JOURNAL OF PROPULSION AND POWER
Vol. 20, No. 3, May–June 2004

Multicriteria Design Optimization of a Supersonic Inlet Based upon Global Missile Performance

A. Gaiddon*
Matra Bae Dynamics Aerospatiale Alenia France, F-92323 Châtillon, France
D. D. Knight†
Rutgers—The State University of New Jersey, Piscataway, New Jersey 08854-8058
and
C. Poloni‡
Università di Trieste, 34127 Trieste, Italy

The flight performance of a ramjet-powered missile is improved through the use of an automated optimization loop relying on computational-fluid-dynamics tools. A generic supersonic airbreathing missile is first described, and its performance is assessed for a representative mission using Reynolds-averaged Navier–Stokes computations for aerodynamics prediction and theoretical engine performance models. The loop links together an optimization algorithm with an aerodynamic software computing the aerodynamic balance of the missile. Several optimizations are performed using different global algorithms such as simplex, evolutionary strategies, or genetic algorithms. The first ones are mono-objective: for each point of the mission (acceleration, cruise, and maneuver), the best inlet shape has to be found. Then multiobjective optimizations are performed in order to find the pareto front, that is, the best set of shapes satisfying the whole mission.

Nomenclature

$A^\star$	=	critical area, m^2
C_A	=	axial-force coefficient
C_N	=	normal-force coefficient
C_Π	=	thrust coefficient
$C^\star$	=	characteristic speed, $m \cdot s^{-1}$
D	=	dynalpy, Pa
f_{sto}	=	stochiometric coefficient
Kp_b	=	base-pressure coefficient
M	=	Mach number
m	=	air mass flow, $kg \cdot s^{-1}$
m_k	=	kerosene mass flow, $kg \cdot s^{-1}$
$N_{x,z}$	=	acceleration factors
P	=	pressure, Pa
Pi	=	total pressure, Pa
q	=	dynamic pressure, Pa
S_R	=	reference section, m^2
T	=	temperature, K
Ti	=	total temperature, K
(X, Z)	=	missile axis
(x, z)	=	air axis
α	=	angle of attack, deg
ϵ	=	mass flow rate coefficient
η_c	=	combustion efficiency
η_N	=	nozzle total pressure recovery
η_{02}	=	air-intake total pressure recovery
η_{24}	=	combustor total pressure recovery
φ_b	=	burnt equivalent ratio
φ_i	=	injected equivalent ratio

Subscripts

b	=	base
0	=	freestream conditions
1	=	capture section
2	=	end of diffuser
3	=	end of combustor
4	=	nozzle throat
5	=	nozzle exit

Presented as Paper 2002-5546 at the AIAA/ISSMO 9th Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, 4 September 2002; received 12 May 2003; revision received 5 August 2003; accepted for publication 5 August 2003.

*Engineer, Vehicle and Propulsion Department. Member AIAA.
†Professor, Department of Mechanical and Aerospace Engineering, 98 Brett Road. Associate Fellow AIAA.
‡Professor, Dipartimento di Energetica, Via Valerio 10.

Introduction

A PRINCIPAL component of an airbreathing propulsion system is the inlet. Its role is to provide the engine with a sufficient mass flow at high total pressure and minimum distortion.[1,2] In addition, the inlet contributes to the global lift, drag, mass, and radar cross section of the missile. The design of an inlet therefore requires consideration of multiple objectives, and the ultimate choice is a compromise between competing requirements. Assuming quantitative criteria are provided for each design objective, the collection of "best" designs is denoted the pareto set,[3] wherein improvement in one design objective is achievable only at the expense of the degradation in performance with respect to one or more other design criteria.

Three major technology developments have revolutionized the process of inlet design.[4] The first is the development of accurate, efficient, and robust simulation software for the predicting of aerodynamic, electromagnetic, and structural performance. For example, a hybrid flow simulation tool,[5] developed by Rutgers University and MBDA (France), can accurately predict the total pressure recovery and mass capture ratio for a significant portion of the performance curve of a three-dimensional supersonic inlet in a few minutes on a single-processor workstation. Also, modern Reynolds-averaged Navier–Stokes simulations using conventional two-equation turbulence models can accurately predict the entire performance envelope of a three-dimensional supersonic inlet.[6] The second is the development of efficient and robust multicriteria design optimization (MDO) algorithms[3] for automated searching of complex, multidimensional design spaces. The third is the continued rapid advances in computer and network performance. For example, processor speeds double every 18 to 24 months.[7]

Up to the present time, automated inlet design optimization studies have focused on inlet aerodynamic performance defined in terms of total pressure recovery and mass flow rate. Shukla et al.[8] performed an automated redesign of the NASA P2 and P8 hypersonic inlets to increase total pressure recovery at the outflow. Zha et al.[9] used both gradient- and stochastic-based optimizers to design a mixed-compression axisymmetric supersonic inlet for a fixed flight condition. Blaize et al.[10] performed an automated optimal design of a two-dimensional supersonic inlet for both single and multiple flight conditions and demonstrated significant improvement in total pressure recovery compared to the baseline design that had been obtained by conventional techniques. Kergaravat et al.[11] performed a MDO for a supersonic inlet combining both aerodynamic and electromagnetic performance. Gaiddon and Knight[12] performed a postdesign of the inlet for the VESTA missile and demonstrated performance equivalent to the original design with an approximate order-of-magnitude reduction in calendar time required for the design. Additional inlet design optimization studies are summarized in Knight.[13,14]

The objective of the present study is the MDO of a supersonic inlet based upon the global performance of the missile. This represents the first MDO of a supersonic inlet wherein the vehicle performance, rather than the surrogate of inlet aerodynamic performance, has been used as the optimization criteria.

Missile Test Case

General Description

The generic missile test case is representative of a small ramjet-powered missile piloted in bank to turn. It has a cylindrical fuselage equipped with wings and fins. Because the flight is mainly conducted at positive angle of attack, one inlet is located under the fuselage. A view of the missile is presented in Fig. 1. Its major geometric characteristics are given in Table 1. The inlet is on design, that is, with the shock wave formed on the ramp focusing on the cowl lip, at $M = 2.2$.

The mission of the missile begins with a rocket boost phase leading to low supersonic speed. The inlet is then opened and the ramjet ignited, and the missile accelerates to its cruise speed and altitude. The cruise phase must be optimized in order to ensure the best range. The final mission phase consists in a series of maneuvers to counter defenses before hitting the target. Although the final performance of the missile can be characterized only with six-degree-of-freedom simulations of complete trajectories involving modelization of aerodynamics, thrust and guidance, characteristic flight points must be used during the design phase. In our case the flight conditions for the ramjet system are the acceleration point where an important total pressure recovery and mass flow are needed at low Mach number and low angle of attack, the cruise point where the fuel consumption has to be minimized and the maneuver point where the total pressure recovery of the inlet has to be sufficient even at a high angle of attack. Flight conditions are summarized in Table 2. No fixed angle of attack is specified because the missile balance (lift vs mass) is sought during the optimization process.

Table 1 Geometrical characteristics of the missile test case

Characteristic	Value
Length	3.2 m
Caliber	250 mm
S_R	0.049087 m^2
Wing area	1.6 S_R
Wing span	500 mm
Fin area	0.7 S_R
Capture area	0.4 S_R
Base area	0.28 S_R

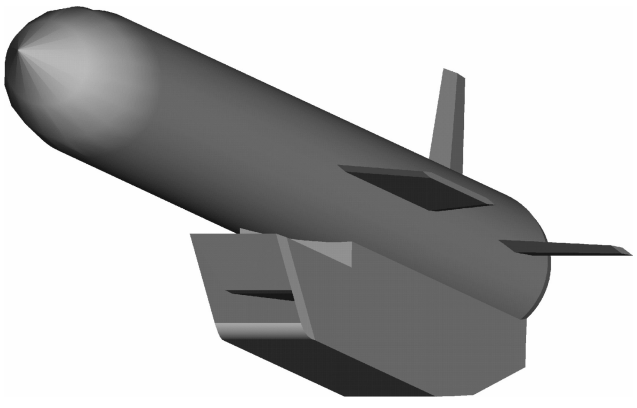


Fig. 1 View of the generic ramjet powered missile test case.

Table 2 Mission characteristic points

Flight point	Mach	Altitude, m
Acceleration	1.9	100
Cruise	2.8	15,000
Maneuver	2.2	100

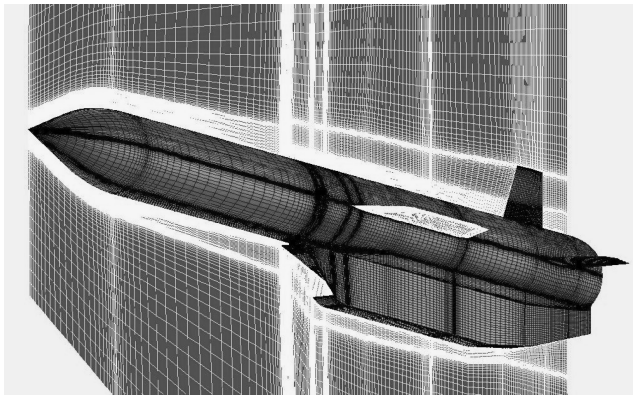


Fig. 2 View of the multidomain structured mesh.

External Flowfield of Missile

The external flowfield of the missile at each flight condition is obtained from a separate computational-fluid-dynamics (CFD) simulation. The Reynolds-averaged Navier–Stokes equations (RANS) are solved with a finite volume approach on a multidomain structured mesh using the GASPex software.[15] Inviscid fluxes are computed with the Roe scheme and Harten correction. Spatial reconstruction is third-order upwind-biased associated with a modified Essentially Non-Oscillatory (ENO) limiter. The Spalart–Almaras model is used to take into account turbulence. Convergence to the steady state is obtained with a Jacobi scheme with inner iterations. A local Courant–Friedrichs–Lewy value of 1 is used to reach convergence, which is monitored both by the numerical residual and the aerodynamic coefficients.

The mesh size is 2,840,000 cells for half a configuration, which is typical of mesh sizes used for real case studies. The mesh was built with the HEXA software[16] distributed by ICEMCFD Engineering GmbH. A view of the mesh is given in Fig. 2. Each computation requires about 130h CPU on one Compaq/Alpha EV68 500-Mega Hertz (MHz) processor. Use of MPI parallelization in GASPex enables a significant reduction in wall clock time.

A view of the result obtained for $M = 2.8, \alpha = 6$ deg is presented in Fig. 3, where pressure contours are plotted on the missile skin and the Mach-number field is plotted on the symmetry plane. One could notice the major physical phenomena occuring in this case, that is, the oblique shock on the nose, the boundary-layer development, the shocks on the wing, and the high level of pressure on the compression ramp of the inlet.

Our computations do not model the base flow, which has no influence on the global flowfield for supersonic speeds. A semi-empirical correction[17] has been applied to take into account the base drag.

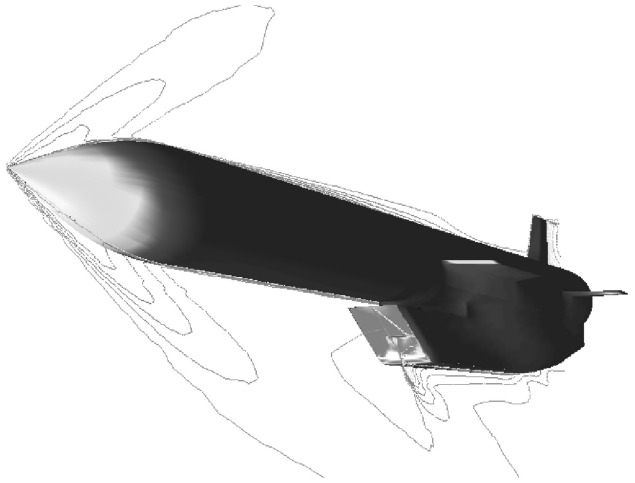


Fig. 3 Pressure repartition on the missile and Mach-number field in the symmetry plane.

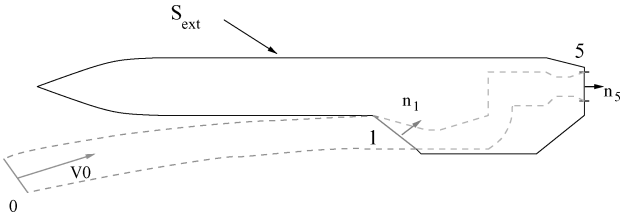


Fig. 4 Decomposition of aerodynamic forces.

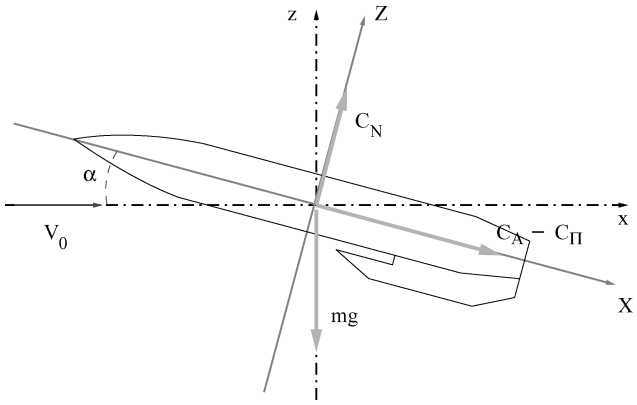


Fig. 5 Axis convention.

Aeropropulsive Balance

Forces Decomposition

For an airbreathing missile the decomposition between internal and external aerodynamics is neither direct nor easy. The inlet plays a great role in terms of global lift and drag. Moreover, when the inlet operates in underdesign regime, it creates a so-called additive drag. It is therefore important to properly decompose the internal and external forces (see Fig. 4).

First, the following convention is used in this study: forces are computed using the relative pressure. For example the pressure-force coefficient over a surface S is computed through

$$\boldsymbol{C} = \frac{\left[\int_S (P - P_0)\boldsymbol{n}\,\mathrm{d}S\right]}{(q_0 S_R)}$$

Also, the axis convention is described in Fig. 5. Two systems of axis are used: one is related to the missile, and the other one is related to the freestream airflow. One should notice that the missile velocity is always horizontal in our study, which corresponds to a slope angle equal to zero.

A common quantity used for propulsion balance is the dynalpy on a surface defined by

$$\boldsymbol{D} = (P - P_0)\boldsymbol{n} + \rho \boldsymbol{v}(\boldsymbol{v}\cdot\boldsymbol{n}) \tag{1}$$

By using the momentum equation on the streamtube going from the inlet entrance (station 1) to the nozzle exit (station 5), it can be shown that the force acting on the internal walls, that is, the inlet, the ramjet, and the nozzle is equal to the difference between the exiting dynalpy flux and the entering dynalpy flux.[18] By using the ext subscript for the external walls, that is, the fuselage, the wings, the fins, etc., the sum of forces acting on the missile is given by

$$\boldsymbol{F} = \boldsymbol{F}_{\text{ext}} + \int_{A_1} \boldsymbol{D}_1\,\mathrm{d}S - \int_{A_5} \boldsymbol{D}_5\,\mathrm{d}S \tag{2}$$

However, one can see here that the expression

$$\int_{A_1} \boldsymbol{D}_1\,\mathrm{d}S - \int_{A_5} \boldsymbol{D}_5\,\mathrm{d}S$$

is itself a balance of forces acting on internal walls. The common notions of drag and thrust can now be introduced for the internal flow. By convention, thrust is positive when it is oriented toward the front of the missile, and drag is positive when it is oriented toward the rear of the missile. It is here convenient to introduce the freestream streamtube to define the ramjet thrust as the difference of dynalpy flux between the nozzle and freestream streamtube

$$\int_{A_5} \boldsymbol{D}_5\,\mathrm{d}S - \int_{A_0} \boldsymbol{D}_0\,\mathrm{d}S$$

and the internal forces as the difference of dynalpy flux between the freestream streamtube and the inlet entrance

$$\int_{A_0} \boldsymbol{D}_0\,\mathrm{d}S - \int_{A_1} \boldsymbol{D}_1\,\mathrm{d}S$$

The freestream streamtube dynalpy flux can be computed through the mass flow rate coefficient, which leads to the expression

$$\int_{A_0} \boldsymbol{D}_0\,\mathrm{d}S = \frac{\gamma_0 P_0 M_0^2 \epsilon A_1 \boldsymbol{v}_0}{v_0}$$

If the decomposition between the thrust of the ramjet and the additive drag of the inlet has a physical sense, this is not the case for the lift. Therefore, three aerodynamic coefficients can be defined: the additive drag of the inlet, the thrust of the ramjet, and the global lift of the internal walls. By convention, the thrust coefficient C_Π is positive when the ramjet produces a thrust so that

$$C_\Pi = \frac{A_5\left[P_5\left(1+\gamma_5 M_5^2\right) - P_0\right]}{q_0 S_R} - 2\epsilon\frac{A_1}{S_R}\cos\alpha \tag{3}$$

$$C_A^{\text{int}} = \frac{\int_{A1} \boldsymbol{D}_1\,\mathrm{d}S}{q_0 S_R}\cdot\boldsymbol{X} - 2\epsilon\frac{A_1}{S_R}\cos\alpha \tag{4}$$

$$C_N^{\text{int}} = \frac{\int_{A1} \boldsymbol{D}_1\,\mathrm{d}S}{q_0 S_R}\cdot\boldsymbol{Z} \tag{5}$$

Ramjet Modelization

A standard ramjet[19] can be described by the sketch shown in Fig. 6. The different stations of the ramjet are as follows: station 0, freestream flow; station 1, entrance of the inlet; station 2, end of the subsonic diffuser; station 3, end of the combustor; station 4, nozzle throat; and station 5, end of the nozzle.

The following methodology is used to compute the ramjet performance. The first step consists in finding the freestream total conditions Pi_0 and Ti_0 from P_0, T_0, γ_0, and M_0. The air mass flow entering the inlet is known with $m_0 = \rho_0 V_0 \epsilon A_1$. For a given injected

Table 3 Data for an air–kerosene mix combustion

φ_b	γ_4	$C_4^\star$, m · s^{-1}
0	1.380	583
0.2	1.323	820
0.4	1.292	982
0.6	1.273	1106
0.8	1.259	1202
1	1.254	1260

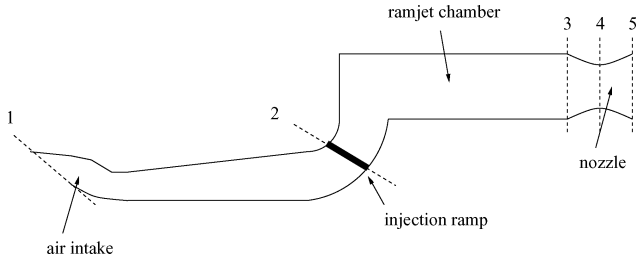


Fig. 6 Notation used for the ramjet modelization.

equivalent ratio φ_i the kerosene mass flow rate is obtained through $m_k = \varphi_i f_{\text{sto}} m_0$. The burnt equivalent ratio depends from the combustion efficiency $\varphi_b = \varphi_i \eta_c$. To be more realistic, a limitation has been taken on the burnt equivalent ratio, which cannot exceed 1.0, a typical limitation for liquid-ramjet engine. If this is the case, the performance of the missile is limited by the ramjet. The air–kerosene combustion process is known through the data of the characteristic speed $C_4^\star$ and of γ_4 as functions of φ_b. For our study the values are linearly interpolated in the data shown in Table 3.

The next step consists in computing the flow at station 4, that is, the nozzle throat, where the flow is sonic. The critical area $A_4^\star$ is given by the relation $\Sigma(M_4) = A_4/A_4^\star$, where $\Sigma(M)$ is the function:

$$\Sigma(M) = \frac{1}{M}\left[\frac{2 + (\gamma - 1)M^2}{\gamma + 1}\right]^{(\gamma+1)/2(\gamma-1)} \tag{6}$$

Because $M_4 = 1$, $A_4^\star = A_4$. Using the definition of $C_4^\star$,

$$C_4^\star = Pi_4 A_4^\star / q_4 \tag{7}$$

leads to the expression giving the total pressure at the nozzle throat:

$$Pi_4 = \frac{m_0 \cdot (1 + \varphi_i f_{\text{sto}}) \cdot C_4^\star \cdot \Sigma(1)}{A4} \tag{8}$$

It is here necessary to verify that the injected equivalent ratio φ_i is consistent with the maximum total pressure recovery η_{02} of the inlet. Because η_{24} is the combustor total-pressure-recovery coefficient, the relation $\eta_{02} = Pi_4/(Pi_0\eta_{24})$ gives the value of η_{02} needed for the already computed Pi_4. If this value is superior to the critical total pressure recovery of the inlet, an iteration is performed on φ_i to obtain the maximum value conducting to the critical regime. In this case the maximum thrust of the ramjet is limited by the performance of the inlet.

The last step of the ramjet computation consists in obtaining the flow condition at the nozzle exit (station 5). Pressure losses are taken into account through a constant coefficient $\eta_N = 0.98$ so that $Pi_5 = \eta_N Pi_4$, which leads to the relation $\Sigma(M_5) = A_5/A_4$ giving the value of the exit Mach number M_5. Static pressure at station 5 is obtained with

$$P_5 = Pi_5 \cdot \left\{1 + [(\gamma_5 - 1)/2]M_5^2\right\}^{-\gamma_5/\gamma_5 - 1} \tag{9}$$

This makes it possible to compute the dynalpy at station 5 and thus the thrust coefficient defined in Eq. (3).

Missile Performance

The acceleration factors N_x and N_z of the missile on its trajectory can be computed by projecting the forces on the "air axis." In the following equations C_A and C_N are the sum of their internal and external components:

$$(C_\Pi - C_A)\cos\alpha - C_N \sin\alpha = N_x(mg/q_0 S_R) \tag{10}$$

$$C_N \cos\alpha + (C_\Pi - C_A)\sin\alpha = N_z(mg/q_0 S_R) \tag{11}$$

By convention $N_x = 0$ and $N_z = 1$ in balanced flight conditions. One should also notice that C_A and C_N are functions of the angle of attack α, whereas C_Π is a function of both the angle of attack α and the injected equivalent ratio φ_i. The evolution of these functions are given by the aerodynamic model and the ramjet model of the missile.

In our case the optimization of the missile performance is different for each flight condition. For the acceleration flight condition the maximum x-acceleration factor in horizontal flight has to be found. Therefore, writing $N_z = 1$ leads to

$$C_N \cos\alpha + (C_\Pi - C_A)\sin\alpha = mg/q_0 S_R \tag{12}$$

Because the maximum acceleration factor has to be reached, the ramjet is set up at maximum thrust and is limited either by the inlet total pressure recovery or by the maximum burnt equivalent ratio $\varphi_b = 1$. Solving the last equation makes it possible to find the balanced flight angle of attack α. Then the acceleration factor is given by

$$N_x = (q_0 S_R/mg)[(C_\Pi - C_A)\cos\alpha - C_N \sin\alpha] \tag{13}$$

For the maneuver condition the best z-acceleration factor has to be reached with no deceleration. $N_x = 0$ leads to the relation $\tan\alpha = (C_\Pi - C_A)/C_N$. In this case also, maximum thrust has to be applied. Solving this equation gives the balanced flight angle of attack. Finally, the acceleration factor is computed through

$$N_z = (q_0 S_R/mg)[(C_\Pi - C_A)\sin\alpha + C_N \cos\alpha] \tag{14}$$

For the cruise flight condition the objective is to minimize the fuel consumption. The balance of the missile leads to $N_x = 0$ and $N_z = 1$ so that the following equation, independent of C_Π

$$C_N(\alpha) = (mg/q_0 S_R)\cos\alpha \tag{15}$$

leads to the cruise angle of attack. Then the second relation $C_\Pi = C_A + C_N \tan\alpha$ gives the value of thrust needed and therefore the fuel consumption through the ramjet model.

Optimization Loop

Loop Structure

The structure of the optimization loop is presented in Fig. 7. It links together an optimizer, which gives design descriptions such as geometric parameters, and one or several evaluation tools, which give back the performance of the proposed design. The optimizer role is to find as fast as possible, that is, with the lowest cost, the best design or the best set of designs for one or several objective(s) under one or several constraint(s).

The automated loop used in this study links together a multiobjective optimization algorithm with a missile performance evaluation tool. This last one computes the balance of a generic missile for three flight conditions: acceleration, maneuver, and cruise. This computation is made using three data tables: a ramjet modelization with theoretical equations, internal aerodynamic coefficients, and external aerodynamic coefficients. Because the inlet shape varies, the internal data are automatically computed with the 2ES2D CFD tool (described later), which predicts for each flight condition of the model the mass flow rate, total pressure recovery, and the additive drag and lift of the considered inlet shape.

It is emphasized that the performance of the inlet in this optimization study is determined by the combined effects of the inlet aerodynamic performance (e.g., total pressure recovery, mass capture ratio), ramjet performance, and overall missile aerodynamics. The best missile performance does not necessarily correspond to the best isolated inlet shape but to a design leading to the best propulsive

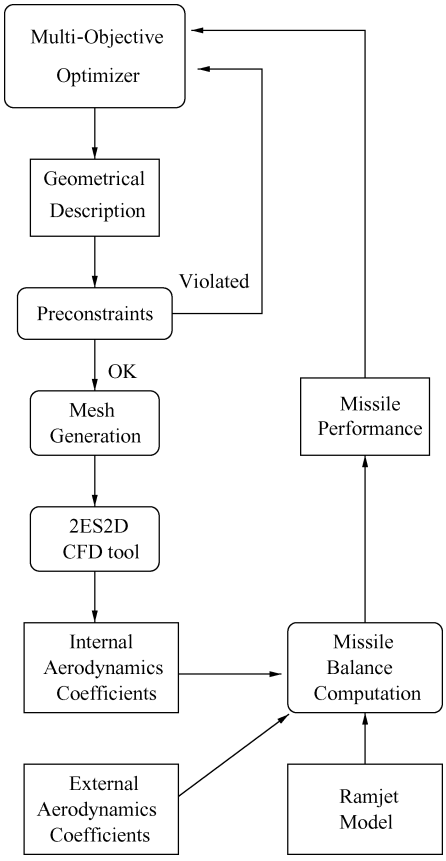


Fig. 7 Structure of the optimization loop.

performance for the inlet/ramjet aeropropulsive system on a missile design having specific external aerodynamics and mass.

ModeFRONTIER Optimization Environment

The multiobjective design environment (mode) FRONTIER software is an outcome of a European Comission ESPRIT project named Frontier, which addresses multipartner collaborative design optimization. The transfer of the knowledge acquired during this project into a commercial product was performed by the Italian company ES.TEC.O. s.r.l.* This software provides an environment dedicated to the setup of design assessment chains and efficient investigation of the design spaces. It includes graphical tools for integration of computation loops, allowing complex logics, and is thus able to address real-life cases where conditional process must be ensured. Most recent programming technologies such as Java or XML are used in order to perform remote calculations in heterogeneous environments. Several tools and algorithms are provided in order to efficiently explore and analyse the design space, such as design of experiment, response surfaces, and optimization algorithms. Response surfaces can be built using classical methods such as linear or quadratic approximation, local interpolation, or more advanced methods like Gaussian processes or neural networks. Optimization algorithms include classic deterministic local algorithms such as Sequential Quadratic Programming (SQP) or Broyden-Fanno-Fletcher-Goldfarb-Schano algorithm (BFGS) and more recent stochastic global technologies like evolutionary strategies of genetic algorithms, including a true multiobjective genetic algorithm (MOGA). New algorithms coupling interpolation techniques with optimization technologies are also available. Finally, graphical analysis tools with scatter and history plots or response surface visualization permit one to quickly understand the design space. Multicriteria decision-making tools allow the designer to define his preferences through a limited number of choices within best designs and are helpful in order to get the best compromise design.

*Data available online at http://www.esteco.it.

Optimization Algorithms

SIMPLEX

The SIMPLEX algorithm implemented in modeFRONTIER is based on the Nelder and Mead method,[20] modified to take into account discrete parameters. A SIMPLEX can be described as a geometric figure with $N+1$ vertices in a N-dimensional design space. Finding of the optimum of the objective function is achieved using heuristic operators such as reflection, expansion, or contraction. This also means that this method does not use gradients and is therefore very robust even in discontinuous design spaces.

Genetic Algorithms

Genetic algorithms (GA) are the best known members of the global stochastic algorithms family. They are typically robust but require a significantly larger number of objective function evaluations than gradient-based schemes. They can also easily be parallelized, which is of course of great interest for CFD optimization on supercomputers. The operators involved in a GA are selection, crossover, and mutation.

Typical optimization problems are usually solved by means of "hill-climbing procedure" possibly based on local gradients of a stated cost function. The typical drawback of this approach is the fact that the search for improvements is done efficiently but is done locally. On the other hand, probabilistic optimization techniques can be used to examine a large but discrete configuration space in order to find a good solution as close as possible to the global optimum. Some of the possible operators are briefly illustrated hereafter pointing out the most suitable to engineering problems, like aerodynamic optimization, for which the objective functions topology is not too complex (nondeceptive) but the calculation of the objectives is computationally intensive.

A genetic algorithm works on a population of designs and performs selection, crossover, and mutation on it. The key points of the GA are therefore the operators used for selection and reproduction that highly influence the robustness and the efficiency of the algorithm.

Several selection schemes are available in modeFRONTIER, such as roulette wheel, tournament, or local geographic selection.[21]

Crossover operators are described next.

Two-points crossover. is the most classical operator for reproduction and is one of the operators that offer the highest robustness to the search. Two point are randomly chosen, and the genetic materials (the design variables) are exchanged between the parent variables vectors.

Directional crossover. is slightly different and assumes that a "direction of improvement" can be detected comparing the fitness values of two reference individuals. A novel operator called evolutionary direction crossover was introduced,[22] and it was shown that even in the case of a complex multimodal function this operator outperforms classical crossover. The direction of evolutionary improvement is evaluated by comparing the fitness f in generation j of the individual i with the fitness of the two parents belonging to generation $j-1$. The new individual is then created moving in a randomly weighted direction that lies within the ones individuated by the given individual and his parents. A similar concept can be however applied on the basis of directions not necessarily linked to the evolution but detected selecting two other individuals in the generation. The new scheme for the directional crossover becomes: $\forall \boldsymbol{x}_i$, select individual $\boldsymbol{x}_1$, select individual $\boldsymbol{x}_2$ and create the offspring $\boldsymbol{x}$ given by

$$\boldsymbol{x} = \boldsymbol{x}_i + S \cdot \text{sign}[f(\boldsymbol{x}_i) - f(\boldsymbol{x}_1)] \cdot (\boldsymbol{x}_i - \boldsymbol{x}_1) + T \cdot \text{sign}[f(\boldsymbol{x}_i) - f(\boldsymbol{x}_2)] \cdot (\boldsymbol{x}_i - \boldsymbol{x}_2)$$

where S and T are random numbers in the interval [0; 1] and f is the value of the fitness function for the corresponding vector of variables $\boldsymbol{x}$.

Genetic Algorithm for Multicriteria Design Optimization

Given a design $\boldsymbol{x}$ described by its geometrical variables (discrete or continuous), the multiobjective optimization problem consists in

maximizing (or minimizing) an objective function f under one or several constraints g_j, $j = 1 \ldots m$.

$$\forall i \in [1; n], \qquad \max f_i(\boldsymbol{x}) \tag{16}$$

$$\forall j \in [1; m], \qquad g_j(\boldsymbol{x}) \leq 0 \tag{17}$$

and it is obvious that in general the solution is not unique if the functions are not linearly dependent. The pareto dominance relation makes it possible to divide any group of solutions into two subgroups: the dominated and the nondominated ones. For a maximization process of all objectives, design $\boldsymbol{x}_1$ dominates design $\boldsymbol{x}_2$ if:

$$\forall i \in [1; n], \qquad f_i(\boldsymbol{x}_1) \geq f_i(\boldsymbol{x}_2) \tag{18}$$

$$\exists j \in [1; n], \qquad f_j(\boldsymbol{x}_1) > f_j(\boldsymbol{x}_2) \tag{19}$$

The set of nondominated design is denoted the pareto front (or pareto set). It represents the best designs for the MDO problem because for designs in the pareto front an objective function can be improved only at the cost of a loss in performance of one or more other objective functions.

Classical optimization algorithms are capable, under strict continuity and derivability hypothesis, of finding the optimal value only in the single objective case, and therefore the problem of finding the group of nondominated solutions (the pareto set) is reduced to several single objective optimizations where the objective becomes a weighted combination of the objectives called the utility function Obj:

$$Obj = \sum_{i=1}^{n} w_i f_i(\boldsymbol{x})$$

where w_i, $i \in [1; n]$ are the weights for the objectives f_i. A more sophisticated and effective way to transform a multiobjective problem into a single-objective problem is the use of an utility function that is not a simple weighted sum of objectives but that is a nonlinear combination of the objectives, that is, the weights are not constant but are given as a monotone function of the objective value.

Although traditional optimization algorithms do need the use of an utility function, the particular structure of genetic algorithms can face the multiobjective optimization problem in a more direct way, developing populations in which the diversities follow the conflicting objectives. Pareto-GA algorithms mainly differ from classical GA in the selection process even though other specific operators might be constructed. A quick list of pareto-GA techniques is given here: Pareto tournament selection,[23,24] ranking-based selection, local Pareto selection, ranking-based selection (nondominated sorting GA),[25,26] local geographic selection based on Pareto dominance, local tournament directional selection, local geographic directional selection, sharing function,[25] vector-evaluated genetic algorithm,[27] and multidirectional crossover.

Constraints Handling Methods

During the past few years, several methods have been proposed for handling constraints by genetic algorithms. These methods can be grouped into two main categories: methods that preserve the feasibility of solutions and methods based on penalty functions.

The methods that preserve the feasibility are quite effective but are not always applicable. In fact, the feasibility of the offsprings can be maintained only by repairing techniques or by specialized crossover operators that are usually problem dependent.

The methods based on penalty functions are based on the concept that the fitness function is decreased according to the intensity of the constraint violation. A drawback of this methodology is that the intensity of the penalization can be problem dependent and might draw the optimization to premature convergence. A way to overcome this problem is to adapt the penalty function to the current population under evaluation.

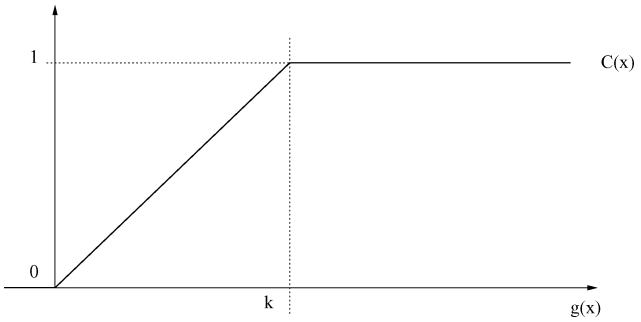


Fig. 8 Constraint handling method.

Two possible methods suitable also for multiobjective optimization are described later. The constraint is transformed into a fuzzy function C as shown in Fig. 8.

For all of the objectives O_i, the fitness function is computed as follows:

$$f_i(\boldsymbol{x}) = O_i(\boldsymbol{x}) - \sum_j C_j(\boldsymbol{x}) * O_{\max}$$

where 0 is the "soft" constraint limit and k is the "hard" constraint limit. If $g(x)$ is negative, the individual is feasible, and the optimizer works as in the case of unconstrained optimization; if $g(x)$ is higher than k, the individual is worse than any feasible solution, and the number of violated constraints dominates the problem. For $0 < g(x) < k$ the distinction between feasible and unfeasible individuals is fuzzy.

When the problem to be solved is dominated by the constraints and the optimization task is in fact to find a feasible solution that might even not exist, the number of constraints violated can become itself a new objective to be minimized $O_{n+1} = O_i(x) - \Sigma_j C_j(x)$.

The MOGA used in modeFRONTIER could use all of the just-mentioned methods, but in order to set up the algorithm in a simple way only most effective operators are directly given as options to the user. In most cases local geographical selection associated with directional crossover is chosen, whereas tournament selection based on pareto ranking is associated with classical two point crossover. Recommended tuning for the MOGA algorithm is to use populations of N with $N \simeq 2 \cdot N_{\text{var}} \cdot N_{\text{obj}}$, where N_{var} is the number of variables and N_{obj} the number of objectives.

Evolutionary Strategies

Evolutionary strategies used in modeFRONTIER are variants of the methods originated by Schwefel[28] and Rechenberg[29] in the early 1960s. The main operator in such algorithm is the mutation, which creates random offsprings from a population of different search points. Refined mutation operators using, for example, self-adaptation are necessary to obtain powerful and robust algorithms.

Two variants of evolutionary strategies are available in modeFRONTIER. The first one is a derandomized evolution strategy (DES) using fast direct adaptation of the multivariate stochastic distribution. The recommended offspring-population size is given by $\lambda \simeq E(3 + 3\,\ell n\, N_{\text{var}})$. The second one is a population-based evolution strategy called multimembered multicriterial evolution strategy (MMES), which allows to perform multicriteria optimizations. The main difference with DES is that MMES works with a population of search points, allowing the performing of recombination and maintaining of diversity. These algorithms are called (μ, κ, λ)-ES, where μ is the number of parents, λ the number of offsprings, and κ the number of iterations an individual can survive in the parent population. A good setup of this algorithm for discontinuous design spaces is to use a value of $\kappa \simeq 5$ and maintain the restriction $\mu \cdot \kappa < \lambda$.

Modelization of Inlet Performance

The inlet performance is mostly assessed by its so-called performance curve, that is, the graph of total pressure recovery coefficient η_{02} vs mass flow rate ϵ. These two important coefficients are

defined by

$$\epsilon = m_2/\rho_0 V_0 A_1 \tag{20}$$

$$\eta_{02} = Pi_2/Pi_0 \tag{21}$$

During the optimization process, the inlet performance is computed with an hybrid solver named 2ES2D (Euler semi-empirical solver for two-dimensional inlets) developed by MBDA France and Rutgers University.[30] This solver computes the mass flow rate and the critical total pressure recovery of a supersonic inlet with a 5% accuracy and requires low CPU times (about 1 min CPU on a Pentium III 500-MHz PC) so that it is an adequate tool for an automated optimization loop. The software solves each part of the inlet flow with the following modelization. First, an Euler code computes the flow from the inlet entrance (using as inflow condition the flowfield generated by the three-dimensional RANS simulation of the missile forebody) to the inlet throat. Then a virtual terminal shock is applied at the geometric throat by simply solving the normal shock equations in each cell. Finally, the total pressure loss in the subsonic diffuser is obtained through a one-dimensionnal integral method.[31] This modelization predicts the supercritical part of the performance curve: this means that for our ramjet optimization, the last operating point of the inlet is the critical regime, which generally gives a safe margin from buzz.

The inlet geometric model for the optimization loop is a mixed-compression two-ramps inlet shown in Fig. 9. The cowl is divided in two segments. The degrees of freedom for the optimization process are therefore the angles and lengths of the ramp and cowl segments and the relative axial position of the cowl apex from the ramp apex and the capture area. The minimum and maximum values for each parameter are given in Table 4.

Preconstraints

Some preconstraints taken from physical considerations are used during the optimization process in order to limit the actual number of computations. They are based on known fundamental results on supersonic inlet designs. For example, the angles of compression ramps have to increase, whereas those of the cowl have to decrease. Also, knowing that mixed compression inlets offer more performance, the cowl apex has to be located rearward of the first ramp end and forward of the second ramp end.

Table 4 Parameters used for the optimization process

Parameter	Min.	Max.
L_i, mm	10	200
θ_i, deg	0	30
L_{ci}, mm	10	200
θc_i, deg	0	30
Width, mm	50	150
X_{cowl}	50	150
Y_{cowl}	50	150

Feasibility of a Design

One of the most important characteristic of the optimization problems involved in the present study is that the major part of the design space contains infeasible designs. A design is feasible if all of the following constraints are satisfied: the geometry must respect the preconstraints, all the 2ES2D CFD computations must converge, and all of the missile balance equations must be satisfied. Such a case is really a good testing for optimization algorithms robustness.

A design of experiments containing 500 individuals was generated with the Sobol algorithm,[32] which creates good uniform distributions. Among these designs, four were feasible for the multiobjective problem, which means that less than 1% of the design space is feasible.

Performance of the Initial Design

The performance of the baseline design is first assessed. This design was built using semi-empirical rules with one first objective: being able to sustain cruise conditions at Mach 2.8 and an altitude of 15,000 m. The results of cruise performance computations are presented in Table 5 for three different possible mass of the missile (450, 500, and 550 kg). In each case the missile is able to sustain cruise. The heavier the missile is, the higher the flight angle of attack and therefore the more thrust needed, which leads to an increase in fuel consumption. One should also notice that the performance margin is not important: η_{max} represents the maximum total pressure recovery possible for the considered inlet, whereas η is the actual total pressure recovery needed to provide the desired thrust. The difference between these two values is small in each case, from 0.03 to 0.011, showing that the missile is not far from its ceiling altitude. To give a simple idea of ranges provided by these level of fuel consumption, let us say that 30 kg of kerosene is available for cruise. This leads to respective range of about 195, 185, and 175 km

Table 5 Performance of the initial missile

Mass	450 kg	500 kg	550 kg
Cruise			
α_{eq}, deg	6.16	6.67	7.17
C_Π	0.566	0.597	0.631
η_{max}	0.561	0.566	0.571
φ_i	0.563	0.584	0.608
η	0.531	0.545	0.560
m_k, kg/s	0.1265	0.1332	0.1404
Acceleration			
α_{eq}, deg	2.053	2.277	2.501
C_Π	0.363	0.361	0.359
η	0.846	0.845	0.844
φ_i	0.413	0.410	0.407
N_x	−0.237	−0.229	−0.223

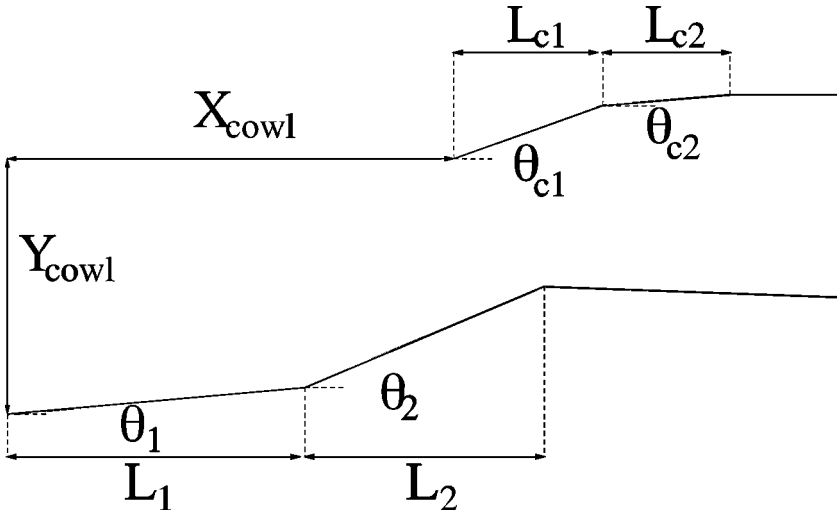


Fig. 9 Two-dimensional inlet geometry parameterization.

for each missile mass, these ranges representing only the distance traveled during the cruise phase.

Thus performance of the initial design seems correct for the cruise condition, but the mission is also composed of a ramjet-accelerated phase and some maneuvers at $M = 2.2$ and low altitude. The acceleration coefficients N_x at Mach 1.9 $Z = 100$ m are presented for the three considered masses in Table 5. In this case the performance does not satisfy the mission because the missile decelerates. The ramjet thrust is limited both by the inlet mass flow and its total pressure recovery, leading to low values of injected equivalent ratio φ_i. As shown by Fig. 10, this could easily be explained by an inlet in under-design conditions with a bow shock wave leading to low values of ϵ around 0.78. Another performance limitation comes from the additive drag caused by the inlet. Inlet optimization is therefore necessary for this flight condition in order to satisfy the mission.

The last performance considered is maneuver at Mach 2.2 $Z =$ 100 m. In this case the inlet also limits the performance of the missile so that it is impossible to find an angle of attack for which the ramjet thrust balances the missile drag. This means that the missile will be able to perform some maneuvers only by decelerating, which of course limits the duration of these maneuvers.

A performance survey of the preliminary design of our missile test case shows that a first objective is reached because the missile is able to sustain cruise. However, it is neither able to accelerate from low supersonic speeds nor perform countering defense maneuvers without decelerating. The main purpose of our study is to demonstrate that an important performance gain is possible only by improving the inlet shape with an automated optimization loop directly considering the global missile performance.

Capture-Area Optimizations

A first way to improve the global performance of an airbreathing missile consists of optimizing the value of the capture area, which is one of the most important parameter because it directly influences both the ramjet thrust and the missile drag. Therefore several optimizations were conducted on the basic inlet shape with only the capture area A_1 as a variable parameter. No real automated optimization was conducted because only simple function analysis of $N_x(A_1)$, $N_z(A_1)$, and $m_k(A_1)$ were sufficient to find the optima.

The inlet considered here is the preliminary design with its computed performance ϵ and η. The internal drag and lift were computed for an initial value of $A_1/S_R = 0.4$, and their variation with the capture area are taken into account in the optimization process.

Monoobjective Analysis

First, several monoobjective optimizations were conducted for each flight condition in order to measure the maximum values of performance that can be reached with the preliminary design of the inlet. The evolution of fuel consumption as a function of A_1/S_R is plotted in Fig. 11. It is important to notice that for each mass the fuel consumption is minimum for the biggest capture area that permits to sustain cruise. The optimum designs are shown in Table 6. These optimizations lead to a minor improvement in fuel consumption, which is normal because the cruise condition was the main objective of the preliminary design. The initial setting of A_1 to a value of 0.4 based on empirical considerations was quite good, and more improvement of the fuel consumption can be reached only by changing the inlet shape.

On the other hand, the performance of the preliminary design was poor for the acceleration flight condition. The evolution of the acceleration factor as a function of A_1/S_R is plotted in Fig. 12. Only a small part of the design space permits the reaching of positive

Table 6 Capture-area optimizations

Mass	450 kg	500 kg	550 kg
Cruise			
A_1/S_R	0.442	0.429	0.416
m_k, kg/s	0.1215	0.1297	0.1386
Acceleration			
A_1/S_R	0.292	0.291	0.290
N_x, g	0.258	0.222	0.192
Maneuver			
A_1/S_R	0.316	0.316	0.316
N_z, g	5.483	6.032	6.702

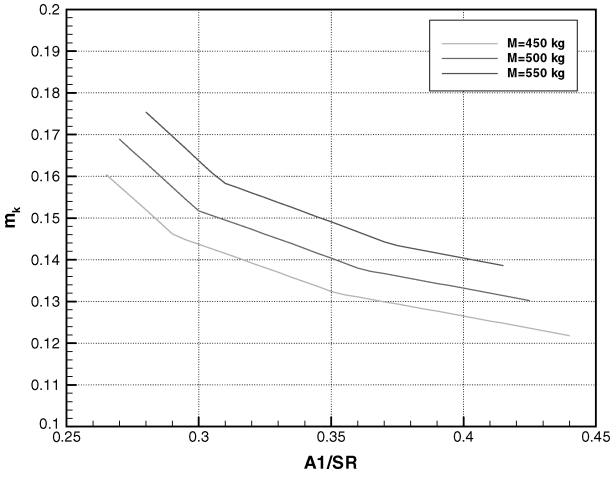


Fig. 11 Fuel consumption as a function of A_1/S_R.

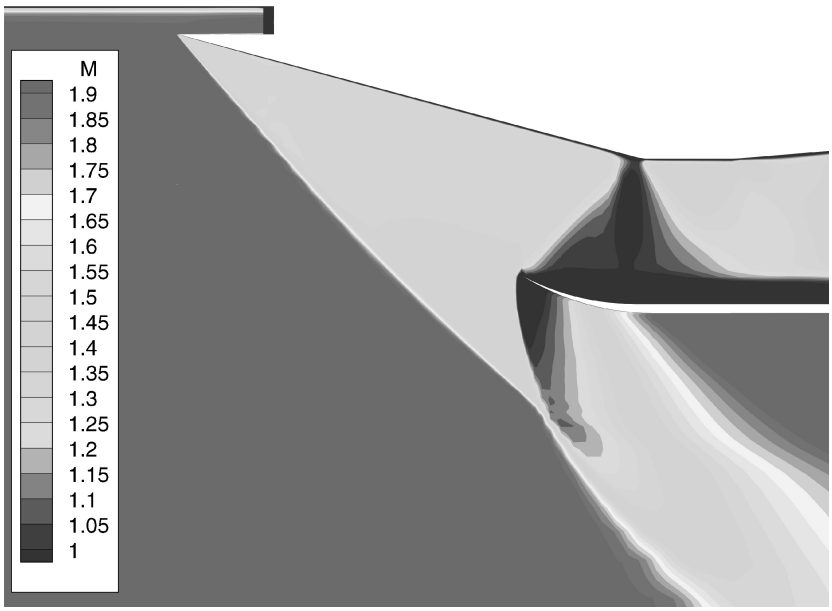


Fig. 10 Mach-number field for the preliminary inlet design at $M = 1.9$, $\alpha = 3$ deg.

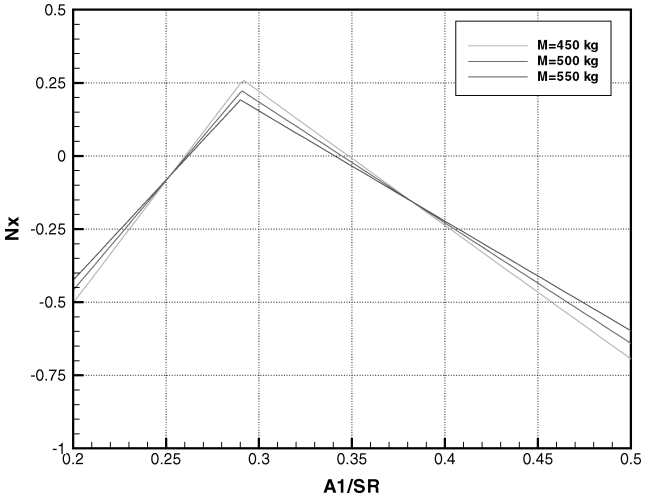


Fig. 12 Acceleration as a function of A_1/S_R.

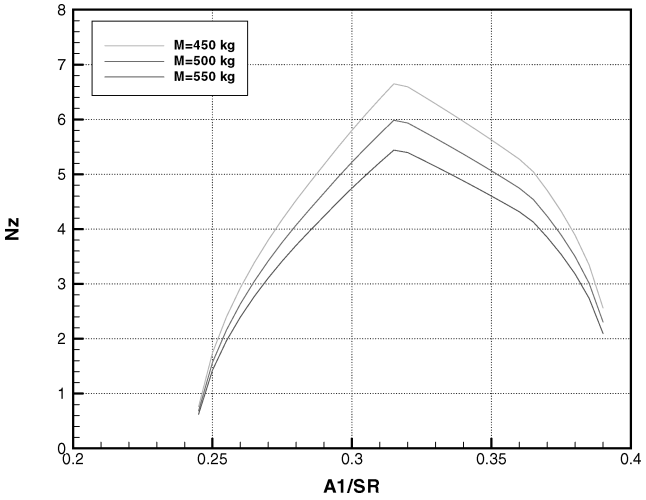


Fig. 13 Maneuver as a function of A_1/S_R.

acceleration. The results of the capture-area optimization for this case are presented in Table 6. Values of about 0.29 conduct to low positive acceleration factors.

Finally, monoobjective optimizations were performed for the maneuver flight condition. The evolution of the maneuver factor as a function of A_1/S_R is plotted in Fig. 13. The results are presented in Table 6, showing that an optimum value of $A_1 = 0.316$ leads to acceleration factors N_z from 5.48 to 6.70 g. Whereas the optimal value of A_1 is independent of the mass because it only comes from the balance of drag and thrust, the acceleration factor of course depends on the missile mass.

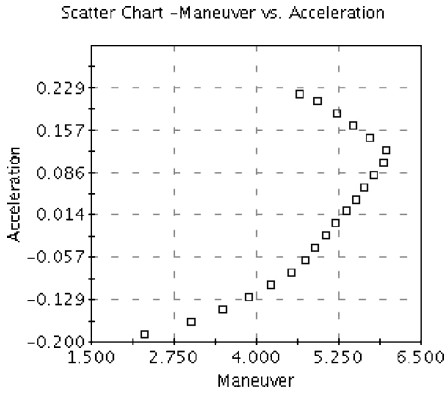
Multiobjective Optimization

The missile performance for its entire mission can be globally optimized only by performing multiobjective optimizations. All three objectives only depend on one variable so that no optimization algorithm needs to be used. The study has been done computing the missile performance for values of A_1 from 0.2 to 0.5 with a step of 0.05. The designs are feasible for $A_1 \in [0.27; 0.39]$. However, even if no algorithm is used the point of interest for a designer is the presentation of the results emphasizing the pareto front.

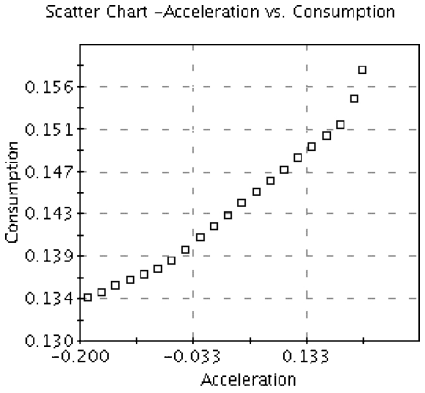
The pareto front obtained for the case $M = 500$ kg is shown in Fig. 14, where each graph is a projection on a two-objectives plane. Several remarkable designs can be extracted from the pareto front and are presented in Table 7. First, the three best individuals for each performance are of course interesting. It is evident that these designs do not necessarily correspond to those found in the monoobjective optimizations because a design must be feasible for all three objectives to belong to the pareto front. The design having

Table 7 Remarkable designs among the pareto front

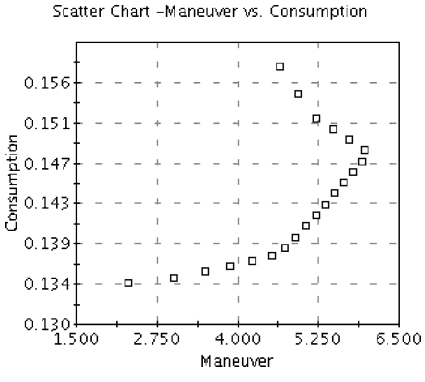
Comments	A_1/S_R	N_x, g	N_z, g	m_k, kg/s
Best m_k	0.390	−0.188	2.305	0.1343
$N_x \simeq 0$	0.340	0.019	5.365	0.1427
Best N_z	0.315	0.122	5.984	0.1484
Best N_x	0.290	0.217	4.659	0.1574



a) N_x vs N_z



b) m_k vs N_x



c) m_k vs N_z

Fig. 14 Pareto front for capture-area optimization.

the best fuel consumption leads also to a deceleration at $M = 1.9$ and to a low maneuver factor at $M = 2.2$. From the best possible consumption reaching the best maneuver costs 10.5% in fuel consumption, and reaching the best acceleration costs 17%. Going from the best maneuver design to the best acceleration, one increases N_x of 78%, whereas N_z decreases by 22% and m_k increases by 6%. Another interesting design is the first design that leads to a 0-g acceleration factor because it gives a reference minimum value for fuel consumption, even if it is not sufficient to satisfy the mission. Such a presentation relying on a pareto front emphasizes the tradeoff between each performance and is a rich source of data for project managers.

Isolated Inlet Performance Optimization

Three separate inlet optimization were performed at $M = 1.9$, $\alpha = 3$ deg; $M = 2.2$, $\alpha = 8$ deg; and $M = 2.8$, $\alpha = 6$ deg with the objective to find the best total pressure recovery. In each case, as shown by Figs. 15–17, the optimization loop found classical on-design inlets, that is, with shock waves from the ramps focusing on the lips. The values of total pressure recovery reached are 0.907 at $M = 1.9$, 0.853 at $M = 2.2$, and 0.630 at $M = 2.8$, which constitute very good performance. This validates the automated optimization process because it leads to a well-known results on supersonic inlet aerodynamics. It is emphasized that optimization of the global performance of the aeropropulsive system does not lead to the same inlet design as obtained by optimization of the inlet aerodynamic performance alone.

Inlet Mono-Objective Optimizations

Three single optimizations were conducted on the $M = 500$ kg case in order to find the best possible inlet design for each flight condition.

Maneuver

A comparison of algorithms was performed for the maneuver optimization, with results summarized in Table 8. All of the algorithms were used with their default or recommended parameter in

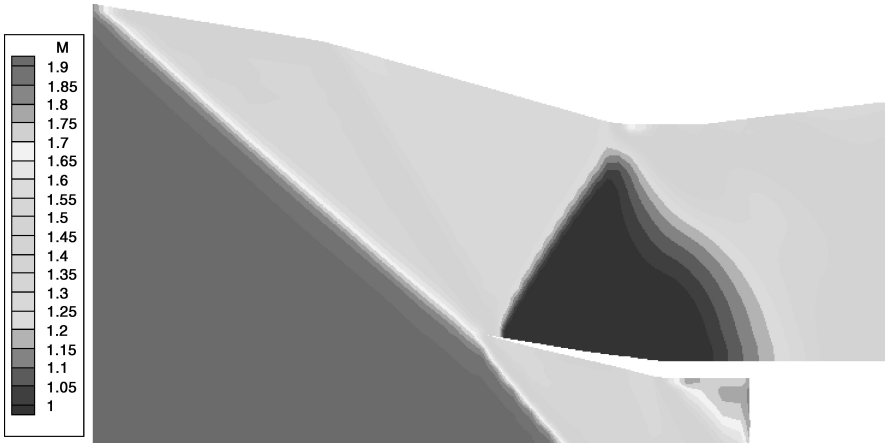


Fig. 15 Mach-number field for best design at $M = 1.9$, $\alpha = 3$ deg.

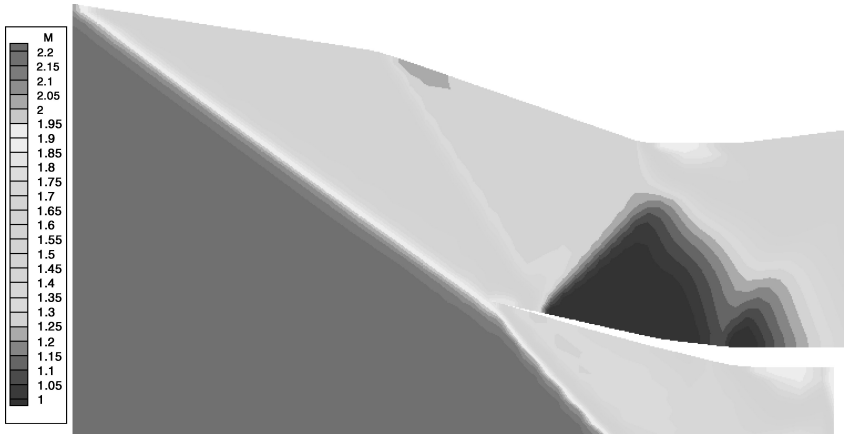


Fig. 16 Mach-number field for best design at $M = 2.2$, $\alpha = 8$ deg.

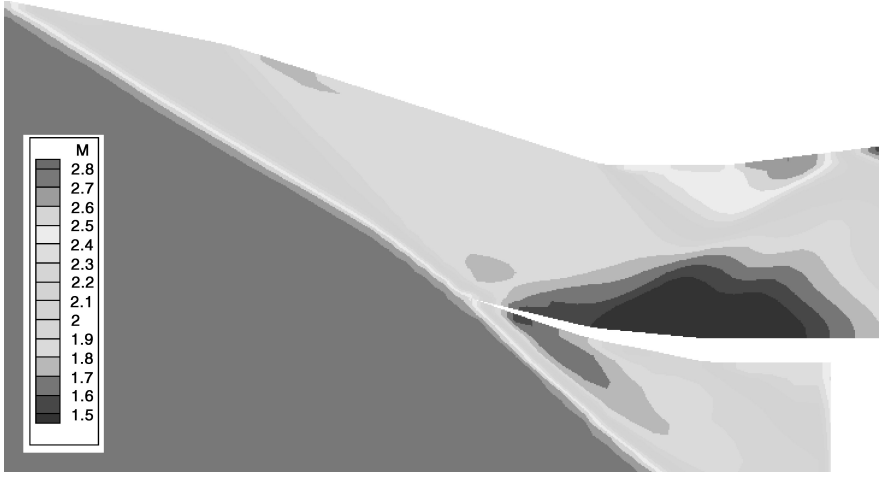


Fig. 17 Mach-number field for best design at $M = 2.8$, $\alpha = 6$ deg.

modeFRONTIER. To clearly assess the quality of each algorithm, it should have been necessary to optimize the algorithm tuning, but this kind of study is generally not possible in a concurrent industrial context, which explains our choice of comparing algorithm with their default parameters.

The first optimization was conducted with the MOGA genetic algorithm. Because there are 11 variables, the initial population size recommended value is 22. After 30 generations leading to 583 different designs[d], MOGA finds 260 feasible designs (about 40% of the evaluated designs) and leads to an optimum of $N_z = 6.735\ g$, which represents an improvement of 11.6% compared with the A_1-only optimization. (MOGA might find the same design several times during an optimization process, which explains that for 660 iterations, 583 actual designs were evaluated.) The optimization history is presented in Fig. 18. It is evident that MOGA, even after more than 20 generations, keeps on finding designs having poor performance, which indicates a high robustness in the sense that diversity is maintained. On the other hand, this algorithm might not be able to properly identify a real optimum, but only finds a region of the design space narrowing this optimum. This is might be caused by a too-high default rate of mutation.

Table 8 Comparison of algorithms for N_z optimization

Algorithm	Best N_z	Nb. Des.	Feasible Des.
MOGA	6.735	583	260
FMOGA	6.201	2200	72
DES	6.866	277	248
SIMPLEX	7.171	209	199

Another optimization was tried with the FMOGA algorithm, which internally incorporates response surface. But even after more than 100 generations, this algorithm did not find more than 72 feasible designs, which shows that such a method can be dangerous for design spaces containing lots of unfeasible design. The final result of $N_z = 6.201\ g$ is therefore disappointing.

An optimization was also performed with the DES evolutionary strategy. The offspring population size was taken at a value of $\lambda = 10$, and 30 generations were formed. The optimization history is plotted in Fig. 19 and shows that DES has more of a tendency to actually converge toward an optimal design than MOGA. It is also remarkable that 89% of the designs found were feasible. The best design is indeed quite better than MOGA's one with a value of $N_z = 6.866\ g$ leading to an improvement of 13.8% compared with A_1 optimization result.

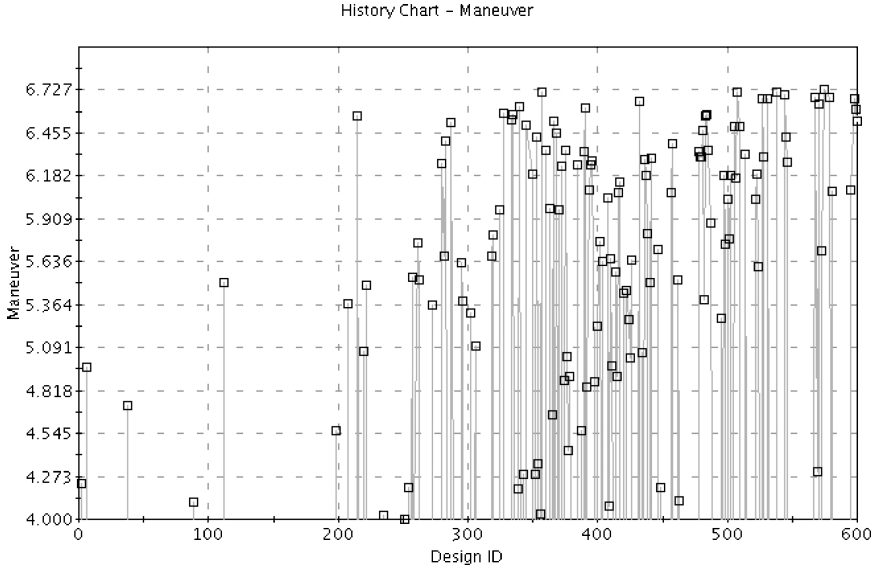


Fig. 18 History for N_z optimization with MOGA.

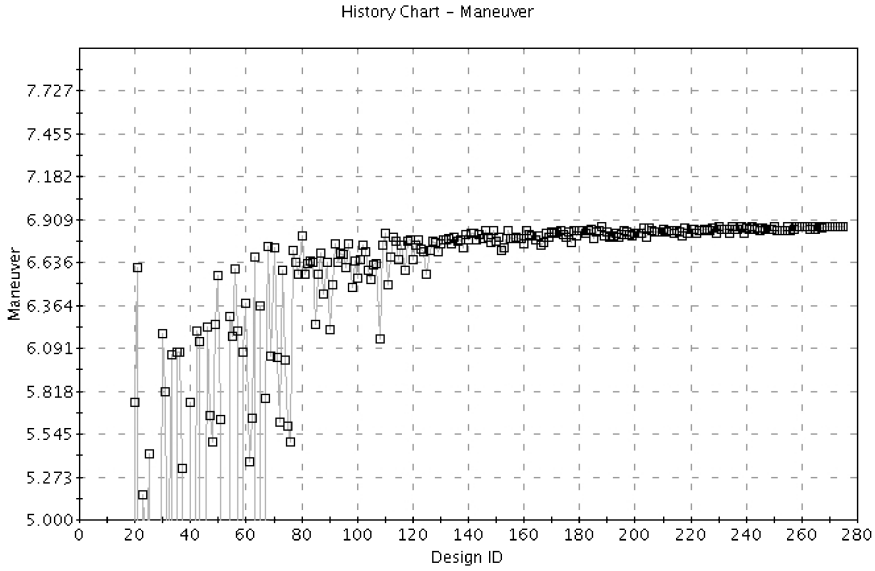


Fig. 19 History for N_z optimization with DES.

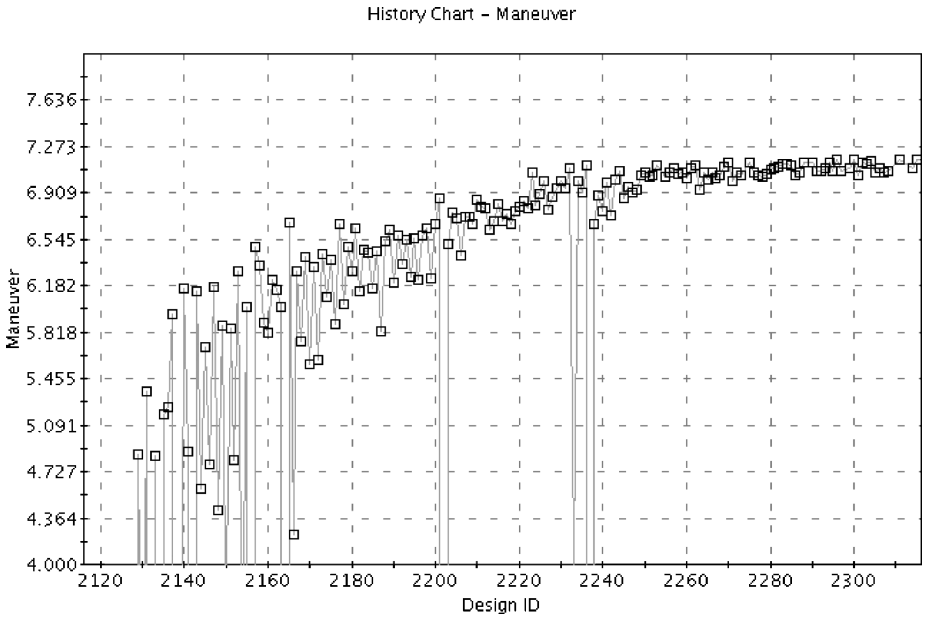


Fig. 20 History for N_z optimization with SIMPLEX.

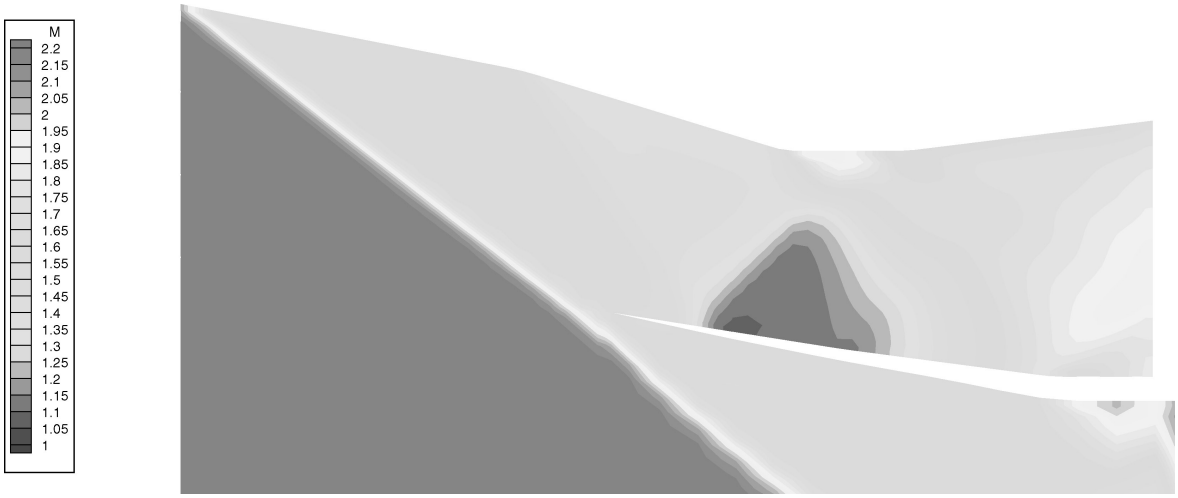


Fig. 21 Mach-number field for best N_z design at $M = 2.2$, $\alpha = 8$ deg.

To find if a final improvement is possible, the 12 best designs found by FMOGA were used as initial design for a SIMPLEX optimization. Such an operation is easily made with an integrated tool-like modeFRONTIER. This algorithm performs 209 iterations until a relative convergence criteria of 10^{-4} is satisfied (see Fig. 20). Ninety-five percent of the evaluated designs are feasible, and the final result of $N_z = 7.171$ g leads to a final improvement of about 18.9% compared with the A_1 optimization. For this best design the maneuverability factor is obtained at an angle of attack of 8.55 deg. The thrust coefficient has a value of $C_\Pi = 0.708$ and is obtained for a burnt equivalent ratio $\varphi_b = 1$, which means that the maximum power of the ramjet is reached. At 8 deg angle of attack the inlet design has a mass flow coefficient of $\epsilon = 1.059$ and a total pressure recovery of $\eta = 0.842$, which correspond to improvement of 15 and 7.8% compared with initial design. The Mach-number field for this flight condition is shown in Fig. 21.

Acceleration

The computation history with the DES algorithm is shown in Fig. 22. Forty generations were formed with an offspring-population size $\lambda = 10$. The best design found has an acceleration factor $N_x = 0.437$ g, which represents a 96.8% improvement compared with the capture-area optimization. The inlet shape leads to respective mass-flow-rate coefficient $\epsilon = 0.93$ and $\epsilon = 0.94$ at angle of attack of 1 and 3 deg and total-pressure-recovery values of $\eta = 0.909$ and 0.911. The Mach-number field for the flight condition $M = 1.9$, $\alpha = 3$ deg is presented in Fig. 23, which compared with the initial design in Fig. 10 shows that the highly subsonic region under the cowl is highly reduced. Finally, the acceleration factor $N_x = 0.437$ is obtained at an angle of attack of 2.75 deg with a thrust coefficient $C_\Pi = 0.600$ and an injected equivalent ratio $\varphi_i = 1.1$ showing that the ramjet is used at full thrust.

An optimization was conducted with the MOGA algorithm for 20 generations in order to compare both algorithms with a similar nuber of evaluations. (MOGA's population is 22, whereas 40 generations were asked with DES and its population of 10.) The evaluation history is plotted in Fig. 24. Again, MOGA seems to widely explore the design space but not to converge to a global optimum. The default mutation rate is perhaps too high, or the number of genrations too low to reach convergence. The best design found has an acceleration factor of only $N_x = 0.346$, which is 20.8% lower than the one found by DES for a comparable number of evaluations. Then, a SIMPLEX optimization with the 12 best designs found by MOGA as starting points were performed: 72 designs were evaluated, all were feasible, but the improvement is poor with a best result of only $N_x = 0.352$. The SIMPLEX algorithm seems here to be trapped by a local optimum.

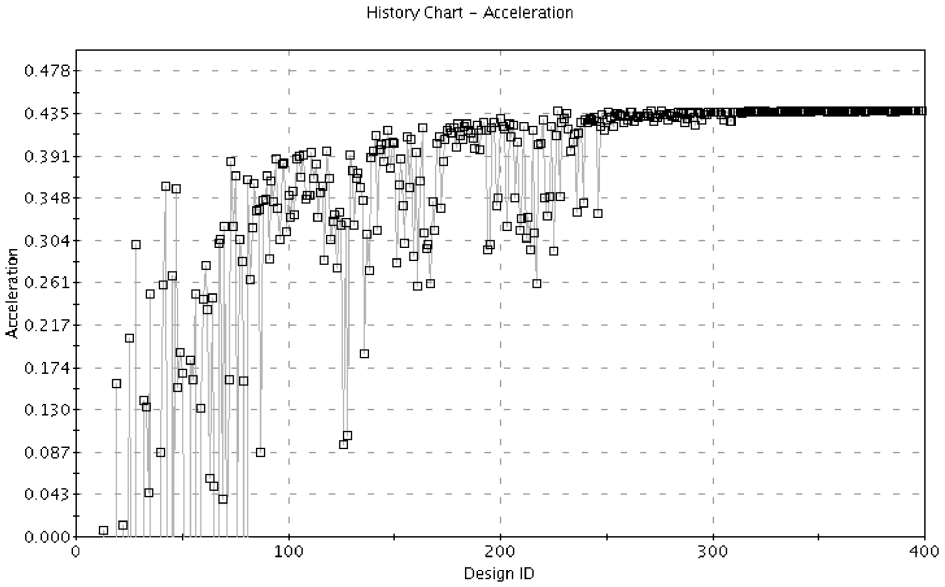


Fig. 22 History for N_x optimization with DES.

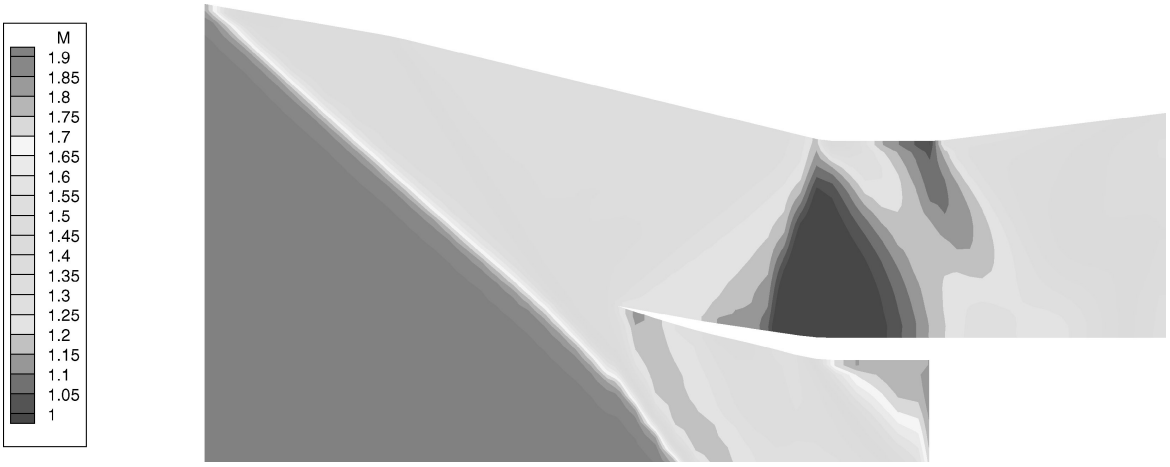


Fig. 23 Mach-number field for best N_x design at $M = 1.9$, $\alpha = 3$ deg.

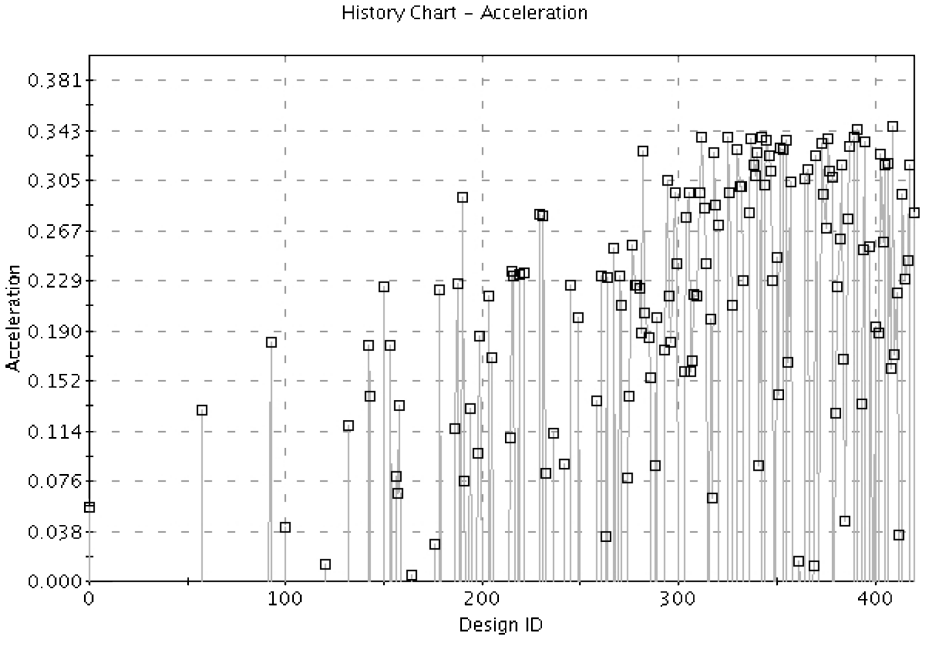


Fig. 24 History for N_x optimization with MOGA.

Cruise

The cruise consumption minimization was performed with the DES algorithm. The optimization history is shown in Fig. 25 for 40 generations computation with an offspring-population size $\lambda = 10$. The best value found is $m_k = 0.1215$ kg/s, which corresponds to a 6.3% improvement compared with the capture-area optimization and of 8.8% compared with the initial design performance. Cruise balance is obtained at an angle of attack $\alpha = 6.63$ deg with a thrust coefficient $C_\Pi = 0.591$ and an injected equivalent ratio $\varphi_i = 0.440$. The inlet has a mass flow rate of the same order as the initial design, which is logical because the regime is above-design Mach number and a slightly better pressure recovery of 0.597 and 0.619 for angles of attack of 6 and 8 deg.

The Mach-number field for the flight condition $M = 2.8$, $\alpha = 6$ deg is presented in Fig. 26. The inlet is on-design clearly confirming that the optimizer reached undoubtebly the best possible design for the modelization it uses.

Survey of Mono-Objective Results

For our mono-objective optimizations the best algorithm provided by modeFRONTIER is the DES evolutionary stategy, which achieves the optimal performance with a small number of evaluations. MOGA with its default parameters seems to widely explore the design space but demands more iterations to converge and actually reaches lower values of performance. A quite efficient strategy as shown by the maneuverability factor optimization is to perform a SIMPLEX optimization with the best designs found by MOGA, but this method also failed in the acceleration case. A final remark has to be made: for each of the optimum design obtained, a final optimization was performed with the Broyden–Fanno–Fletcher–Goldfarb–Shanno gradient algorithm, and no more improvement was found, which shows that an optimum (local or global) was actually reached by the global methods.

The best designs found for each objective are described in Table 9. Their performance for the two other criteria are also given. The design having the best acceleration factor is not able to sustain cruise but has a quite good maneuver factor. Similarly, the design leading to the best maneuver has a limited acceleration factor but induces a 25.4% increase compared with the best possible consumption. Finally, the minimal fuel consumption design has a very poor maneuverability capacity and strongly decelerates at $M = 1.9$.

Table 9 Geometrical description of best designs for mono-objective optimizations

Geometry	Best N_x	Best N_z	Best m_k
L1	83	139	146
θ_1	9.8 deg	11 deg	13.8 deg
L2	189	107	134
θ_2	13.6 deg	17 deg	17.3 deg
LC1	39	96	60
θc_1	10.3 deg	9 deg	12.9 deg
LC2	49	86	81
θc_2	8.3 deg	7.5 deg	9.5 deg
X_{cowl}	180	173	219
Y_{cowl}	132	125	158
A_1/S_R	0.261	0.293	0.483
N_x, g	0.437	0.134	−0.534
N_z, g	5.709	7.171	0.845
m_k, kg/s	X	0.1524	0.1215

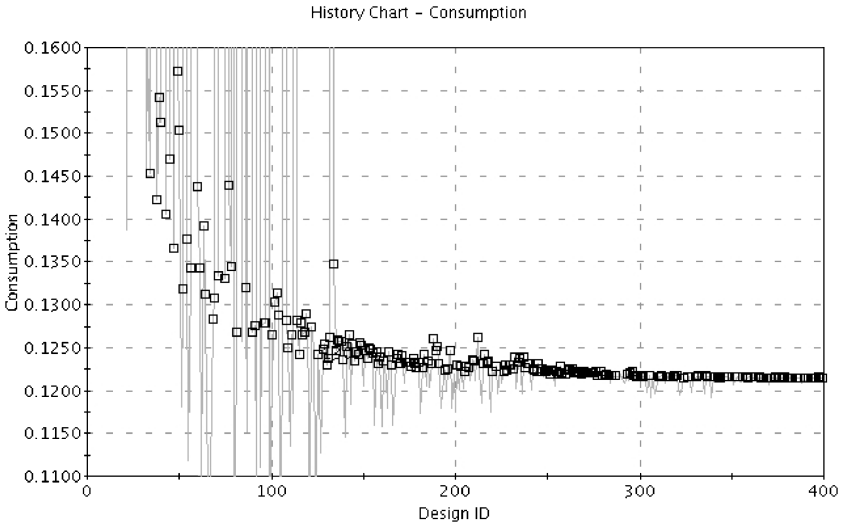


Fig. 25 History for m_k optimization with DES.

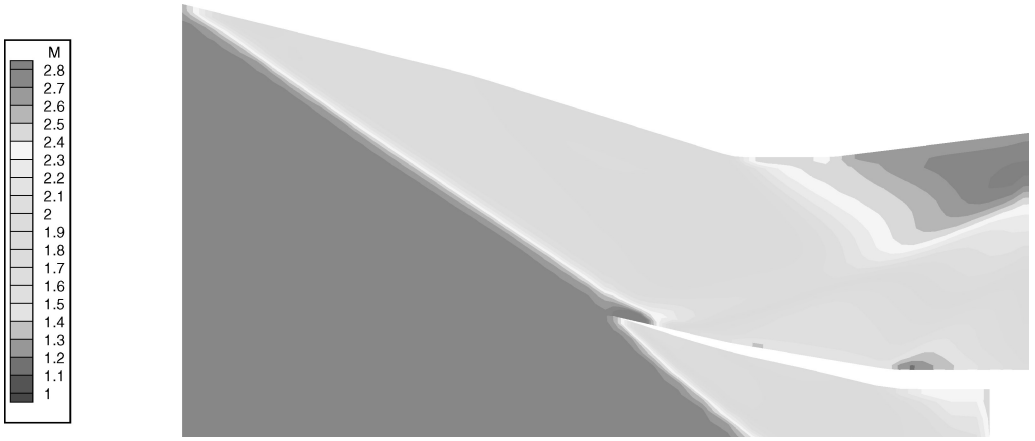


Fig. 26 Mach-number field for best m_k design at $M = 2.8$, $\alpha = 6$ deg.

The three mono-objective optimizations lead to designs showing the best possible individual performance, but no shape actually satisfies the global mission.

Inlet Multiobjective Optimizations

Computation Details

Each design evaluation requires seven CFD computations and three missile balance computations, which costs about 15 min CPU on a Pentium III 500-Mhz PC. The CPU time caused by the search algorithm is less than 1 s per iteration. The entire optimization proces takes about two weeks. Each of these operations has to be monitored in term of convergence. Such an automated optimization loop needs each part of the evaluation tools to be robust and reliable.

One multiobjective inlet optimization of N_x, N_z, and m_k was performed with the MOGA algorithm. Fifty generations were formed on a population size of 70 designs. A total of 3271 designs were actually evaluated, and 1323 of them were found feasible.

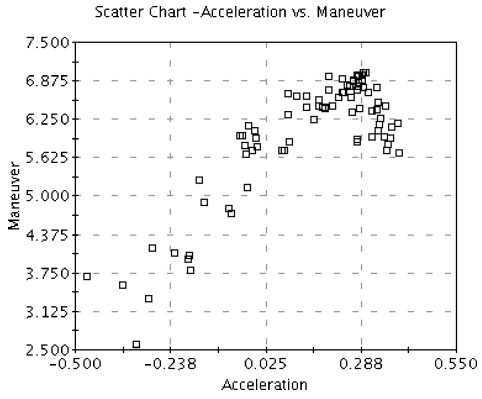
Another computation was performed with the MMES (μ, κ, λ) evolutionary strategy, which is the multiobjective version of the DES algorithm. The following settings were used: the number of parents is $\mu = 10$, the number of offspring is $\lambda = 70$, and the maximum life span is $\kappa = 5$. Recombination is set to off because this case is multiobjective. These values are typical settings recommended for such problems. Also, 50 generations were asked in order to be coherent with the computation performed by MOGA; 2537 designs were actually evaluated, and 1519 of them were found feasible.

MOGA Pareto-Front Analysis

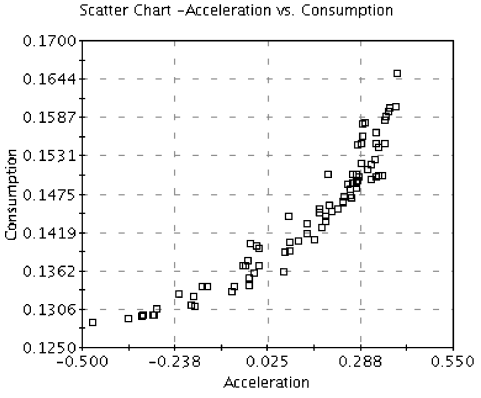
The pareto front is composed of 133 designs. It is presented in Fig. 27. Each picture is a projection of the actual three-dimensional front on each two-dimensional performance plane. That particularly explains the nonconvex shape of the front because the dominance relation has to be understood as three-dimensional.

MMES Pareto-Front Analysis

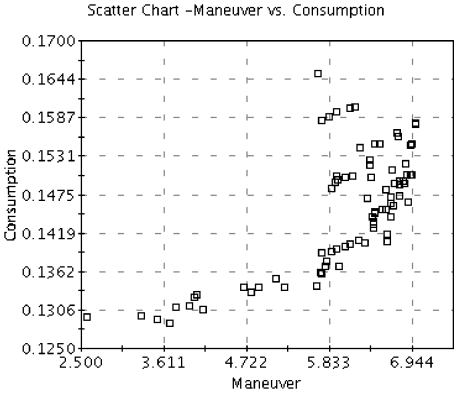
The pareto front is composed of only 40 designs, which is far less than with MOGA. It is presented in Fig. 28. One could notice



a) N_z **vs** N_x

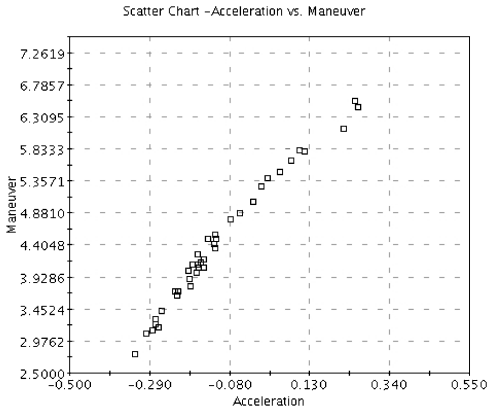


b) m_k **vs** N_x

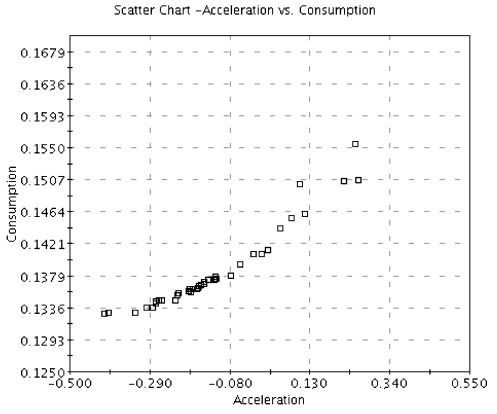


c) m_k **vs** N_z

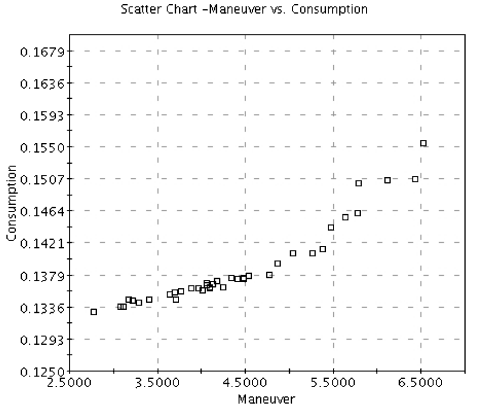
Fig. 27 Pareto front obtained for the MOGA multiobjective optimization.



a) N_z **vs** N_x



b) m_k **vs** N_x



c) m_k **vs** N_z

Fig. 28 Pareto front obtained for the MMES multiobjective optimization.

Table 10 Final performance comparison

Performance	N_x, g	N_z, g	m_k, kg/s
Initial design	−0.229	X	0.1332
A_1 Multiobj.	0.122	5.984	0.1484
A_1 Multiobj.	0.217	4.659	0.1574
Inlet Mono-obj.	0.437	5.709	X
Inlet Mono-obj.	0.134	7.171	0.1524
Inlet Mono-obj.	−0.534	0.845	0.1215
Inlet Multiobj.	0.158	6.237	0.1407
Inlet Multiobj.	0.239	6.901	0.1463
Inlet Multiobj.	0.320	5.959	0.1495
Inlet Multiobj.	0.358	6.447	0.1548

that the evolutionary strategy leads to a more regular pareto front with almost equally distributed designs. But this algorithm is not able to find the boundaries, that is, the extreme values of the pareto front found by the genetic algorithm. For example, no design with an acceleration factor N_x better than 0.3 is found. In this multiobjective case, and with the settings used, the genetic algorithm MOGA seems to be more efficient than the (μ, κ, λ) evolutionary strategy.

Comparison of a Few Feasible Designs

Table 10 presents remarkable designs obtained throught the multiobjective optimizations with their respective performance. The main purpose of the inlet design was to find a shape satisfying the whole mission. One can see that only multiobjective optimizations reach such designs. Whereas the capture-area optimization of the initial design gives results with limited performance, such as $N_x = 0.217$, $N_z = 4.659$, and $m_k = 0.1574$ g, the inlet multiobjective optimization reaches shapes having good global performance: for example, the design leading to $N_x = 0.158$, $N_z = 6.237$, and $m_k = 0.1407$ g emphasizes consumption, whereas the one leading to $N_x = 0.358$, $N_z = 6.447$, and $m_k = 0.1548$ g emphasizes acceleration. From the first one to the second one, an increase of 126% in acceleration is made at the expense of an increase of 10% in fuel consumption. The final choice among the pareto front must rely on other consideration depending on the global project, such as structures, trajectories, stealth, cost, etc.

This clearly shows that multicriteria design optimizations leading to a pareto front are a remarkable source of data in order to find the best tradeoff between several performances.

Conclusions

Complex optimization loops involving several evaluation tools have been used to maximize the aeropropulsive performance of an airbreathing missile design for three flight conditions representative of a complete mission.

Despite the fact that the design space contained a large majority of infeasible points, with several causes of infeasiblility, global stochastic algorithms such as SIMPLEX, evolutionary strategies, and genetic algorithms have proven to be robust enough to efficiently find the optimum shapes.

Performing real multiobjective optimization and finding a pareto front is the only effective way to find a set of designs satisfying several performance criteria in an industrial context and moreover to emphasize the tradeoff between each performance. One should however pay attention to the uncertainties involved in real-world problems, which can be handled for example through a statistical approach[33] that does not change the methodology explained in th present study.

The generic airbreathing missile test case presented in this paper is a good basis to perform studies on optimization. Future work will consist in improving its performance by optimizing the external aerodynamics. Another possible field of interest would be optimizations coupling several aerodynamic performance tools for internal and external coefficients, a ramjet model, and a trajectory module in order to simulate complete missions.

Acknowledgments

This research was performed during August 2002 at the Rutgers University Center for Computational Design Summer Institute with the assistance of D. D. Knight.

Y. Kergaravat and J. N. Gréard from MBDA France are thanked for their contribution to this study.

The modeFRONTIER license was provided by ES.TEC.O. srl. The authors want also to acknowledge C. Lévêque, J. J. Maisonneuve, and J. P. Le Goff from Sirehna S. A., French distributor of modeFRONTIER.

References

[1]Sedon, J., and Goldsmith, E., *Practical Intake Aerodynamic Design*, AIAA, Washington, D.C., 1993.

[2]Mahoney, J., *Inlets for Supersonic Missiles*, AIAA, Washington, D.C., 1999.

[3]Deb, K., *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley, New York, 2001.

[4]Gaiddon, A., and Pagan, D., "Design Methodology for Supersonic Air Intakes," AIAA Paper 2001-3881, July 2001.

[5]Bourdeau, C., Blaize, M., and Knight, D., "Performance Analysis for High-Speed Missile Inlets," *Journal of Propulsion and Power*, Vol. 16, No. 6, 2000, pp. 1125–1131.

[6]Montazel, X., Kergaravat, Y., and Blaize, M., "Navier–Stokes Simulation Methodology for Supersonic Missile Inlets," *Proceedings of the International Symposiumon Airbreathing Engines*, Paper 97-7029, Sept. 1997.

[7]Berkowitz, B., "Information Age Intelligence," *Foreign Policy*, Vol. 103, June 1996, pp. 35–50.

[8]Shukla, V., Gelsey, A., Schwabacher, M., Smith, D., and Knight, D., "Automated Design Optimization for the P2 and P8 Hypersonic Inlets," *Journal of Aircraft*, Vol. 34, No. 2, 1997, pp. 228–235.

[9]Zha, G.-C., Smith, D., Schwabacher, M., Rasheed, K., Gelsey, A., and Knight, D., "High-Performance Supersonic Missile Inlet Design Using Automated Optimization," *Journal of Aircraft*, Vol. 34, No. 6, 1997, pp. 697–705.

[10]Blaize, M., Knight, D., and Rasheed, K., "Automated Optimal Design of Two-Dimensional Supersonic Missile Inlets," *Journal of Propulsion and Power*, Vol. 14, No. 6, 1998, pp. 890–898.

[11]Kergaravat, Y., Vives, E., and Knight, D., "Inlet/Body Integration Preliminary Design for Supersonic Air-Breathing Missiles Using Automated Multi-Disciplinary Optimization," *Proceedings of the RTO AVT Symposium on Aerodynamic Design and Optimization of Flight Vehicles in a Concurrent Multi-Disciplinary Environment*, RTO-AVT, Ottawa, Oct. 1999.

[12]Gaiddon, A., and Knight, D., "Multi-Criteria Design Optimization of Integrated Three-Dimensional Supersonic Inlets," *Journal of Propulsion and Power*, Vol. 19, No. 3, 2003.

[13]Knight, D., "Application of Genetic Algorithms to High Speed Air Intake Design," Lecture Series Programme 1999–2000, Von Karman Inst. for Fluid Dynamics, Rhode-Saint-Genese, VKI LS 2000-07, Belgium, May 2000.

[14]Knight, D., "Automated Optimal Design of Supersonic and Subsonic Diffusers Using CFD," ECCOMAS 2000, Barcelona, Spain, Sept. 2000.

[15]*GASPex v4.1 Reference Guide*, AeroSoft, Inc., Blacksburg, VA, 2001.

[16]*Hexa User's Manual*, ICEMCFD Engineering GmbH, München, Germany, 1999.

[17]Love, E., "Base Pressure at Supersonic Speeds on Two-Dimensional Airfoils and on Bodies of Revolution," NACA TN 3819, Jan. 1957.

[18]Carrière, P., "Aérodynamique Interne des Réacteurs," ENSAE Supaero, Toulouse, France, July 1977.

[19]Pagan, D., and Lacau, R., "Missile Aerothermodynamics and Propulsion Integration," AGARD, AGARD-R-813, Orléans, France, Oct. 1996.

[20]Nelder, J., and Mead, R., "A Simplex Method for Function Minimization," *The Compututer Journal*, Vol. 7, No. 7, 1965, pp. 308–313.

[21]Collins, R., and Jefferson, D., "Selection in Massively Parallel Genetic Algorithms," *Fourth International Conference on Genetic Algorithms*, edited by Morgan-Kaufman, San Matco, CA, 1991.

[22]Yamamoto, K., and Inoue, O., "New Evolutionary Direction Operator for Genetic Algorithms," *AIAA Journal*, Vol. 33, No. 10, 1995, pp. 1990–1993.

[23]Horn, J., and Nafpliotis, N., "Multiobjective Optimization Using the Niched Pareto Genetic Algorithm," IlliGAL, Rept. 93005, Univ. of Illinois at Urbana–Champaign, Urbana, April 1993.

[24]Belegundu, A. D., Murthy, D. V., Salagame, R. R., and Constants, E. W., "Multiobjective Optimization of Laminated Ceramic Composites Using Genetic Algorithms," Fifth AIAA/USAF/NASA Symposium on Multidisciplinary Analysis and Optimization, AIAA Paper 84-4363-CP, Panama City, Florida, 1994, pp. 1015–1022.

[25]Michielssen, E., and Weile, D., "Electromagnetic System Design Using Genetic Algorithms," *Genetic Algorithms in Engineering and Computer Science*, Wiley, New York, 1995, pp. 345–369.
[26]Goldberg, D., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Longman, Reading, MA, 1998.
[27]Hajela, P., "Genetic Search Strategies in Multicriterion Optimal Design," *Structural Optimization*, Vol. 4, Springer-Verlag, Berlin, 1992, pp. 99–107.
[28]Schwefel, H., *Evolution and Optimum Seeking*, Wiley, New York, 1995.
[29]Rechenberg, I., *Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution*, Fromman-Hozlboog Verlag, Stuttgart, 1973.
[30]Bourdeau, C., Blaize, M., and Knight, D., "Performance Analysis for an Automated Optimal Design of High Speed Inlets," AIAA Paper 99-0611, Jan. 1999.
[31]Johnston, J., "Diffuser Design and Performance Analysis by Unified Integral Methods," AIAA Paper 97-2733, July 1997.
[32]Sobol, I., *Die Monte-Carlo Methode*, Deutscher Verlag der Wissenschaften, Berlin, Germany, 1991.
[33]Gaiddon, A., Gréard, J., and Pagan, D., "Automated Optimization of Supersonic Missile Performance Taking into Account Design Uncertainties," AIAA Paper 2003-3879, June 2003.