Chapter 1
CFD-based Optimization for Automotive Aerodynamics

Laurent Dumas

Abstract The car drag reduction problem is a major topic in the automotive industry because of its close link with fuel consumption reduction. Until recently, a computational approach of this problem was unattainable because of its complexity and its computational cost. A first attempt in this direction has been presented by the present author as part of a collaborative work with the French car manufacturer Peugeot Citroën PSA [4]. This article described the drag minimization of a simplified 3D car shape with a global optimization method that coupled a Genetic Algorithm (GA) and a second-order Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. The present chapter is intended to give a more detailed version of this work as well as its recent improvements. An overview of the main characteristics of automotive aerodynamics and a detailed presentation of the car drag reduction problem are respectively proposed in Sects. 1.1 and 1.2. Section 1.3 is devoted to the description of various fast and global optimization methods that are then applied to the drag minimization of a simplified car shape discussed in Sect. 1.4. Finally in Sect. 1.5, the chapter ends by proposing the applicability of CFD-based optimization in the field of airplane engines.

Laurent Dumas
Université Pierre et Marie Curie
Laboratoire Jacques-Louis Lions
4, place Jussieu, 75230 Paris Cedex 05
(e-mail: laurent.dumas@upmc.fr)

1.1 Introducing Automotive Aerodynamics

1.1.1 A Major Concern for Car Manufacturers

In the past, the external shape of cars has evolved particularly for safety reasons, comfort improvement and also aesthetic considerations. Consequences of these guidelines on car aerodynamics were not of major concern for many years. However, this situation changed in the 70's with the emergence of the oil crisis. To promote energy conservation, studies were carried out and it was discovered that the amount of the aerodynamic drag in the fuel consumption ranges between 30% during an urban cycle and 75% at a 120 km/h cruise speed. Since then, decreasing the drag force acting on road vehicles and thus their fuel consumption, became a major concern for car manufacturers. Growing ecological concerns within the last decade further make this a critically relevant issue in the automotive research centers.

The process of drag creation and the way to control it was first discovered experimentally. In particular, it was found that the major amount of drag was due to the emergence of flow separation at the rear surface of cars. Unfortunately, unlike in aeronautics where it can be largely excluded from the body surface, this aerodynamic phenomenon is an inherent problem for ground vehicles and can not be avoided. Moreover, the associated three-dimensional flow in the wake behind a car exhibits a complex 3D behavior and is very difficult to control because of its unsteadiness and its sensitivity to the car geometry.

The pioneering experiments of Morel and Ahmed done in the late 70's on simplified geometries also called bluff bodies, are now described in Sects. 1.1.2 to 1.1.4.

1.1.2 Experiments on Bluff Bodies

Two major experiments have been done on bluff bodies, the first one by Morel in 1978 [18] and the second by Ahmed in 1984 [1]. The objective was to study the flow behavior around cars with a particular type of rear shape called hatchback or fastback. These experiments are even now used as a reference in many numerical studies [8, 10, 12, 13, 16].

The bluff body used by Ahmed, similar to the one used by Morel, is illustrated in Fig. 1.1. It has the same proportions as a realistic car but with sharp edges. More precisely, the ratio of length/width/height is equal to 3.33/1.5/1. In both cases, the rear base is interchangeable by modifying the slant angle denoted here as α. The Reynolds numbers are taken equal to 1.4×10^6 and 4.29×10^6 in the Morel and Ahmed experiments, respectively.

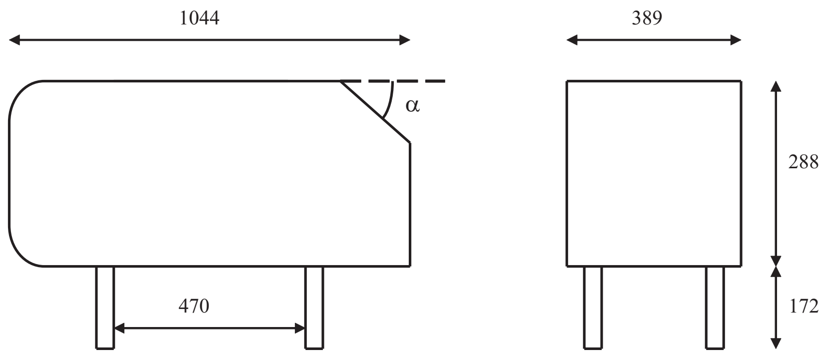

Fig. 1.1 The Ahmed bluff body

1.1.3 Wake Flow Behind a Bluff Body

The most difficult flow region to predict is located at the wake of the car where recirculation and separation occur. It is also the region which is responsible for most of the car drag (see Sect. 1.1.4).

In a time-averaged sense, two distinct regimes depending on the slant angle α, called Regime I and II, have been observed in the experiments done by Morel and Ahmed. The value of the critical angle α_c between both regimes is approximately equal to 30 degrees in each experiment but can slightly change depending on the Reynolds number and the exact geometry.

- Regime I ($\alpha_c < \alpha < 90°$): In this case, the flow exhibits a full 3D behavior with a separation area including the whole slant and base area. The recirculation zones, coming from the four parts of the car (roof, floor and the two base sides) gather and form a pair of horseshoe vortices situated one above another in a separation bulb (see zones A and B in Fig. 1.2). Vortices, coming off the slant side edges are also present (zone C in Fig. 1.2).
- Regime II ($0 < \alpha < \alpha_c$): For low values of α, the flow remains two-dimensional and separates only at the rear base. Two counter-rotating vortices appear from the roof and the floor similar to what happens around airfoils. When α increases up to α_c, the flow becomes three-dimensional because of the appearance of two longitudinal vortices issued from the side walls of the car.

The critical value of α_c corresponds to an unstable configuration associated with a peak in the drag coefficient (see Sect. 1.1.4). In this case, a slight change can generate a high modification of the wake flow. For these reasons, it is essential to avoid such angle value in the design of real cars.

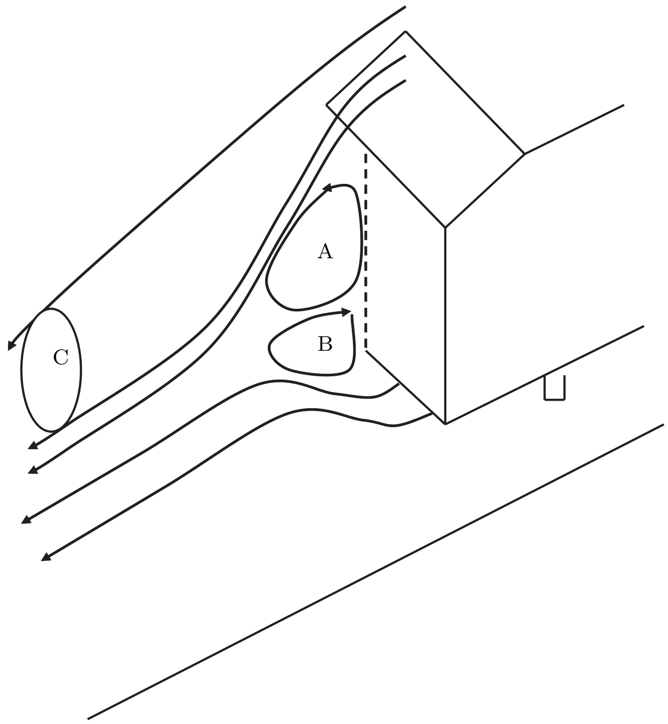

Fig. 1.2 Wake flow behavior behind Ahmed's bluff body

1.1.4 Drag Variation with the Slant Angle

A dimensionless coefficient, called drag coefficient and related to the drag force acting on the bluff body, is defined as follows:

$$C_d = \frac{F_d}{\frac{1}{2}\rho V_\infty^2 S} \ . \tag{1.1}$$

In this expression, ρ represents the air density, V_∞ is the freestream velocity, S is the cross section area and F_d is the total drag force acting on the car projected on the longitudinal direction. Note that the drag force F_d can be decomposed into a sum of a viscous drag force and a pressure drag force.

A first striking result observed by Morel is that the slant surface and the rear base are responsible for more than 90% of the pressure drag force. Moreover, the latter represents more than 70% of the total drag force. These observations have been confirmed by the Ahmed experiment where only 15% to 25% of the drag is due to the viscous drag. Such results can be explained by the analysis of the wake flow discussed in Sect. 1.1.3, that is, the large separation area will induce the major part of the total drag force.

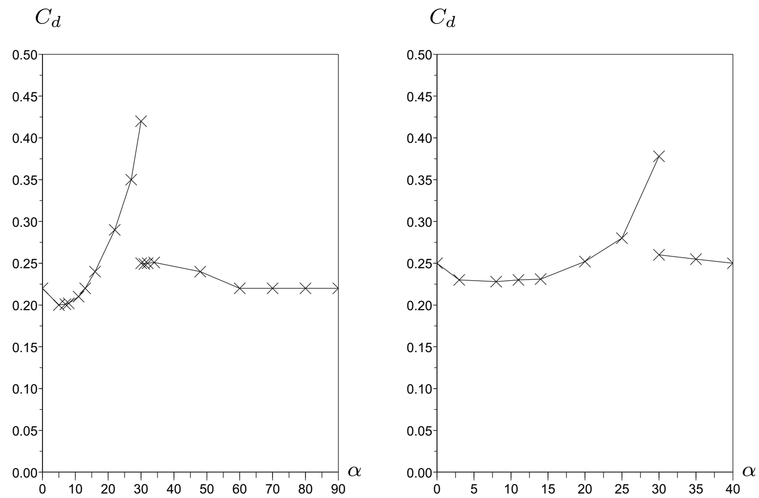

Fig. 1.3 Drag measured for the Morel (left) and Ahmed (right) bluff-body for various slant angles

The variations of the drag coefficient for the Morel and the Ahmed bluff body with respect to the slant angle α are displayed in Fig. 1.3.

Both graphs exhibit the same variations, in particular with a peak value at a critical angle α_c near 30 degrees, already introduced in the previous subsection. The shape of the curve can also be explained by referring to the wake flow behind the car: for small values of α for which the flow is two-dimensional, the drag is directly linked to the dimensions of the separated area. Then, when the flow becomes three dimensional, the separation bulb that appears absorbs a growing amount of the flow energy, thus leading to a large increase of the drag coefficient until α reaches α_c. Above this value, the airflow no longer feeds the vortex systems. Consequently, the static and total pressure experience a sudden rise at the base thus drastically reducing the drag coefficient almost to its value at a zero slant angle.

1.2 The Drag Reduction Problem

After the description in the previous section of the main features of automotive aerodynamics, the car drag reduction problem is now stated for real cars in Sect. 1.2.1. The numerical modelization of this problem in view of an automatic drag minimization is then presented in Sect. 1.2.2.

Table 1.1 Examples of drag coefficient of old or modern cars

Car	C_d
Ford T (1908)	0.8
Hummer H2 (2003)	0.57
Citroen SM (1970)	0.33
Peugeot 407 (2004)	0.29
Tatra T77 (1935)	0.212

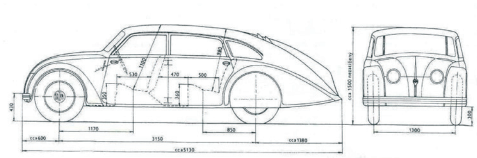

Fig. 1.4 Side and front view of Tatra T77 (1935, $C_d = 0.212$). With permission of Tatra Auto Klub, Slovakia

1.2.1 Drag Reduction in the Automotive Industry

The dimensionless drag coefficient C_d defined in Eq. (1.1), is the main coefficient for measuring the aerodynamic performance of a given car. Examples of values of C_d for past or existing cars are presented in Table 1.1.

It can be seen from this table that the drag coefficient of cars has been decreasing during the last century even though other considerations like comfort or aesthetics have also been taken into account to popularize a high drag model (see Hummer H2) or abandon a low drag model (see Tatra T77).

The last model of Tatra T77 had a remarkable low drag value of 0.212. A schematic side and front view of this car is illustrated in Fig. 1.4. The short forebody compared to the extended tailored rear shape conforms to the experimental observations stated in the previous section, saying that the separation zone at the slant and base area, largely reduced here, is responsible for the major part of C_d.

For a standard car, Table 1.2 displays the repartition of the relative contribution of various elements on the total drag.

It shows in particular that 70% of the drag coefficient depends on the external shape. Such a large value justifies the interest of a numerical modelization of the problem in order to find numerically innovative external shapes that will largely reduce the drag coefficient.

Table 1.2 Drag repartition on a realistic car

Position	percent of C_d
Upper surface	40%
Lower surface	30%
Wheels	15%
Cooling	10%
Others	5%

1.2.2 Numerical Modelization

1.2.2.1 The Navier-Stokes Equations

The incompressible Navier-Stokes equations govern the flow around the car shape. Denote S_c the car surface, G the ground surface and Ω a large volume around S_c and above G, then using the Einstein notation, the Navier-Stokes equations are written as follows:

- Incompressibility:

$$\frac{\partial u_i}{\partial x_i} = 0 \quad \text{on } \Omega \tag{1.2}$$

- Momentum ($1 \leq j \leq 3$):

$$\frac{\partial u_j}{\partial t} + \frac{\partial u_i u_j}{\partial x_i} = -\frac{1}{\rho}\frac{\partial p}{\partial x_j} + \frac{\partial}{\partial x_i}\left[\nu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \quad \text{on } \Omega \tag{1.3}$$

where $u_j(t,x)$, $p(t,x)$ and ρ are respectively the flow velocity, pressure and density. The boundary conditions for the velocity are of Dirichlet type:

$$u_i = 0 \quad \text{on } S_c \cup G \quad \text{and} \quad u_i = V_\infty \quad \text{on } \partial\Omega \setminus (S_c \cup G)\ . \tag{1.4}$$

As the Reynolds number is very high in real configurations, usually more than 10^6, a turbulence model must be added. This model must be of reduced computational cost in view of the large number of simulations, more than 100, that need to be done during the optimization process. This explains why the Large Eddy Simulation (LES) model can not be used here (see [8] and references herein for an example of the application of LES for a single computation around the Ahmed bluff-body).

A Reynolds-Averaged Navier-Stokes (RANS) turbulence model which consists of averaging the previous equations (1.2) and (1.3), is chosen here. Denoting $\bar{u}_i$ the averaged velocity, a closure principle for the term $\overline{u_i u_j}$ has to be defined. The most popular way to do it leads to the well known k–ε model. In this model, the averaged equations (1.3) are rewritten as:

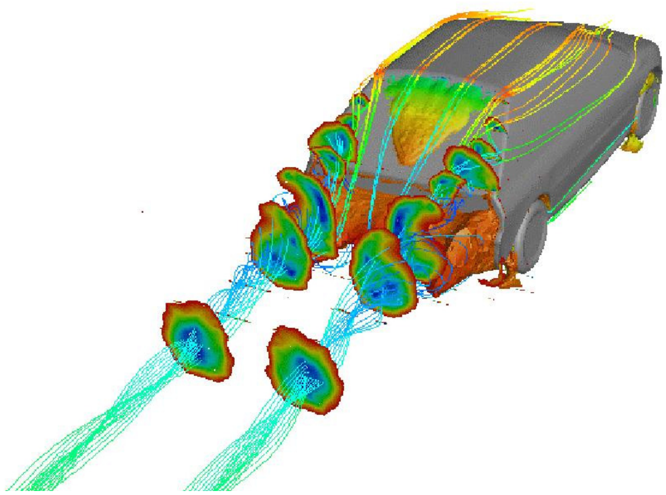

Fig. 1.5 Numerical flow field around a realistic car (Peugeot 206)

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_i} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_j} + \frac{\partial}{\partial x_i}\left[(\nu + \nu_t)\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) - \frac{2}{3}\delta_{i,j} k\right] \quad \text{on } \Omega \tag{1.5}$$

where ν_t is called the eddy viscosity and is related to the turbulent kinetic energy k and its rate of dissipation ε by

$$\nu_t = C_\mu \frac{k^2}{\varepsilon} \ . \tag{1.6}$$

In this expression, C_μ is a constant and the new variables k and ε are obtained from a set of two equations (see [17]).

In the present configuration, it has been observed that the second-order closure model called Reynolds Stress Model (RSM) with an adequate wall function gives better results than the k–ε method (see [16]). This model will be preferred in the forthcoming simulations despite the small computational overcost on the order of 40% A first example of such flow computations around a realistic car is given in Fig. 1.5.

1.2.2.2 The Cost Function to Minimize

The cost function that will have to be minimized is the drag coefficient already introduced in (1.1). It can be rewritten in the following way after separating the pressure part and the viscous part:

$$C_d = \frac{\iint_{S_c} (p - p_\infty)\boldsymbol{n} d\sigma}{\frac{1}{2}\rho V_\infty^2 S} + \frac{\iint_{S_c} \boldsymbol{\tau} \cdot \boldsymbol{\nu} d\sigma}{\frac{1}{2}\rho V_\infty^2 S} \tag{1.7}$$

where $\boldsymbol{n}$ is the normal vector, $\boldsymbol{\tau}$ the viscous stress tensor and $\boldsymbol{\nu}$ is the projection of the velocity vector to the element of shape $d\sigma$.

The optimization will thus consist to reduce this drag coefficient by changing the car shape S_c and particularly its rear shape. Of course, two types of constraints will have to be added: geometric type (on volume, total length or cross section) and aerodynamic type (by fixing other aerodynamic moments for instance).

The numerical computation of the drag coefficient being very costly and very sensitive to the rear geometry explains why the numerical approach of the drag reduction problem has been for so long unattainable. The next section that presents fast and global optimization methods tries to make it possible.

1.3 Fast and Global Optimization Methods

There exists many methods for minimizing a cost function J defined from a set $\mathcal{O} \subset \mathbb{R}^n$ to $\mathbb{R}_+$. Among them, the family of evolutionary algorithms, including the well known methods of Genetic Algorithms (GAs) and Evolution Strategies (ES) whose main principles are recalled in the next subsection, has the major advantage to seek for a global minimum. Unfortunately, in view of the drag reduction problem that will be considered in Sect. 1.4, this type of method needs to be improved because of the large number of cost function evaluations that is needed. The hybrid optimization methods presented in Sect. 1.3.2 greatly reduce this time cost by coupling an evolutionary algorithm with a deterministic descent method. Another way to speed up the convergence of an evolutionary algorithm is described in Sect. 1.3.3 and aims at doing fast but approximated evaluations during the optimization process. All these methods are validated in Sect. 1.3.4 on classical analytic test functions.

1.3.1 Evolutionary Algorithms

The family of evolutionary algorithms gathers all stochastic methods that have the ability to seek for a global minimum of an arbitrary cost function. Among them, the population-based methods of GAs and ES are widely used in many applications and will serve as the core tool in the “real world” applications presented in Sects. 1.4 and 1.5, respectively. Their main principles are recalled in the next two paragraphs.

1.3.1.1 Genetic Algorithms (GA)

GAs are global optimization methods directly inspired from the Darwinian theory of evolution of species [11]. They require following the evolution of a certain number N_p of possible solutions, also called population. A fitness value is associated to each element (or individual) $x_i \in \mathcal{O}$ of the population that is inversely proportional to $J(x_i)$ in case of a minimization problem. The population is regenerated N_g times by using three stochastic principles called selection, crossover and mutation, that mimic the biological law of "survival of the fittest".

The GA that will be used in the drag reduction problem in Sect. 1.4 acts in the following way: at each generation, $N_p/2$ couples are selected by using a roulette wheel process with respective parts based on the fitness rank of each individual in the population. To each selected couple, the crossover and mutation principles are then successively applied with a respective probability p_c and p_m. The crossover of two elements consists in creating two new elements by doing a barycentric combination of them with random and independent coefficients in each coordinate. The mutation principle consists of replacing a member of the population by a new one randomly chosen in its neighborhood. A one-elitism principle is added in order to be sure to keep in the population the best element of the previous generation.

Thus, the algorithm can be written as:

- *Choice of an initial population* $P_1 = \{x_i^1 \in \mathcal{O}, 1 \leq i \leq N_p\}$
- $n_g = 1$. *Repeat until* $n_g = N_g$
- *Evaluate* $\{J(x_i^{n_g}), 1 \leq i \leq N_p\}$ *and* $m = \min\{J(x_i^{n_g}), 1 \leq i \leq N_p\}$
- 1-*elitism: if* $n_g \geq 2$ *&* $J(X_{n_g-1}) < m$ *then* $x_i^{n_g} = X_{n_g-1}$ *for a random* i
- *for* k *from* 1 *to* $N_p/2$
-     *Selection of* $(x_\alpha^{n_g}, x_\beta^{n_g})$ *with a roulette wheel process*
-     *with probability* p_c*: replace* $(x_\alpha^{n_g}, x_\beta^{n_g})$ *by* $(y_\alpha^{n_g}, y_\beta^{n_g})$ *by crossover*
-     *with probability* p_m*: replace* $(y_\alpha^{n_g}, y_\beta^{n_g})$ *by* $(z_\alpha^{n_g}, z_\beta^{n_g})$ *by mutation*
- *end for*
- $n_g = n_g + 1$.
- *Generate the new population* P_{n_g}.
- *Call* X_{n_g} *the best element.*

1.3.1.2 Evolution Strategies (ES)

Evolution Strategies (ES) have been first introduced by H.P. Schwefel in the 60's [2]. As it is the case for GAs, it requires following the evolution of a population of potential solutions through the same three stochastic principles, selection, recombination and mutation. However, unlike the GAs, the major process is the mutation process and the selection is made deterministic.

The Evolution Strategy that will be used in the application of Sect. 1.5 is based on the $(\mu+\lambda)$ selection principle and on the 1/5th rule for the mutation strength. An intermediate recombination with two parents is also included. The algorithm is thus written as:

- *Choice of an initial population of* μ *parents:* $P_1 = \{x_i^1 \in \mathcal{O}, \quad 1 \leq i \leq \mu\}$
- $n_g = 1$. *Repeat until* $n_g = N_g$
- *Creation of a population of* $\lambda \geq \mu$ *offsprings* O_{n_g} *by:*
- *Recombination:* $y_i^{n_g} = \frac{1}{2}(x_\alpha^{n_g} + x_\beta^{n_g})$
- *Normal mutation:* $z_i^{n_g} = y_i^{n_g} + \mathcal{N}(0, \sigma)$
- *Update of the mutation strength* σ *with the* $1/5th$ *rule*
- *Evaluate* $\{J(z_i^{n_g}), z_i^{n_g} \in O_{n_g}\}$.
- $n_g = n_g + 1$.
- *Selection of the best* μ *new parents in the population* $P_{n_g} \cup O_{n_g}$.
- *Call* X_{n_g} *the best element.*

1.3.2 Adaptive Hybrid Methods (AHM)

In order to improve the convergence of evolutionary algorithms for time-consuming applications like the drag reduction problem presented in Sect. 1.4, the idea of coupling a population-based algorithm with a deterministic local search, for instance a descent method, has been explored for many years (see e.g., [20]). However, the obtained gain can be very different from one function to another, depending on the level of adaptivity of the coupling and the way it is done.

The method presented here called Adaptive Hybrid Method (AHM) whose general principles are summarized in Fig. 1.6, tries to remedy these drawbacks by answering in a fully adaptive way the three fundamental questions in the construction of a hybrid method: when to shift from global to local, when to return to global and to which elements apply a local search. This method includes some criteria introduced in [7], and defines new ones as the reduced clustering strategy.

Note that this adaptive coupling can be implemented with any type of population-based global search methods (GA, ES, etc.) and any type of deterministic local search methods (steepest descent, BFGS, etc.).

From global to local

The shift from a global search to a local search is useful when the exploration ability of the global search is no longer efficient. With this aim, a statistical coefficient associated to the cost function repartition values is introduced. It is equal to the ratio of the mean evaluation of the current population to its

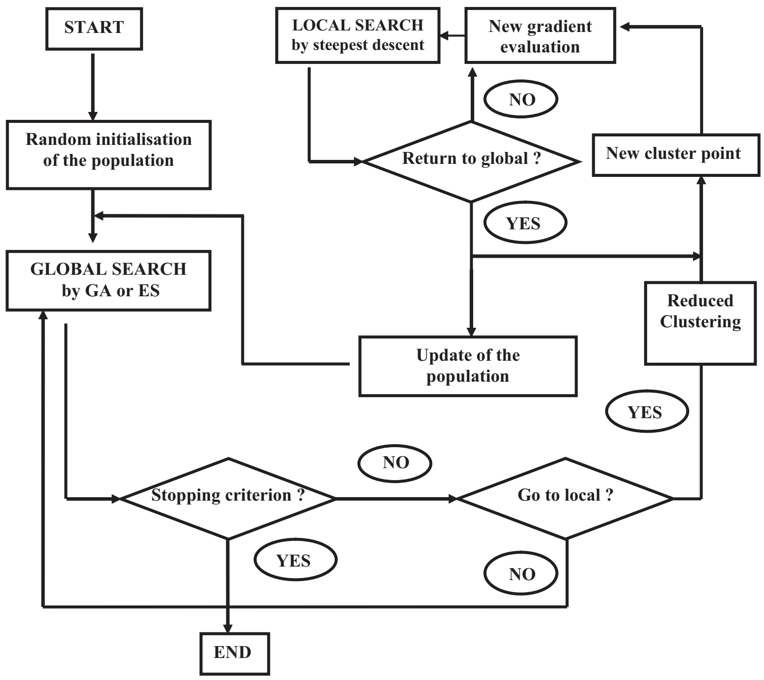

Fig. 1.6 General principle of the AHM

corresponding standard deviation computed with its variance:

$$CV = \frac{m}{\sigma} = \frac{\text{mean}\{J(x),\, x \in P_{n_g}\}}{\sqrt{\text{var}\{J(x),\, x \in P_{n_g}\}}} \tag{1.8}$$

and is named coefficient of variation CV. A local search will be utilized when this ratio increases within two consecutive generations of the evolutionary algorithm (either GA or ES).

From local to global

The local search is aimed at locally decreasing the cost function more efficiently than the random mutation. However, this gain must be counterbalanced after each evaluation with a characteristic gain of the global method. More precisely, the local search will continue here while:

$$G_{local} > G_{global}$$

where G_{local} is equal to the gain when passing from a point to the next one in the steepest descent algorithm and G_{global} is the gain of the last global phase evaluated with the decrease of m in formula (1.8). Both gains are scaled with the number of evaluations of the cost function needed to achieve them.

Reduced clustering

In order to spread as much as possible the local search in the whole domain, the population is divided into a certain number of sub-populations called clusters. To do so, a very classical and fast algorithm is used where each cluster is constructed such that all its associated elements are closer to its center of mass than to any other. After this preliminary step called clustering, the local search is applied to the best element (with respect to J) of each cluster.

A careful study of the appropriate number of clusters had never been done yet even though it appears to be rather important for the algorithm performance. To overcome the difficulty of choosing this number, a new method called reduced clustering has been proposed in [3] where the number of clusters is progressively decreased during the optimization process. It corresponds to the natural idea that the whole process will progressively focus on a reduced number of local minima. To do so, a deterministic rule of arithmetic decrease plus an adaptive strategy including the aggregation of too near clusters has been considered here and exhibits better results than any case with a fixed number of clusters as shown in Sect. 1.3.4.

1.3.3 Genetic Algorithms with Approximated Evaluations (AGA)

Another idea to speed up the convergence of an evolutionary algorithm when the computational time of the cost function $x \mapsto J(x)$ is high, is to take benefit of the large and growing data base of exact evaluations by making fast and approximated evaluations $x \mapsto \tilde{J}(x)$ leading to what is called surrogate or meta-models (see [9, 14, 15, 19]). In the present work, the chosen strategy is required to perform exact evaluations only for all the best fitted elements of the population (in the sense of $\tilde{J}$) and for one randomly chosen element. The new algorithm, called AGA is thus deduced from the algorithm of Sect. 1.3.1 by changing the evaluation phase into the following:

- *if* $n_g = 1$ *then make exact evaluations* $\{J(x_i^{n_g}), 1 \leq i \leq N_p\}$
- *elseif* $n_g \geq 2$
-   *for* i *from* 1 *to* N_p
-     *Make approximated evaluations* $\tilde{J}(x_i^{n_g})$.

- *if* $\tilde{J}(x_i^{n_g}) < J(X_{n_g-1})$ *then make an exact evaluation of* $J(x_i^{n_g})$
- *end for*
- *for a random i: make an exact evaluation of* $J(x_i^{n_g})$
- *end elseif*

The interpolation method chosen here comes from the field of neural networks and is called Radial Basis Function (RBF) interpolation [9]. Suppose that the function J is known on N points $\{T_i, 1 \leq i \leq N\}$, the idea is to approximate J at a new point x by making a linear combination of radial functions of the type:

$$\tilde{J}(x) = \sum_{i=1}^{n_c} \psi_i \Phi(||x - \hat{T}_i||) \tag{1.9}$$

where:

- $\{\hat{T}_i, 1 \leq i \leq n_c\} \subset \{T_i, 1 \leq i \leq N\}$ is the set of the $n_c \leq N$ nearest points to x for the euclidian norm $||.||$, on which an exact evaluation of J is known.
- Φ is a radial basis function chosen in the following set:

$$\Phi_1(u) = \exp(-\frac{u^2}{r^2}),$$

$$\Phi_2(u) = \sqrt{u^2 + r^2},$$

$$\Phi_3(u) = \frac{1}{\sqrt{u^2 + r^2}},$$

$$\Phi_4(u) = \exp(-\frac{u}{r}),$$

for which the parameter $r > 0$ is called the attenuation parameter.

The scalar coefficients $(\psi_i)_{1 \leq i \leq n_c}$ are obtained by solving the least square problem of size $N \times n_c$:

$$\text{minimize} \quad \text{err}(x) = \sum_{i=1}^{N} (J(T_i) - \tilde{J}(T_i))^2 + \lambda \sum_{j=1}^{n_c} \psi_j^2$$

where $\lambda > 0$ is called the regularization parameter.

In order to attenuate or even remove the dependence of this model to its attached parameters, a secondary global optimization procedure (a classical GA) has been over-added in order to determine for each x, the best values (with respect to $\text{err}(x)$) of the parameters n_c, $r \in [0.01, 10]$, $\lambda \in [0, 10]$ and $\Phi \in \{\Phi_1, \Phi_2, \Phi_3, \Phi_4\}$. As this new step introduces a second level of global optimization, it is only reserved to cases where the time evaluation of $x \mapsto J(x)$ is many orders of magnitude higher than the time evaluation of $x \mapsto \tilde{J}(x)$, as in the car drag reduction problem.

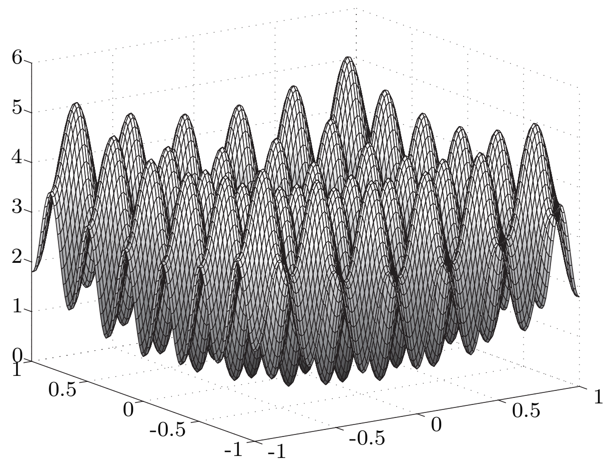

Fig. 1.7 The Rastrigin function $Rast_2$

1.3.4 Validation on Analytic Test Functions

Before applying them on real world applicative problems, all the previous global optimization algorithms have been tested and compared on various analytic test functions and among them the well-known Rastrigin function with n parameters:

$$Rast_n(x) = \sum_{i=1}^{n} \left(x_i^2 - \cos(2\pi x_i) \right) + n \tag{1.10}$$

defined on $\mathcal{O} = [-5, 5]^n$, for which there exists many local minima and only a global minimum located at $x_m = (0, ..., 0)$ and equal to 0 (see Fig. 1.7).

1.3.4.1 AHMs vs. Evolutionary Algorithms

A rather exhaustive comparison has been made between the classical evolutionary algorithm ES and the AHM introduced in Sect. 1.3.2. The statistical results are summarized in Table 1.3 for the Rastrigin function with 6 parameters. In this table, the success rate represents the rate of runs which were able to locate the correct attraction basin of the global minimum after a given number of evaluations of the cost function (respectively 500, 1000 and 2000). Note that any gradient evaluation counts for n evaluations of the cost function as it is the case in a finite-difference approximation.

Table 1.3 Comparison of ES and AHM for the Rastrigin function $Rast_6$

Method	Mean best (success rate) 500 evaluations	same after 1000 evaluations	same after 2000 evaluations
ES	6.47(0%)	3.46(0.5%)	0.46(80.5%)
AHM, 4 clust.	4.97(2.5%)	1.86(6%)	0.47(63.5%)
AHM, 8 clust.	3.54(7%)	1.77(11%)	0.34(67%)
AHM	3.52(10.5%)	1.51(17%)	0.34(68%)

As can be seen from this table, any AHM overperforms the ES at the early stage of the process in both performance criteria. Moreover, during this phase, the strategy of reduced clustering (last line in Table 1.3) significantly improves the results of a hybrid algorithm with a fixed number of clusters. When a large number of evaluations are done, the pure ES takes a slight advantage on the number of successes (but not on the mean best value) in this special case compared to any AHM.

These results can be summarized by saying that a hybrid algorithm will hasten convergence by enhancing the best elements in the population but on the other hand, such strategy can sometimes lead to a premature convergence. However, such drawback may not be too critical in a real applicative situation as it has only been observed with very special functions with a huge number of local minima like the Rastrigin function. Moreover, the main performance criterion of an algorithm for industrial purposes is its ability to achieve the best decrease of the cost function for a given amount of computational time.

1.3.4.2 AGA vs. GAs

Another statistical study has also been realized on the Rastrigin function with 3 parameters in order to compare the GA in Sect. 1.3.1 and the so-called AGA in Sect. 1.3.3. In order to achieve a quasi-certain convergence, the population number is fixed equal to $N_p = 30$ whereas the crossover and mutation probability are set to $p_c = 0.3$ and $p_m = 0.9$.

In this case, the average gain of an AGA compared to a classical GA is nearly equal to 4. It means that on average, the number of exact evaluations to achieve a given convergence level has been divided by a factor of 4.

In view of their promising results, both global optimization methods AHM and AGA are now used in the next two sections for solving realistic optimization problems.

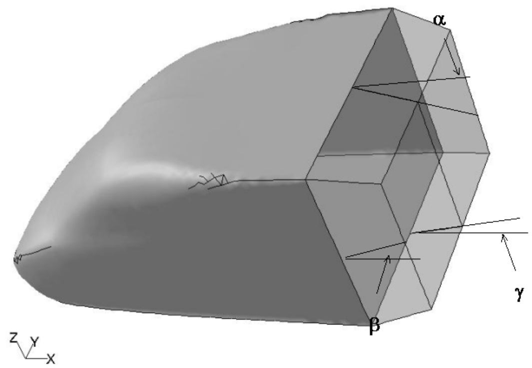

Fig. 1.8 3D car shape parametrized by its three rear angles α, β and γ

1.4 Car Drag Reduction with Numerical Optimization

In this section, the main results obtained on a car drag reduction problem are presented. All of them have been done in collaboration with two research engineers from Peugeot Citroën PSA, V. Herbert and F. Muyl, and have already been published in various journals ([4, 5, 6]).

1.4.1 Description of the Test Case

In order to test the fast and global optimization methods presented in Sect. 1.3 on a realistic car drag reduction problem, a simplified car geometry has been extensively studied. It comprises minimizing the drag coefficient, also called C_d and defined in Eq. (1.7), of a simplified car shape with respect to the three geometrical angles defining its rear shape (see Fig. 1.8): the slant angle (α), the boat-tail angle (β) and the ramp angle (γ). The forebody of the vehicle is fixed and closely resembles the shape of an existing vehicle, namely the Xsara Picasso from Citroën. The objective is thus to find the best rear shape that will reduce the total drag coefficient of the car, ignoring any aesthetic considerations. As it has been previously seen in Sect. 1.1 on the Ahmed bluff body, it is expected that modifying the rear shape will lead to a very important drag reduction.

1.4.2 Details of the Numerical Simulation

An automatic optimization loop has been implemented and is summarized in Fig. 1.9. This loop includes the following steps:

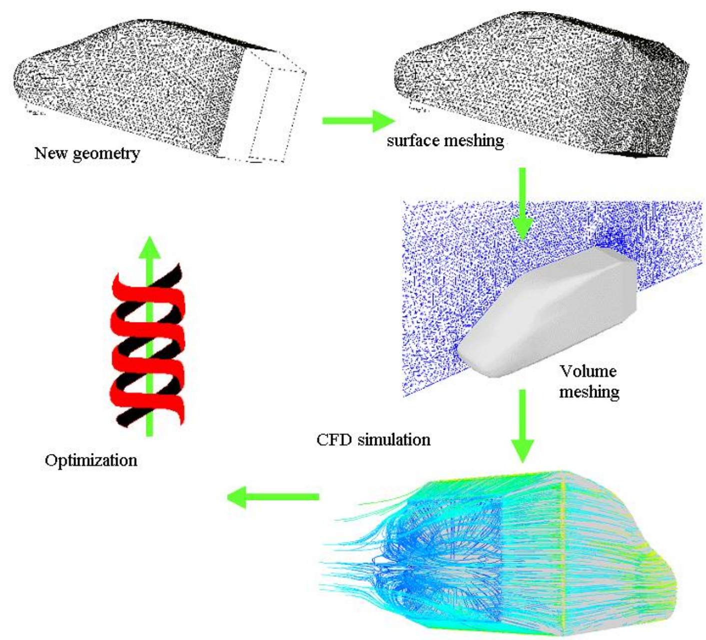

Fig. 1.9 General principles of the automatic optimization loop

(i) Car shape generation and meshing

In view of the experimental results obtained from using low drag car shapes presented in section 2, the three rear angles are sought in the following intervals (in degrees):

$$(\alpha, \beta, \gamma) \in [15, 25] \times [5, 15] \times [15, 25] \ .$$

For any given geometry, the 3D-mesh around the car shape is generated with the commercial grid generator Gambit. It contains a total of approximately 6 million cells that include both tetrahedrons and prisms. In order to simulate accurately the flow field behind the car which is responsible for the major part of the drag coefficient, the mesh is refined particularly in this region.

(ii) CFD simulation

The commercial CFD code Fluent is used for the computation of the flow field around the car. The Reynolds number based on the body length (3.95 m) and the velocity at infinity (40 m/s) is taken equal to 4.3×10^6 as in the Ahmed experiment [1]. The 7-equation RSM turbulence model is chosen as it gives better results in this case than the classical k–ε model (see [16]). The com-

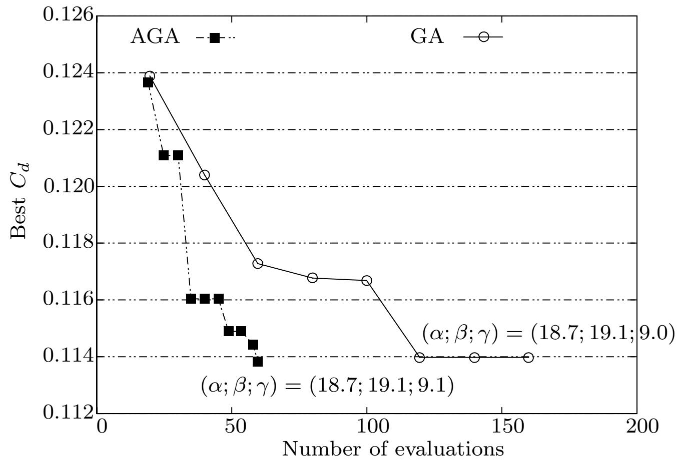

Fig. 1.10 Convergence history

putation is performed until a stationary state is observed for the main aerodynamic coefficients. This requires approximately 14 hours computational (CPU) time on a single-processor machine. In order to achieve a reasonable computational time, parallel evaluations on a cluster of workstations have been done.

(iii) Optimization method

Two different global optimization methods have been compared on this problem. The first one is a classical GA with a population number N_p equal to 20, a crossover and a mutation coefficient equaling to 0.9 and 0.6, respectively. The second method is similar to GA but with fast and approximated evaluations as presented in Sect. 1.3.3 (AGA). Note that the hybrid methods introduced in Sect. 1.3.2 have also been tested on this problem but are not presented here since AGA performs better. In contrast to these three global optimization methods, it is worth mentioning that a pure deterministic method like BFGS fails to find the global drag minimum at all (see [5] for a further comparison of all these methods).

1.4.3 Numerical Results

The convergence history of both optimization methods GA and AGA for the present drag reduction problem is depicted in Fig. 1.10. This figure shows in particular that both methods have nearly reached the same drag value,

Table 1.4 Four examples of characteristic shapes

shape	slant angle (α)	boat-tail angle (β)	ramp angle (γ)	C_d
(a)	21.1	24.1	14.0	0.1902
(b)	15.5	15.7	5.6	0.1448
(c)	16.9	17.7	11.3	0.1238
(d)	18.7	19.1	9.0	0.1140

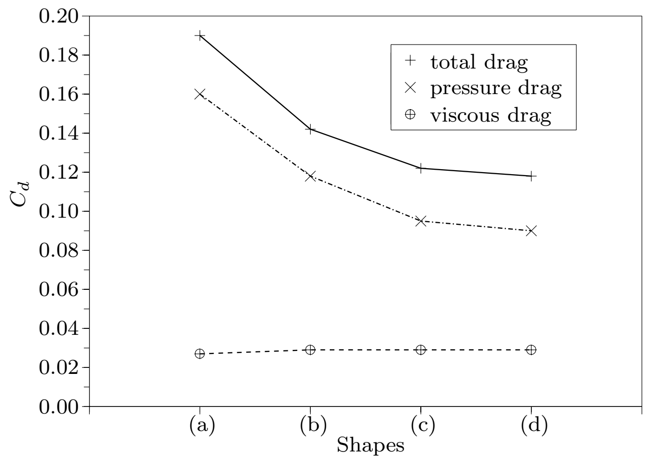

Fig. 1.11 Pressure and viscous part of drag for shapes (a) to (d)

namely 0.114, but with a different number of cost function evaluations. More precisely, the AGA algorithm has permitted to reduce the exact evaluation number by a factor of 2 compared to a classical GA, leading to the same proportional time saving.

The optimal angles obtained by both global optimization methods are nearly equal to $(\alpha, \beta, \gamma) = (18.7, 19.1, 9.0)$. These values have been experimentally validated in the Peugeot wind tunnel to be associated with the lowest drag value that can be reached with this particular forebody.

In order to understand in depth the complex phenomena involved in the variations of the drag coefficient, four characteristic shapes are presented in Table 1.4 and carefully studied.

The first shape (a) corresponds to a high-drag configuration whereas shapes (b), (c) and (d) correspond to low-drag configurations with an increasing value of slant and boat-tail angles. More precisely, shape (c) corresponds to the best shape obtained after the first generation of the GA whereas shape (d) is the best shape obtained in the whole range of admissible angles. Compared to shape (a), note that the value of the drag coefficient of shape (d) is almost divided by a factor of two which confirms the high dependence of C_d on the rear shape.

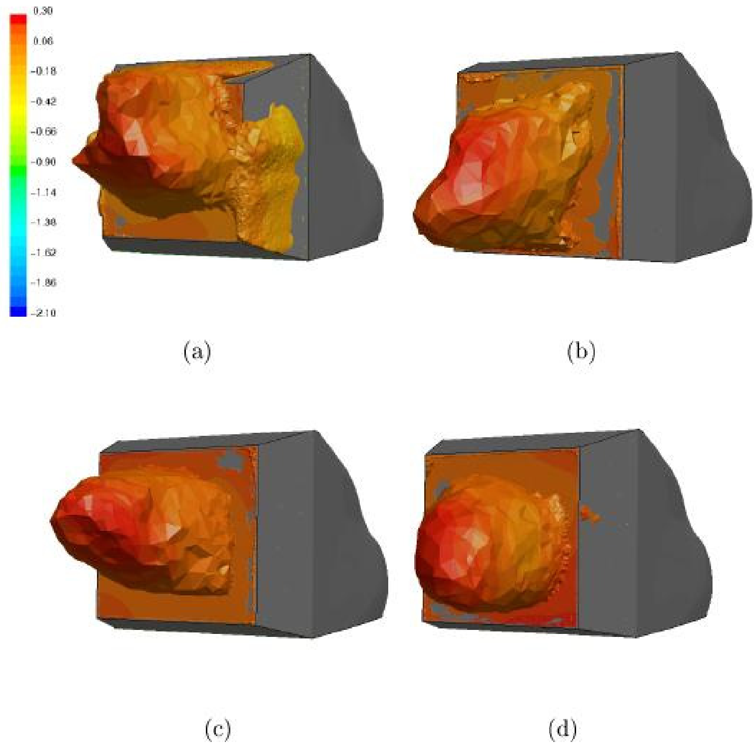

Fig. 1.12 Isosurface of null longitudinal velocity colored with pressure coefficient for shapes (a) to (d)

Figure 1.11 displays for the four chosen configurations, the relative part of the pressure drag and the viscous drag in the value of the drag coefficient C_d. It is worth noticing that the pressure drag represents the major contribution to the total drag and also that the viscous drag remains almost constant for all cases. In particular, a higher pressure at the rear of the car will automatically reduce the drag. To see more precisely the topology and the pressure values at the wake flow, Fig. 1.12 depicts for shapes (a) to (d) the isosurface of null longitudinal velocity colored with the pressure coefficient. The latter corresponds to a dimensionless pressure value and is given by:

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho V_\infty^2} \, . \tag{1.11}$$

It can be seen in particular on shapes (a) and (b) that respectively, either too high or too low slant and boat-tail angles will not generate a sufficient pressure level at the rear of the vehicle, thus increasing the pressure drag. On the contrary, intermediate and similar values of these two angles, coupled

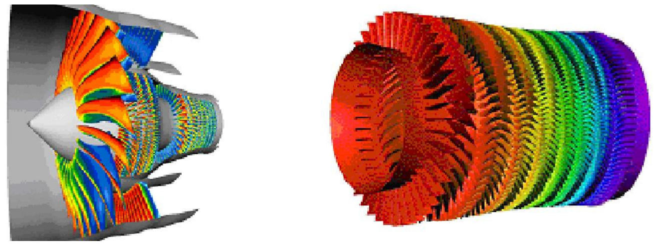

Fig. 1.13 Blades in the fan (left) and the high pressure compressor module (right)

with a low ramp angle as it is in the case for shapes (c) and (d), will improve the recompression at the car base. Note also that the optimal shape (d) is associated to a very narrow and regular recirculation bulb.

All these observations thus corroborate the qualitative trends well-known to car engineers since the experiments done by Morel [18] and Ahmed [1]. The numerical automatic tool presented here is now validated and appears to be very promising for car manufacturers to realistically design low drag car shapes in the near future.

1.5 Another Possible Application of CFD-O: Airplane Engines

In this section, another application of CFD-based optimization is given in the field of airplane engines. All the results presented here have been obtained in collaboration with two research engineers from Snecma-Moteurs (part of Safran group), B. Druez and N. Lecerf, and will be presented in more detail in [3].

1.5.1 General Description of the Optimization Case

In a turboreactor, the blades, which represent a big amount of an engine price (nearly 35%), are designed to create and control the aerodynamic flow through the engine (see Fig. 1.13).

The objective here is to optimize the design of the blades in the high pressure compressor module in order to minimize the mechanical efforts applied on them. Actually, it represents only a first step in the field of blade optimization since the main goal of a high pressure compressor designer is to increase the isentropic efficiency of the compressor. Nevertheless, this goal can not be achieved regardless of other engine features. Among them is the stall margin. This aerodynamic instability phenomenon consists in the stall of the flow

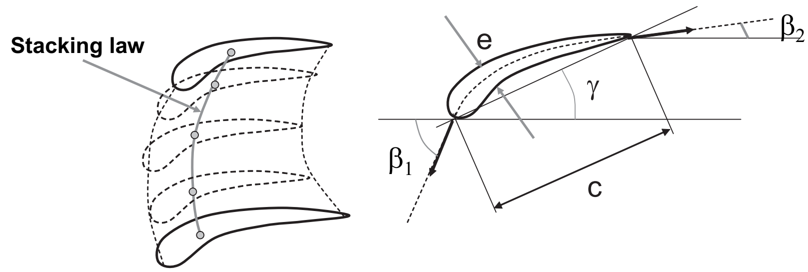

Fig. 1.14 Design parameters of a 3D blade

around the blades. This leads to backward flow inside the compressor and can result in engine shutdown, overtemperature in the low pressure turbine, high level of vibration or blade-out. To prevent such events, the designer will have to increase the compressor pressure ratio for low mass flow rates.

1.5.2 Details of the Computation

A 3D blade can be broken down into a set of several 2D airfoils profiles. The different airfoils are linked to the original blade through the stacking law (see Fig. 1.14). Each airfoil can then be described by a set of design parameters which reflect physical phenomenon that can be seized by the human designer. Figure 1.14 shows some common design parameters of the 2D profiles such as chord (c), maximum thickness value (e), upstream and downstream skeleton angles (β_1 and β_2), and stagger angle (γ). In the presented case, these parameters are kept fixed whereas the parameters to optimize, on the number of six, are all associated to the stacking law.

In order to minimize the mechanical efforts on the blade, the associated function to minimize is equal to the maximal value on 2D profiles of the von Mises constraints. Such a problem is highly non-linear, has a large number of constraints, many local minima and is also time consuming. A global, fast and robust method is thus needed.

1.5.3 Obtained Results

The AHM method presented in Sect. 1.3.2 has been used here to solve the blade optimization problem described above. Note in particular that constraints are handled with a penalization term whereas the gradients are approximated by finite differences.

Table 1.5 Convergence results on the blade optimization problem

Algorithm	number of evaluations	best obtained value
Evolutionary algorithm	460	163.5
AHM	480	158.6

The results are compared in Table 1.5 with those obtained with a classical evolutionary algorithm, here a simulated annealing. It can be seen that for the same simulation time (approximately 80 hours CPU), the AHM overperforms the simulated annealing. Indeed, even if the relative decrease obtained on the cost function appears to be small (3% approximately), it actually represents a significant improvement for the blade design.

1.6 Conclusion

In order to reduce fuel consumption, the minimization of the drag coefficient of cars has become a major research topic for car manufacturers. The development of fast and global optimization methods based either on hybridization of evolutionary algorithms with a local search process or on the use of surrogate models, has allowed only recently a first numerical and automatic approach for the drag reduction problem.

The results obtained from a simplified geometry called Ahmed bluff body are presented here. They confirm the experimental analysis saying that the drag coefficient is very sensitive to the rear geometry of the car due to the presence of separation and recirculation zones in this region, and thus can be largely reduced by shape optimization.

After this experimental validation, the numerical tool is now ready to be used by car designers for improving the drag coefficient of future car models.

References

1. Ahmed, S.R., Ramm, R., Faltin, G.: Some salient features of the time averaged ground vehicle wake. SAE Paper 840300 (1984)
2. Beyer, H.G., Schwefel, H.P.: Evolution Strategies. Kluwer Academic Publisher (2002)
3. Druez, N., Dumas, L., Lecerf, N.: Adaptive hybrid optimization of aircraft engine blades. Journal of Computational and Applied Mathematics, special issue in the honnor of Hideo Kawarada 70th birthday (in press) (2007)
4. Dumas, L., Muyl, F., Herbert, V.: Hybrid method for aerodynamic shape optimization in automotive industry. Computers and Fluids **33**, 849–858 (2004)

5. Dumas, L., Muyl, F., Herbert, V.: Comparison of global optimization methods for drag reduction in the automotive industry. In: Lecture Notes in Computer Science, vol. 3483, pp. 948–957. Springer (2005)
6. Dumas, L., Muyl, F., Herbert, V.: Optimisation de forme en aérodynamique automobile. Mécanique et Industrie **6**(3), 285–288 (2005)
7. Espinoza, F., Minsker, B., Goldberg, D.E.: A self adaptive hybrid genetic algorithm. In: Proceedings of the Genetic and Evolutionary Computation Conference GECCO 2001, pp. 75–80. Morgan Kaufmann Publishers (2001)
8. Franck, G., Nigro, N., Storti, M., d'Elía, J.: Numerical simulation of the Ahmed vehicle model near-wake. Int. J. Num. Meth. Fluids (in press) (2007)
9. Giannakoglou, K.C.: Acceleration of ga using neural networks, theoretical background. GA for optimization in aeronautics and turbomachinery. In: VKI Lecture Series (2000)
10. Gillieron, P., Chometon, F.: Modelling of stationary three dimensional separated flows around an Ahmed reference model. In: ESAIM Proceedings. Third International Workshop on Vortex Flows and Related Numerical Methods, vol. 7, pp. 173–182 (1999)
11. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
12. Han, T.: Computational analysis of three-dimensional turbulent flow around a bluff body in ground proximity. AIAA Journal **27**(9), 1213–1219 (1988)
13. Han, T., Hammond, D.C., Sagi, C.J.: Optimization of bluff body for minimum drag in ground proximity. AIAA Journal **30**(4), 882–889 (1992)
14. Jin, Y.: A comprehensive survey on fitness approximation in evolutionary computation. Soft Computing **9**, 3–12 (2005)
15. Jin, Y., Olhofer, M., Sendhoff, B.: A framework for evolutionary optimization with approximate fitness functions. IEEE Transactions on Evolutionary Computation **6**, 481–494 (2002)
16. Makowski, F.T., S.-E., K.: Advances in external-aero simulation of ground vehicles using the steady RANS equations. SAE Paper 2000-01-0484 (2000)
17. Mohammadi, B., Pironneau, O.: Analysis of the k–ε turbulence model. John Wiley & Sons (1994)
18. Morel, T.: Aerodynamic drag of bluff body shapes characteristic of hatch-back cars. SAE Paper 7802670 (1978)
19. Ong, Y.S., Nair, P.B., Keane, A.J., Wong, K.W.: Surrogate-assisted evolutionary optimization frameworks for high-fidelity engineering design problems. In: Knowledge Incorporation in Evolutionary Computation, Studies in Fuzziness and Soft Computing Series, pp. 307–332. Springer Verlag (2004)
20. Poloni, C.: Hybrid GA for multi objective aerodynamic shape optimisation, Genetic algorithms in engineering and computer science. In: G. Winter, J. Periaux, M. Galan, P. Cuesta (eds.) Genetic Algorithms in Engineering and Computer Science. John Wiley & Sons, Inc., New York, NY, USA (1995)