JOURNAL OF AIRCRAFT
Vol. 44, No. 2, March–April 2007

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Particle Swarm Approach in Finding Optimum Aircraft Configuration

L. Blasi* and G. Del Core†
Seconda Università degli Studi di Napoli, 81031 Aversa, Italy

DOI: 10.2514/1.24399

I. Introduction

OPTIMIZATION techniques based on *swarm theory* (particle swarm optimization, PSO), originally developed to simulate the social behavior of animals [1], are nonlinear methods belonging to the class of evolutionary computational techniques, like genetic algorithm (GA). Both of them are population-based optimizers that find solution through a probabilistic search process only guided by a fitness function value. In the particle swarm approach, however, neither individuals (or particles) are replaced nor new offspring are generated during evolution: individuals stay alive and simply change their position within the search space during the optimization process. Such position change (named *velocity*) is guided by personal experience of each particle and by swarm experience shared among all the individuals. To this end, each particle keeps a memory of its own best position as well as of the best position reached so far by all the particles. Following the PSO approach, we can get population evolution through cooperation rather than competition among individuals.

The main advantage of the PSO algorithm is its simpler implementation compared to GA: besides velocity, no additional operator is requested in PSO, whereas GA needs three different operators (selection, crossover, mutation). Many reports compare particle swarm to genetic algorithm showing, in some cases, that PSO can give better results [2,3]. For this reason, PSO applicability to an aircraft conceptual design has been deemed an interesting problem to be investigated. The definition of a preliminary aircraft configuration is, in fact, a critical task for deterministic approaches, whereas probabilistic-type methods (e.g., GA) have already proved their effectiveness [4,5].

II. PSO Algorithm

In a D-dimensional search space, each particle can be represented by a D-dimensional vector $\boldsymbol{X} = (x_1, x_2, \ldots, x_i, \ldots, x_D)^T$ and its velocity can be also represented by another D-dimensional vector $\boldsymbol{V} = (v_1, v_2, \ldots, v_i, \ldots, v_D)^T$.

Received 3 April 2006; revision received 4 December 2006; accepted for publication 21 December 2006.

*Senior Research Scientist, Dipartimento di Ingegneria Aerospaziale e Meccanica, Via Roma 29.
†Associate Professor, Dipartimento di Ingegneria Aerospaziale e Meccanica, Via Roma 29. Senior Member AIAA.

At time step $k + 1$, the ith variable will be updated according to the following equation:

$$x_i(k + 1) = x_i(k) + v_i(k + 1) \tag{1}$$

where velocity v_i is calculated as

$$v_i(k + 1) = w(k)v_i(k) + c_1\rho_1(k)[p_{l,i}(k) - x_i(k)] + c_2\rho_2(k)[p_{g,i}(k) - x_i(k)] \tag{2}$$

In Eq. (2) w is the inertia weight, $p_{l,i}(k)$ and $p_{g,i}(k)$ are the ith variable values related, respectively, to the particle best position (local optimum) and population best position (global optimum), ρ_1 and ρ_2 are random numbers uniformly distributed over the range [0,1]. Proper selection of the so-called cognitive parameter c_1 and social parameter c_2 allows improved algorithm efficiency in finding an optimum solution [6–8]; a commonly used value for their sum is 4.

As for inertia weight w, a large value favors swarm global search capability whereas a small value favors swarm local search capability. Better results have been obtained by selecting a larger value at the beginning followed by a gradual reduction during evolution [9,10]. Moreover, velocity needs to be limited to avoid the fact that particles move outside the allowable search space. In particular, previous studies [11] report that PSO algorithm performs better if such velocity limitation is defined in terms of the dynamic range of variables. However, maximum velocity as well as inertia weight does not assure that swarm particles stay within the design domain, so that a further control on the variables value has been added. In particular, two different boundary conditions [12] have been tested: whenever a particle crosses the solution space boundary in one of the dimensions, velocity in that dimension is either zeroed (absorbing wall technique) or reversed to reflect the particle back inside the design domain (reflecting wall technique).

III. Binary PSO Algorithm

Velocity evaluation by Eq. (2) is the core of the PSO algorithm; however, this formula is well suited to handle continuous variables. To process discrete variables too, a binary PSO algorithm has been implemented. This technique, introduced by Kennedy and Eberhart [13], is based on the following concept: the algorithm encodes each discrete variable in a string of binary numbers and then it operates on each bit of the binary string to evaluate velocity. In this case, however, bit velocity is used as an argument of a sigmoid function whose value gives the probability that each bit takes on the value one or zero. At time step $k + 1$, the jth bit value related to the binary representation of the ith discrete variable will be updated as follows:

$$\psi_i^j(k + 1) = \begin{cases} 0 & \text{if } \rho_{i,j}(k) \geq s[v_i^j(k + 1)] \\ 1 & \text{if } \rho_{i,j}(k) < s[v_i^j(k + 1)] \end{cases} \tag{3}$$

where $\rho_{i,j}(k)$ is a random number uniformly distributed over the range [0,1], $v_i^j(k + 1)$ is the jth bit velocity, and $s(v)$ is the sigmoid function

$$s(v) = \frac{1}{1 + e^{-v}} \tag{4}$$

To avoid sigmoid saturation, original authors of this technique recommend to clamp a velocity value in the range $[-4, 4]$. One of the

Table 1 Design variables

Variable, unit	Value	
Continuous variables:	Min	Max
Wing sweep, deg	10	30
Wing t/c change	0.0	0.05
Wing area, m^2	80	130
Wing taper ratio	0.15	0.35
Wing aspect ratio	7.0	9.5
Engine thrust scaling factor (T/T_{ref})	1.0	1.3
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.

Table 2 Constraint functions

Constraint function, units	Allowable value
Rate of climb at cruise altitude, m/s	≥ 1.5
Balanced field length, m	≤ 1676
Landing field length, m	≤ 1372
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

advantages of this technique, compared to GA, is the possibility to handle effectively a very short binary string, even a one-bit-coded discrete variable.

IV. Problem Definition

The PSO procedure has been applied to define a preliminary short/medium range aircraft configuration, fully compliant with given requirements, that allows a minimum direct operating cost (DOC). To estimate block time, block fuel, and aircraft weight, a proper mission profile [4] has been defined with the following main characteristics: design range of 2963 km in addition to typical civil aircraft reserves (45 min extended cruise +185 km alternate).

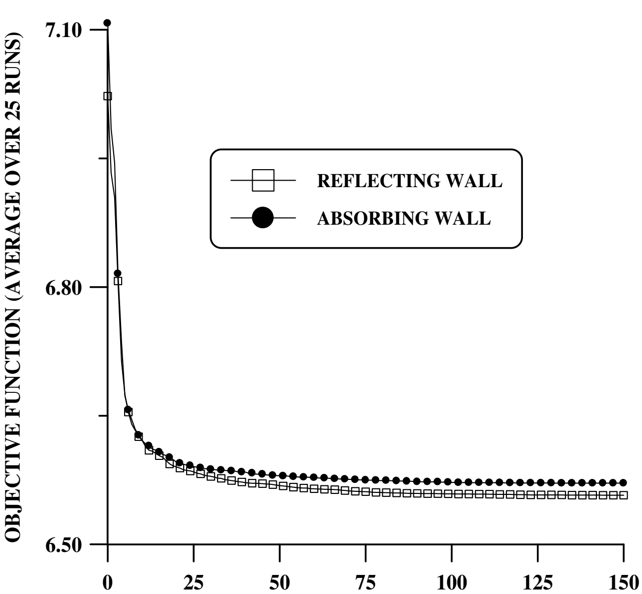

Fig. 1 Effect of different boundary conditions on objective function evolution (average over 25 runs).

The authors have already dealt with this problem by developing an optimization procedure based on a multiconstraint genetic optimizer. Therefore, previously obtained results [4] have been used as terms of comparison to evaluate the effectiveness of the particle swarm optimization technique. Design variables as well as constraint functions are summarized in Tables 1 and 2. Proper penalty functions have been defined that degrade fitness value whenever one or more constraints are violated.

V. Results

Before starting the optimization procedure, some parameters must be fixed. The weight factor linearly varies from 0.9 at the beginning to 0.4 at last iteration; for both cognitive and social parameters, the value 2 has been selected; a population size of 50 has been used; the maximum allowable velocity in each dimension has been set equal to 50% of the dynamic range of that dimension. The optimization process stops once the 150th iteration is reached.

Because PSO is a probabilistic-type technique, to obtain a more reliable result 25 optimization tasks have been performed starting from randomly generated swarms. As described earlier, two different boundary conditions have been tested: reflecting wall and absorbing wall. Figure 1 compares average objective function evolutions obtained over 25 runs with these two conditions. As we can see, the reflecting wall technique provides a slightly better performance (lower objective function value), so that all of the next results are obtained using this kind of boundary condition.

Figure 2 shows some examples of best run results. Figures 2a and 2b show that a correct evolution toward the optimum occurs with the selected number of iterations. In fact, whereas an objective function (i.e., DOC) converges rapidly to the minimum value, the swarm average objective function keeps a higher value for a longer time: this means that particles are exploring new areas of design space trying to find a better solution. Of course, when particles converge to the best position, the swarm average objective function approaches the minimum value. Figures 2c and 2d show the most critical constraint functions: balanced field length and mission fuel/maximum fuel capacity ratio. As we can see, penalty functions are very effective in keeping an optimum solution on the boundary of the allowable region.

The main data of the solution allowing minimum DOC (best solution) are reported in the first column of Table 3. Average value and standard deviation of continuous variables, calculated over 25 runs, are given in columns 2 and 3. As for discrete variables, the selection rate over 25 runs is reported. As we can see, the average solution is very close to the best one; this result means that the PSO algorithm is able to provide very similar solutions, that is, to define the most promising area within the design space. The main data of the optimum solution, obtained with a genetic optimizer [4], are reported in the last column. Comparison with the particle swarm best solution indicates that both configurations have quite the same characteristics, thus proving PSO effectiveness in finding a global minimum.

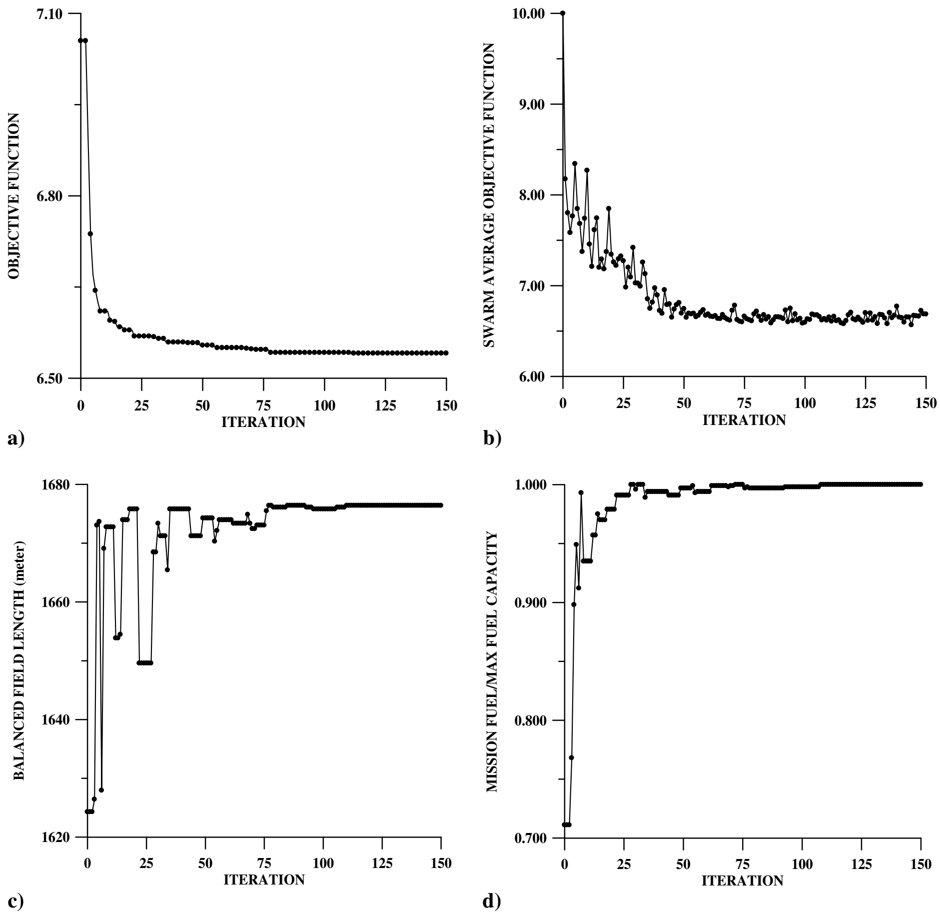

Fig. 2 Some examples of best run results: a) objective function, b) swarm average objective function, c) balanced field length, and d) mission fuel/max. fuel capacity ratio evolution.

VI. Conclusions

The potential of the PSO technique in the field of conceptual design has been investigated in this Note. We have tested two different techniques to control variable values: reflecting wall and absorbing wall technique. Comparison between the objective function evolutions shows that the reflecting wall technique provides a slightly better performance (lower objective function value). Moreover, a binary PSO variant has been implemented, which is capable of handling simultaneously continuous and discrete

Table 3 Particle swarm and genetic optimized configuration main data comparison

	Particle swarm optimizer			Genetic optimizer
	Best value	Average value (25 runs)	Standard deviation (25 runs)	
Continuous design variables, unit:				
Wing sweep, deg	21.9	20.9	2.0	21.4
Wing t/c change	0.0	2.0e-04	5.0e-04	0.0
Wing area, m^2	89.9	90.0	1.085	89.8
Wing taper ratio	0.271	0.285	2.73e-02	0.274
Wing aspect ratio	9.50	9.47	4.53e-02	9.46
Engine thrust scaling factor	1.241	1.238	0.023	1.238
Discrete design variables, unit:		Selection rate		
Takeoff flap deflection, deg	15	10	16%	15
		15	84%	
Landing flap deflection, deg	40	40	100%	40
Configuration index	3	3	100%	3
Cruise altitude, FL	350	340	20%	350
		350	60%	
		360	20%	
Main characteristics, unit:				
Max. takeoff weight, daN	44,867	44,774	276	44,788
Operating empty weight, daN	24,760	24,687	221	24,695
Block fuel, daN	8056	8044	70	8049
Cruise Mach number	0.759	0.756	5.06e-03	0.757
Mission fuel/max. fuel capacity	1.000	0.999	1.25e-03	0.999
Balanced field length, m	1676	1676	2.93e-01	1674
Landing field length, m	1339	1330	13.7	1335
Objective function (DOC), c$/nm/pax	6.541	6.552	1.19e-02	6.549

variables. This feature gives the optimization procedure a very useful flexibility in aircraft configuration as well as mission profile modeling. The analysis of 25 different optimization tasks indicates that the PSO technique, although unable to provide a single optimum solution, is able to explore effectively design space and identify the most promising design variables set. Finally, comparison between the best solution obtained with a genetic optimizer and PSO best solution shows that both configurations have quite the same characteristics. The effectiveness of the PSO algorithm, combined with its simpler implementation compared to genetic algorithm, makes the PSO technique a very attractive tool to perform aircraft preliminary configuration selection and sizing.

References

[1] Kennedy, J., and Eberhart, R. C., "Particle Swarm Optimization," *Proceeding of the 1995 IEEE International Conference on Neural Networks*, IEEE, Piscataway, NJ, 1995, Vol. 4, pp. 1942–1948.

[2] Kennedy, J., and Spears, W. M., "Matching Algorithms to Problems: An Experimental Test of the Particle Swarm and Some Genetic Algorithms on the Multimodal Problem Generator," *Proceeding of the 1998 International Conference on Evolutionary Computation*, IEEE, Piscataway, NJ, 1998, Vol. 1, pp. 78–83.

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

[4] Blasi, L., Iuspa, L., and Del Core, G., "Conceptual Aircraft Design Based on a Multiconstraint Genetic Optimizer," *Journal of Aircraft*, Vol. 37, No. 2, 2000, pp. 350–354.

[5] Blasi, L., Iuspa, L., and Del Core, G., "Speed-Sensitivity Analysis by a Genetic Multiobjective Optimization Technique," *Journal of Aircraft*, Vol. 39, No. 6, 2002, pp. 1076–1079.

[6] Kennedy, J., "The Behaviour of Particles," *Evolutionary Programming VII*, edited by V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben, Vol. 1447, Lecture Notes in Computer Science, Springer–Verlag, Berlin, 1998, pp. 581–589.

[7] Kennedy, J., Eberhart, R. C., and Shi, Y., "Variation of the Particle Swarm Paradigm," *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, 2001, pp. 329–361.

[8] Parsopoulos, K. E., and Vrahatis, M. N., "Recent Approach to Global Optimization Problems Through Particle Swarm Optimization," *Natural Computing*, Vol. 1, Nos. 2–3, 2002, pp. 235–306.

[9] Shi, Y., and Eberhart, R. C., "Parameter Selection in Particle Swarm Optimization," *Evolutionary Programming VII*, edited by V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben, Vol. 1447, Lecture Notes in Computer Science, Springer–Verlag, Berlin, 1998, pp. 591–600.

[10] Shi, Y., and Eberhart, R. C., "Empirical Study of Particle Swarm Optimization," *Proceeding of the 1999 Congress of Evolutionary Computation*, IEEE, Piscataway, NJ, 1999, Vol. 3, pp. 1945–1950.

[11] Eberhart, R. C., and Shi, Y., "Particle Swarm Optimization: Developments, Application, and Resources," *Proceeding of the 2001 Congress on Evolutionary Computation CEC2001*, IEEE, Piscataway, NJ, 2001, Vol. 1, pp. 81–86.

[12] Robinson, J., and Rahmat-Samii, Y., "Particle Swarm Optimization in Electromagnetics," *IEEE Transactions on Antennas and Propagation*, Vol. 52, No. 2, 2004, pp. 397–407.

[13] Kennedy, J., Eberhart, R. C., and Shi, Y., "A Model of Binary Decision," *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, 2001, pp. 289–309.