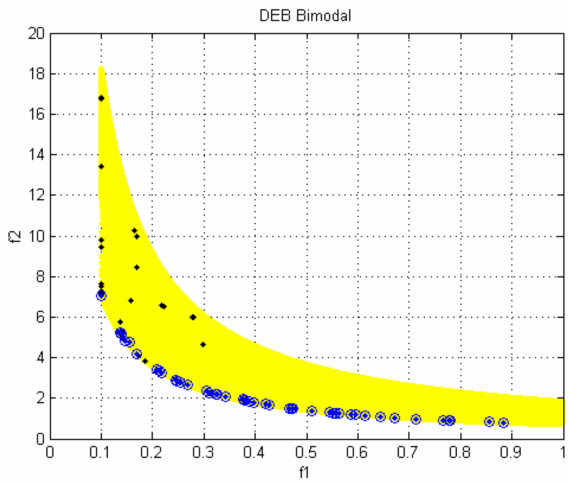


FIG. 3 MOPSO application to DEB-Bimodal problem (SWITCH ON)

$$a\,\frac{\dfrac{\left|\omega\,\mathbf{v}\right|+\left|\phi_1(\mathbf{p}-\mathbf{x})+\phi_2(\mathbf{g}-\mathbf{x})\right|}{\left|\phi_{1\max}(\overline{\mathbf{x}}-\underline{\mathbf{x}})+\phi_{2\max}(\overline{\mathbf{x}}-\underline{\mathbf{x}})\right|}+1}{\dfrac{\left|\phi_1(\mathbf{p}-\mathbf{x})-\phi_2(\mathbf{g}-\mathbf{x})\right|}{\left|\phi_{1\max}(\overline{\mathbf{x}}-\underline{\mathbf{x}})+\phi_{2\max}(\overline{\mathbf{x}}-\underline{\mathbf{x}})\right|}+1}<1 \qquad (8)$$

In relation (8) $\phi_1=c_1r_1$, $\phi_2=c_2r_2$; $\overline{\mathbf{x}}$ e $\underline{\mathbf{x}}$ are the upper and lower bounds of the search space; $\phi_{1\max}= c_1$, $\phi_{2\max}= c_2$ and a is a user-defined constant value acting as a trigger in the elimination of the cognitive component. The result obtained with the *switch* implementation is shown in Figure 3. This tool allows almost all the particles to reach the optimal zone, and consequently a much more homogeneous Pareto front is obtained.

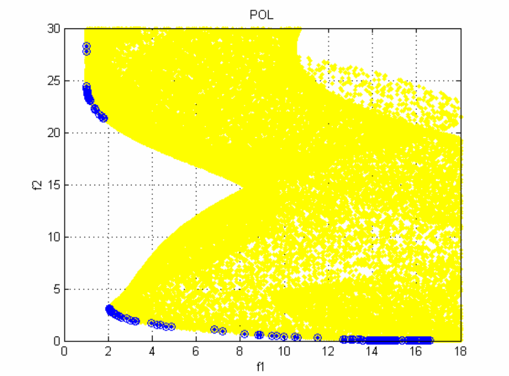
3.2. Advances in Pareto front finding: sampling density

In multi-objective optimization problems, the use of relation (2) determines the need for a criterion affecting the choice of the global best position and the personal best position. We refer to the global best position as the place of the current Pareto front that has the minimal distance from the particle whose velocity is being updated. To avoid particles crowding in the same regions, a sampling density threshold is introduced. This approach resembles in some ways ranking methods used in GAs-based multi-modal analysis. The sampling threshold establishes the maximum number of individuals that can be attracted by each known global best position. When a global best position counter reach the maximum allowed density, particles potentially in the same neighbourhood have to move to another global best position. The density implementation purpose is to find Pareto fronts as more homogeneous as possible. The effect of this parameter can be retrieved in Figure 4, where the objective space and the Pareto front are referred to the classical problem POL, available in literature[12].

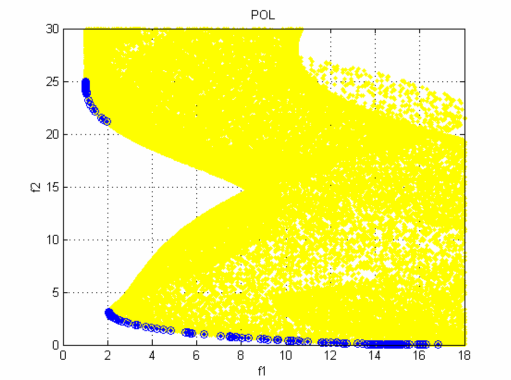
3.3. Recovery and turbulence operators

To increase Pareto front consistency once again, two further mechanisms have been introduced in MOPSO procedure: *recovery* and *turbulence* operators.

Recovery consists of a mechanism that allows particles to recover their own cognitive component that was removed by switch operator, once a

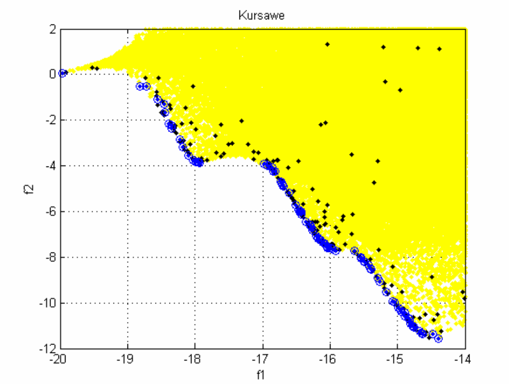


a)

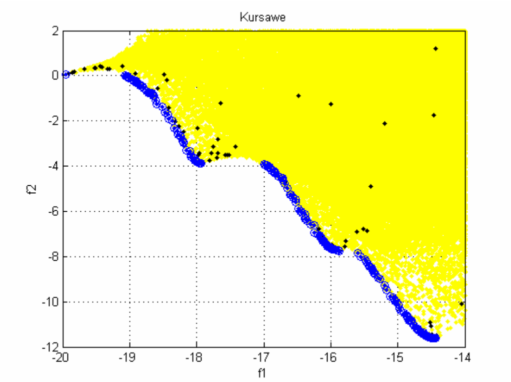


b)

FIG. 4 MOPSO application to POL problem: a) without sampling density implementation; b) with sampling density implementation



a)



b)

FIG. 5 MOPSO application to KURSAWE problem: a) 250 particles without archives; b) 100 particles with archives neighbourhood of a global best solution is reached. Turbulence[6] is an operator that introduces a random additional contribution to the particle velocity, when the mutual distance from the Pareto front is less than a specific threshold value. The magnitude of this additional contribution is of the same order of the particle distance from the Pareto front.

3.4. Memory expansion: archives

One of PSO specific features is the particle tendency to move around the global best position once it has reached the Pareto front. In this way, each particle is able to discover further solutions belonging to the front. The capability to hold these new positions greatly help in finding more homogeneous and wider fronts, also when less extensive populations are used. To keep track of equivalent solutions, we can use local arrays (archives) instead of simple memory locations[13,14]. In this context, we use two types of archives: the first archive is used to store the sequence of global best positions; the other one stores the sequence of personal best positions. The global best position is chosen as previously stated (minimum distance from the particle) until the number of archived global best positions is less than the swarm size. When the number of stored global best positions exceeds it, the more isolated particle (i.e. the global best position having the largest minimum distance to any other particle) is chosen to update velocity. We can use three different strategies for personal best selection: *i*) The *oldes*t personal best position; *ii*) The *newest* personal best position; *iii*) the *more isolated* personal best position.
The effectiveness of these criterions strictly depends on the specific optimization problem and therefore a proper selection should be made to obtain an improved Pareto front. Strategy *iii*) was selected by authors to solve all the following optimization problems.
Figure 5 shows results obtained by using the MOPSO methodology to solve Kursawe problem[15]. In particular, Figure 5-a shows the Pareto front obtained with a population size of 250 individuals without archives. Figure 5-b shows the Pareto front obtained with a reduced population of 100 individuals using archives. Figures comparison shows archives effectiveness in finding a wider and uniform Pareto front with a reduced computational cost.

4. PROBLEM DEFINITION

MOPSO procedure has been applied to define a preliminary short/medium range transport aircraft configuration powered by turbofan engines, fully compliant with given requirements. Maximum takeoff weight and block time have been selected as the two objective functions to be minimized. In order to estimate block time, block fuel and aircraft weight, a proper mission profile has been defined: 3000 km design range + 45 min extended cruise + 185 km alternate (see Figure 6).
Authors have already dealt with this problem[16] by developing an optimization procedure based on a multi-objective genetic optimizer. Therefore, we take previously obtained results as terms of comparison to evaluate MOPSO effectiveness as an alternative tool to be used in the early stage of aircraft configuration definition. Design variables are summarized in Table 1. A binary version of the basic PSO algorithm has been implemented in order to handle discrete variables too. This technique, introduced by Kennedy et al.[17] has already shown its effectiveness in previous works[4]. Constraint functions are summarized in Table 2. Proper penalty functions have been defined that degrade particles fitness whenever one or more constraints are violated.
Starting population is formed by 300 particles whereas the optimization process is stopped once the

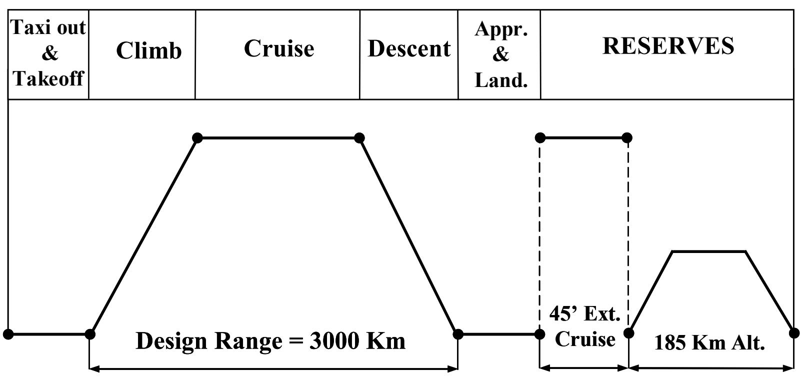


FIG. 6 Mission profile

Variable, unit	Value	
Continuous variables:	**Min**	**max**
Wing sweep, deg.	5	35
Wing t/c change	0.0	0.05
Wing area, m^2	80	130
Wing taper ratio	0.15	0.50
Wing aspect ratio	7.0	9.5
Engine thrust scaling factor (T/T_{ref})	1.0	1.5
Discrete variables:		
Takeoff flap deflection, deg.	0, 10, 15, 20	
Landing flap deflection, deg.	25, 30, 35, 40	
Configuration index	1, 2, 3, 4 [a]	
Cruise altitude, Flight Level (FL)	290, 300, 310, 320, 330, 340, 350, 360	

[a] 1 = 5 abreast, fuselage mounted engines
2 = 6 abreast, fuselage mounted engines
3 = 5 abreast, wing mounted engines
4 = 6 abreast, wing mounted engines

TAB. 1 Design variables

Constraints function, unit	Allowable value	
Rate of climb at cruise altitude, m/s	≥	1.5
Balanced field length, m	≤	1830
Landing field length, m	≤	1525
Approach speed, km/h	≤	240
Cruise range/Design range	≥	0.5
2nd segment climb gradient	≥	0.024
Mission fuel/ Max fuel capacity	≤	1.0
Wing tip chord, m	≥	1.0

TAB. 2 Constraint functions

200th iteration is reached. As for optimization control parameters, cognitive and social parameter are 2.8 and 1.3 respectively, inertia weight is 0.8, hold constant during all the optimization process, whereas k parameter, used to define constriction factor[9], is 0.95. As for personal best selection strategy, the more isolated one has been used.

5. RESULTS

Figure 7 shows Pareto curve we get at the end of the optimization process. Comparison with previous results[16] obtained with a genetic-based multi-objective optimizer (MOGA) shows MOPSO capability to define a wider Pareto front with a uniform particles distribution. MOPSO solutions appear quite similar to genetic ones for block time higher than 3.9 hrs. For lower values, Pareto curve appears wider and uniform, moreover MOPSO solutions are significantly better than genetic ones. Figures 8, 9 show examples of configuration evolution along Pareto curve.
In particular, Figure 8 shows wing planform evolution from the slowest configuration to the fastest one, whereas Figure 9 shows wing relative

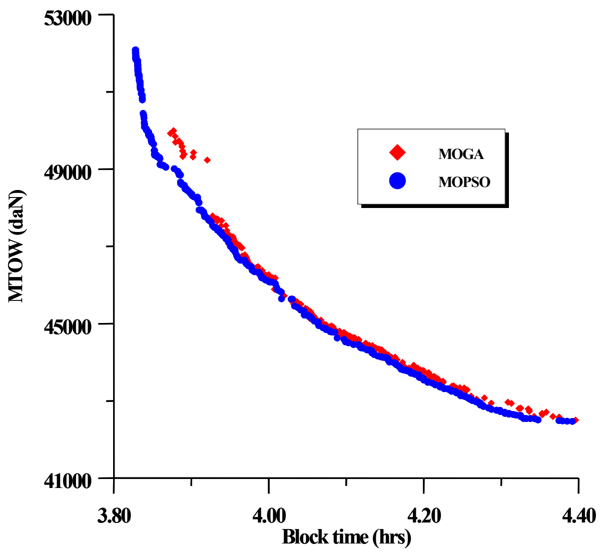


FIG. 7 Comparison between Pareto curves obtained by MOPSO procedure and a genetic-based multi-objective optimizer (MOGA)

thickness distribution related to the max cruise speed and min cruise speed solution. Min DOC configuration has the same thickness distribution as the max cruise speed one. Figure 10 shows some examples of sensitivity curves we can get with a multi-objective optimization procedure.
These curves show how aircraft main characteristics are affected by the selected requirement. In particular, cruise speed effect on operating empty weight (OEW), direct operating cost (DOC), fuel burned and engine thrust level is shown. Such a type of analysis provides the designer with useful information concerning aircraft configuration evolution and it can be used as a very effective tool aimed at the final freeze of requirements.
Moreover, to evaluate MOPSO effectiveness in sampling design domain and final solutions reliability as well, three specific solutions (i.e. max cruise speed, min DOC and min cruise speed solution) have been compared to the similar ones provided by the above mentioned MOGA procedure. Though solutions provided by both multi-objective optimizers lie on the same trend lines, MOPSO capability in finding better solutions for block time lower than 3.9 hrs (i.e. cruise speed higher than 850 km/h) is shown again.
Main configuration data are summarized in Table 3. As this procedure provides evolution along Pareto curve of constraint functions too, we can identify the most demanding ones concerning aircraft sizing. Table 3 summarizes these critical requirements in italic.
As we can see, all three configurations lie on the boundary of fuel volume as well as second segment

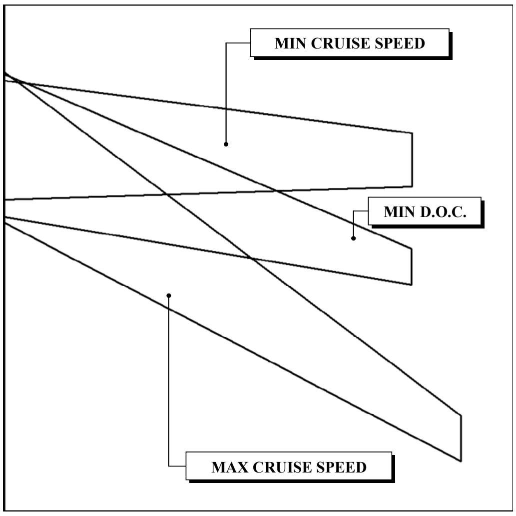


FIG. 8 Wing planform evolution along Pareto curve

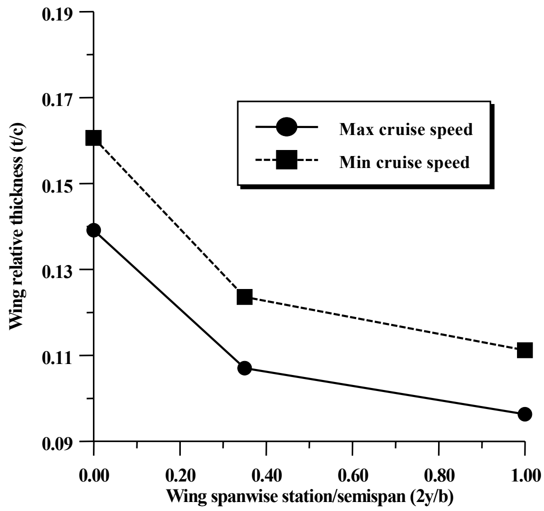


FIG. 9 Wing relative thickness related to max cruise and min cruise speed optimum configuration

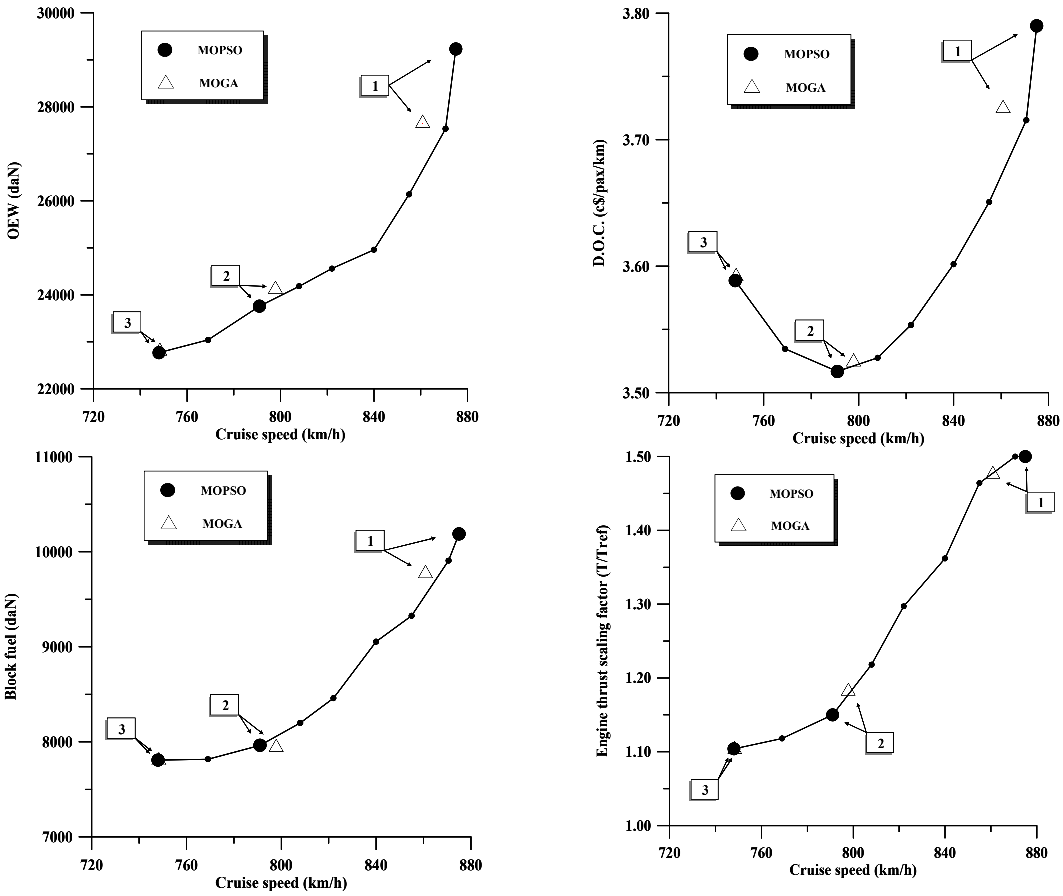


FIG. 10 Examples of cruise speed effect on aircraft characteristics (OEW, DOC, block fuel and engine thrust scaling factor). Comparison between max cruise speed (1), min DOC (2) and min cruise speed (3) optimized configurations obtained by MOPSO procedure and a genetic-based multi-objective optimizer (MOGA)

climb gradient requirement whereas they are fully compliant with landing distance and residual rate of climb requirement.

6. CONCLUSIONS

In this paper, a multi-objective optimization procedure, based on Particle Swarm algorithm, has

Design variable, unit	① Max cruise speed	② Min DOC	③ Min cruise speed
Wing sweep, deg.	34.9	20.0	5.0
Average wing relative thickness, t/c	0.107	0.107	0.124
Wing area, m^2	106.8	87.0	84.3
Wing taper ratio	0.306	0.247	0.455
Wing aspect ratio	9.28	9.05	9.38
Engine thrust scaling factor (T/T_{ref})	1.500	1.150	1.104
Takeoff flap deflection, deg.	10	10	10
Landing flap deflection, deg.	40	40	40
Configuration index	3	3	3
Cruise altitude, Flight Level (FL)	300	340	340
Main characteristics, unit			
Max takeoff weight , daN	52090	43715	42466
Operatine empty weight, daN	29231	23759	22767
Block fuel, daN	10188	7963	7808
Cruise speed, km/hr	875	791	748
Block time, hr	3.828	4.184	4.392
DOC, c$/pax/km	3.790	3.517	3.589
Constraint functions, unit			
Mission fuel/Max fuel capacity	*0.997*	*0.995*	*0.992*
Balanced field length, m	1764	*1825*	1771
Landing field length, m	1418	1332	1271
2^{nd} segment climb gradient	*0.0242*	*0.0255*	*0.0241*
Approach speed, km/h	*240*	230	220
Rate of climb at cruise altitude, m/s	7.82	3.86	3.52

TAB. 3 MOPSO optimized configurations main data

been developed (MOPSO). Preliminary application of this procedure to test-case functions has allowed to select a proper set of optimization control parameters. In addition, this analysis has allowed to choose the most suitable selection strategy concerning global best as well as local best position.
Specific tools have been developed (e.g. the switch) to improve the algorithm effectiveness in finding a Pareto front as wide and uniform as possible. The optimization procedure has then been applied to the preliminary definition of a civil transport aircraft configuration. Maximum takeoff weight and block time have been selected as the two objective functions to be minimized. Results have showed optimization procedure effectiveness to define a satisfactory Pareto front. At the end of the optimization process different sensitivity curves can also be obtained that show how aircraft main characteristics (empty weight, fuel burned, direct operating costs, etc.) change when different cruise speed requirement is selected. These curves provide useful information concerning the selected requirement effect on the aircraft configuration, allowing to reduce uncertainties about cruise speed final definition. In addition, this optimization procedure provides evolution of constraint functions along Pareto curve. Therefore, it is possible to show the most demanding requirements concerning aircraft configuration sizing.
Finally, a comparison with results previously obtained by a genetic-based multi-objective optimization technique has been performed. This comparison indicates MOPSO capability to provide quite similar solutions for block time higher than 3.9 hrs. For lower values, Pareto curve appears wider and uniform, moreover MOPSO solution are significantly better than genetic ones. These results, combined with Particle Swarm quite simple implementation, make MOPSO technique a very attractive tool for trade-off studies aimed at the final freeze of requirements in the conceptual design stage.

REFERENCES

[1] Kim, Y., Jeon, Y. H., Lee, D. H., "Multi-Objective and Multidisciplinary Design Optimization of Supersonic Fighter Wing", *Journal of Aircraft*, Vol. 43, No. 3, 2006, pp. 817-824.
[2] Kennedy, J., Eberhart, R. C., "Particle Swarm Optimization", Proceedings of the Fourth IEEE International Conference on Neural Networks, Perth, Australia, 1995, IEEE Service Center, pp. 1942-1948.

[3] Eberhart, R. C., Shi, Y., “Comparison Between Genetic Algorithm and Particle Swarm Optimization”, *Evolutionary Programming VII,* edited by V. W. Porto, N. Saravanan, D. Waagen, A.E. Eiben, Vol. 1447, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1998, pp. 611-616.

[4] Blasi, L., Del Core, G., “Particle Swarm approach in finding optimum aircraft configuration”, *Journal of Aircraft*, Vol. 44, No. 2, 2007, pp. 679-682.

[5] Coello, C. A. C., Lechunga, M. S., “MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization”, Proceedings of the 2002 Congress on Evolutionary Computation, part of the 2002 IEEE World Congress on Computational Intelligence, Hawaii, May 12-17, 2002, IEEE Press, pp. 1051-1056.

[6] Fieldsend, J. E., Singh, S., “A Multi-Objective Algorithm based upon Particle Swarm Optimisation, an Efficient Data Structure and Turbulence”, Proceedings of UK Workshop on Computational Intelligence (UKCI’02), Birmingham, UK, Sept. 2-4, 2002, pp. 37-44.

[7] Hu, X., Eberhart, R., “Multiobjective Optimization Using Dynamic Neighbourhood Particle Swarm Optimization”, Proceedings of the 2002 Congress on Evolutionary Computation, part of the 2002 IEEE World Congress on Computational Intelligence, Hawaii, May 12-17, 2002, IEEE Press.

[8] Parsopoulos, K. E., Vrahatis, M. N., “Particle Swarm Optimization Method in Multiobjective Problems”, Proceedings of the 2002 ACM Symposium on Applied Computing (SAC 2002), 2002, pp. 603-607.

[9] Engelbrecht, A., P., “Basic Particle Swarm Optimization”, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, U.K., 2005, pp. 93-129.

[10] Engelbrecht, A., P., “Multi-Objective Optimization”, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, U.K., 2005, pp. 41-47.

[11] Deb K., “Multi-objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems”, *Evolutionary Computation Journal 7*(3), 1999, pp. 205-230.

[12] Poloni, C., Giurgevich, A., Onesti, L., Pediroda, V., “Hybridization of a multiobjective genetic algorithm, a neural network and a classical optimizer for complex design problem in fluid dynamics”, *Computer Methods in Applied Mechanics and Engineering* 186(2-4), 2000, pp. 403-420.

[13] S. Mostaghim, S., Teich, J., “Strategies for finding good local guides in multi-objective particle swarm optimization (mopso), IEEE 2003 Swarm Intelligence Symposium, 2003.

[14] Jürgen Branke, Sanaz Mostaghim, “About Selecting the Personal Best in Multi-objective Particle Swarm Optimization”, *Parallel Problem Solving from Nature (PPSN)*, Vol. 4193, Lecture Notes in Computer Science, Springer, Reykjavik, Island, 2006, pp. 523-532.

[15] Kursawe F., “A Variant of Evolution Strategies for Vector Optimization”, in Schwefel H.P., Männer R., eds., *Parallel Problem Solving from Nature*, 1st Workshop, PPSN I, Vol. 496, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1991, pp. 193-197.

[16] Blasi L., Iuspa L., Del Core G., “Speed-sensitivity analysis by a genetic multi-objective optimization technique”, *Journal of Aircraft*, Vol.39, No. 6, 2002, pp. 1076-1079.

[17] Kennedy, J., Eberhart, R. C, Shi, Y., “A Model of Binary Decision”, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, 2001, pp. 289-309.