Applied Energy 88 (2011) 767–777

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Multi-objective optimization of internal combustion engine by means of 1D fluid-dynamic models

G. D'Errico *, T. Cerri, G. Pertusi

Dipartimento di Energia, Politecnico di Milano, Via Lambruschini 4, 20156 Milan, Italy

ARTICLE INFO

Article history:
Received 22 January 2010
Received in revised form 1 September 2010
Accepted 4 September 2010
Available online 16 October 2010

Keywords:
Multi-objective optimization
Fluid dynamic model
Internal combustion engine
GASDYN
MADS
NSGA-II

ABSTRACT

The definition of an efficient optimization methodology for internal combustion engine design using 1D fluid dynamic simulation models is presented. This work aims at discussing the fundamental numerical and fluid dynamic aspects which can lead to the definition of a best practice technique, depending on the complexity of the problem to be dealt with, on the number of design parameters, objective variables and constrains. For these reasons, both single-and multi-objective problems will be addressed, where the former are still of relevant interest (i.e. optimization of engine performances), while the latter have a much wider range of applications and are often characterized by conflicting objectives.

The Mesh Adaptive Direct Search (MADS) was chosen among the class of direct search methods and compared with the Genetic Algorithms to solve single-objective problems, and similarly two different algorithms were chosen and compared to solve multi-objective problems: the ε-constraint method and the NSGA-II (Non-Dominated Sorting Genetic Algorithm)

A single cylinder spark ignition engine, used in a motorbike application, was chosen as test case, to allow reduced computational times, without any loss of generality of the results. The analysis evaluate the convergence and efficiency of each methodology for the different problems which are solved. The achieved goal is not the definition of an ever valid mathematical strategy, but here focus is given on the parallel application of a detailed fluid dynamic analysis and automated optimization techniques to suggest a best practice technique to be employed depending on the characteristic of the optimization problem to be solved.

1. Introduction

The design and optimization of a modern internal combustion engine is a very challenging task which is characterized by synergic experimental and computational investigations. In all the different fields of applications, the tightening limitations on pollutant and carbon dioxide emissions and the increasing targets of efficiency and fuel consumption have continuously stimulated the research activity, making the design phase one of the most critical step in the engine production process. To accomplish the different and very often conflicting requirements imposed by the legislation and by the market, an important effort is carried out by scientists and researchers to define reliable optimization techniques, which involve the single component as well as all the engine system. As a matter of fact, the prototyping procedure can become sensibly expensive and time consuming if many configurations have to be tested. The aid of Computational Fluid Dynamic tools can remarkably reduce the duration and the costs of this stage. Multi-dimensional CFD models can provide useful information about the flow pattern inside combustion chambers, ducts and other devices (DPF, SCR, air filters, ...) and they are primarily used to improve the engine component design. 1D fluid dynamic models are definitely simpler and reduce significantly the computational time, due to the fact that they use a one-dimensional approach to simulate the unsteady flows and the planar wave motion inside the duct system. Despite the availability of computational resources of increasing power, currently they are still the most widely used numerical tools, since they allow a complete modeling of the entire engine configuration, giving sufficiently accurate predictions in short times and supporting the optimization and development work of any prototype.

Some of the potentialities of the combined use of an optimization tool with a 1D or 3D fluid dynamic code have been previously shown in literature [1–4], while different examples of the most common employed optimization techniques in the study of internal combustion engines are given in [5–8]. The motivation of this paper is the definition of an efficient optimization methodology for internal combustion engine design using 1D fluid dynamic

* Corresponding author. Tel.: +39 0223998636; fax: +39 0223998050.
E-mail address: gianluca.derrico@polimi.it (G. D'Errico).

0306-2619/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.apenergy.2010.09.001

simulation models. To these ends, it discusses the fundamental numerical and fluid dynamic aspects which can lead to the achievement of a best practice technique, depending on the complexity of the problem to be dealt with, on the number of design parameters, objective variables and constrains. Both single-objective and multi-objective problems will be addressed, where the former are still of relevant interest (i.e. optimization of engine performances), while the latter have a much wider range of applications and are often characterized by conflicting objectives.

In the next sections the authors will briefly review the fundamental equations governing the unsteady fluid dynamic motion in the duct system of an internal combustion engine and the chosen optimization techniques. Then different test cases will be presented, considering both single-and multi-objective problems, which typically occur in the application of thermo-fluid dynamic codes for the design and optimization of modern internal combustion engines.

2. Fluid dynamic model

The 1D fluid dynamic code adopted for the engine simulations is a research code developed by the authors, GASDYN [9–12]. It has been recently coupled with a new toolkit, named OptimICE, which includes all the aforementioned optimization methodologies and specific functionalities to exchange all the required information.

The analysis is based on the solution of a set of 1D conservation equations for one-dimensional, unsteady, compressible flows in ducts with variable cross-section written in a conservative form as follows:

$$\frac{\partial(\rho F)}{\partial t} + \frac{\partial(\rho u F)}{\partial x} = 0 \tag{1}$$

$$\frac{\partial(\rho u F)}{\partial t} + \frac{\partial(\rho u^2 F + pF)}{\partial x} = p\frac{\partial F}{\partial x} - \frac{2}{D}\rho u|u| f_w F \tag{2}$$

$$\frac{\partial(\rho e_0 F)}{\partial t} + \frac{\partial(\rho e_0 u F + upF)}{\partial x} = \rho q_{re} F \tag{3}$$

where ρ is the gas density, p is the gas pressure, u is the gas velocity, e_0 is the specific total energy, f_w is the gas-wall friction coefficient, F is the pipe cross-section, D is the duct diameter, q is the heat transferred between gas and duct walls per unit of mass. The closure equation of the system in this case is the perfect gas equation of state. The system of equations is solved using shock-capturing numerical methods based on finite difference techniques. The numerical scheme adopted for the calculations is the second order Lax–Wendroff method, used with the addition of TVD flux limiting techniques based on the Davis symmetric limiter. The treatment of the boundary conditions is based on a characteristics approach, which has been used to evaluate the interaction of waves, in terms of incident and reflected Riemann variables.

The in-cylinder thermodynamic processes are described by a quasi-D approach, which applies the conservation of mass and energy over the cylinder volume, modeled as a single open system [10].

3. Optimization techniques

An optimization process typically entails the employment of a three-step methodology: a Design of Experiments method to explore the design space, a response surface to estimate a possible optimal point and finer optimization algorithm to determine the best solution. The first two steps are not always necessary and the opportunity to perform them needs to be carefully assessed, depending on the complexity of the problem, the number of design parameters and the time required for each function evaluation. At the same time, in many engineering problems the use of a finer optimization algorithm to determine the best solution might not be necessary, due to the capability of the DoE techniques to efficiently explore the design iper-space. For what concerns the employment of fine optimization techniques, three relevant classes of solution approaches can be addressed, namely [13–15]: relaxation methods outer approximation, generalized benders decomposition, branch and bound, extended cutting plane method), search heuristics (simulated annealing, Tabu search, evolutionary algorithms), and generalized pattern search methods (Hooke and Jeeves, the simplex algorithm of Nelder and Mead, Lewis and Torczon, MADS). Classical gradient-based or Newton-based methods are generally not suitable for internal combustion engine optimization problems and, if applied, their strong dependence on the initial guess, might cause them to find local optima rather than global ones.

In a multiobjective optimization problem (MOP) different objectives must be optimized simultaneously, with respect to some constraints. Mathematically, the problem can be written as follows:

$$\text{minimize } \mathbf{F}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix} \tag{4}$$

$$\text{with } \begin{cases} h_i(\mathbf{x}) = 0 & i = 1,2,\ldots,n_{eq} \\ g_j(\mathbf{x}) \leqslant 0 & j = 1,2,\ldots,n_{ineq} \\ x_{k,min} \leqslant x_k \leqslant x_{k,max} & k = 1,2,\ldots,m \end{cases} \tag{5}$$

In this equation, $\mathbf{x}$ represents the set of variables to be optimized, $\mathbf{F}(\mathbf{x})$ is the vector of objectives, $h_i(\mathbf{x})$ and $g_j(\mathbf{x})$ are the equality and inequality constraints, $x_{k,min}$ and $x_{k,max}$ are the range to which the variables belong. In a typical MOP, there are different solutions, and they are called *nondominated* or *Pareto-optimal* solutions. A set of variables $\mathbf{x}_1$ *dominates* another set $\mathbf{x}_2$ (mathematically: $\mathbf{x}_1 \prec \mathbf{x}_2$) if $f_i(\mathbf{x}_1) \leqslant f_i(\mathbf{x}_2)$ for each objective function and $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2)$ for one objective function at least. The solutions which do not dominate each other are called *nondominated* solutions and they are the solutions of the MOP.

There are different algorithms designed to solve MOP problems, but all of them belong to two families: the first one solves a single objective problem for each Pareto-optimal solution, while the second one searches for all the nondominated solutions simultaneously. Examples of the first family are the *weighted sum method* [16], the *ε-constraint method* [17], the *normal boundary intersection method* [18,19]. The second family is represented by evolutionary algorithms, in particular Genetic Algorithms: examples are the *Vector Evaluated Genetic Algorithm* (VEGA) proposed by Schaffer [20], the *Fonseca and Fleming's Genetic Algorithm* (FFGA) [21], the *Non-dominated Sorting Genetic Algorithm* (NSGA) by Srinivas and Deb [22] and its improved version, the NSGA-II [23], the *Strength Pareto Evolutionary Algorithm* (SPEA) by Zitzler and Thiele [24], the *Pareto Archived Evolution Strategy* (PAES) by Knowles and Corne [25] and the *Niched-Pareto Genetic Algorithm* (NPGA) [26] (a survey of these algorithms can be found in [27]). Particle swarm algorithms are also used (a survey is proposed in [28]).

The authors decided to implement two algorithms, one for each family. The ε-constraint method [17] was chosen from the first family, because of its simplicity and of its performances on the analyzed problems. The ε-constraint method [17] creates a single-objective subproblem for each nondominated solution we want to find. The Mesh Adaptive Direct Search (MADS) algorithm [29,30] was chosen among the class of direct search methods to solve single-objective problem. For the second family, NSGA-II was adopted because it grants elitism and the diversity preservation is automatically handled by the algorithm [23]. More details about the chosen optimization techniques, their implementation and application to the considered test problems are given in Appendix A.

4. Results

In this section several test problems will be presented, in order to evaluate the properties of the implemented optimization methodologies.

4.1. Single objective problems

The most important single objective problem in the design of an engine is the achievement of the best performances at full load. In fact, at these conditions, other fundamental objectives such as fuel consumption and emitted emissions play a less important role, while they will significantly characterize the determination of the engine map at partial load conditions, which include all the legislative driving cycle and most of the effective use of the engine during its lifetime. The variables which affect most the achievement of optimal performances in a spark ignition engine are the inlet and exhaust manifold lengths, the inlet and exhaust valve timing and lift profile and the ignition timing. To exploit the ability of a simulation-based optimization framework to find the optimal set-point combination of variables, different engine configurations were considered, with an increasing number of parameters to be optimized. The first test was run considering a single cylinder spark ignition engine, whose model was built to be as close as possible to an ideal engine. Therefore, a 180° rectangular valve lift was imposed, and the duct system was made of a single duct for the intake and the exhaust, along which friction and heat transfer were neglected. Despite the simplicity of such a configuration, the determination of the optimal inlet and exhaust duct lengths is not that trivial, due to the highly unsteady nature of the induction and exhaust processes. In Fig. 1 the volumetric efficiency, computed as function of the inlet duct length, ranging from 0.1 m to 1.2 m, at 3000 rpm is plotted in black line. From a mathematical point of view, the optimization tool, to be successful, needs to find the absolute optimal value, which in this case corresponds to a length equal to 0.76 m and a volumetric efficiency of 124%. Nevertheless the curve profile is characterized by very close local maximums and is very oscillating. The simplicity and low computing demand of this problem would allow to use also a very expensive DoE technique, such as a full factorial with numerous levels. However this solution would not be suitable to more complex problems with multiple variables, therefore a D-optimal method was used, constructing a quadratic response surface (i.e. a line for one variable) on the basis of the tested DoE points. Then, the estimated optimal point was used to start the chosen optimization technique. Hence the DoE points, the response line and the results of the different points tested by the MADS method and by the Genetic Algorithms are also shown in Fig. 1. Both methods locate correctly the absolute optimal configuration, requiring respectively 11 and 14 runs. A better understanding of the observed dependence of the volumetric efficiency on the inlet duct lengths, on a fluid dynamic point of view, can be achieved by analyzing the computed pressure pulses. Fig. 2 shows these trends in the intake duct upstream of the inlet valve for two significant lengths, 0.76 m and 0.85, which correspond to very different values of the volumetric efficiencies. In this figure, the induction period is outlined by the shadow zone (360° 540°). We can observe that the longer duct causes a higher pressure at the inlet valve opening (IVO), which reduces the intensity of the rarefaction wave, starting at IVO on the inlet valve sections and propagating upstream along the inlet duct. The intensity of the positive reflected wave at the open-end is lower with the 0.85 m duct than with the 0.76 m duct. Hence, the resulting pressure is definitely higher for the shorter duct during the second part of the induction period (from about 450° to IVC), whose influence on the in-cylinder trapped mass is very high.

This test was then repeated varying the engine speed from 2000 rpm to 8000 rpm with a step of a 1000 rpm. Both optimization methods achieve the same results apart from one engine speed at which two different inlet lengths are found but they produce an identical volumetric efficiency. The MADS method required 48 runs, while the GA needed 78 runs. In Fig. 3 these results are shown together with the corresponding volumetric efficiency.

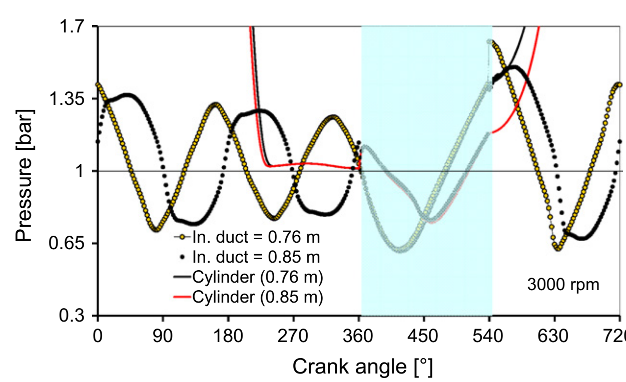

Fig. 2. Computed pressure pulses in the intake duct upstream the cylinder valve of the simulated ideal single cylinder engine at 3000 rpm and full load.

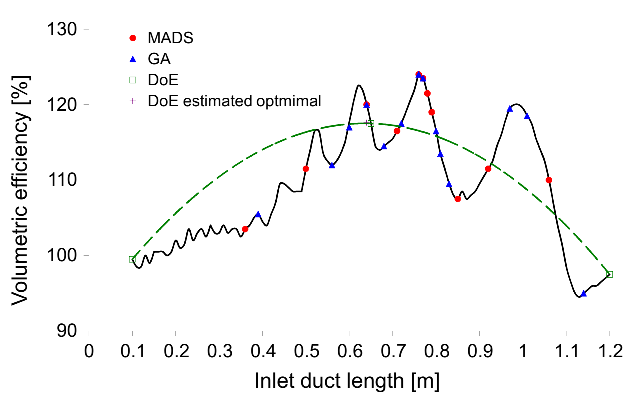

Fig. 1. Volumetric efficiency, computed as function of the inlet duct length, ranging from 0.1 m to 1.2 m, at 3000 rpm.

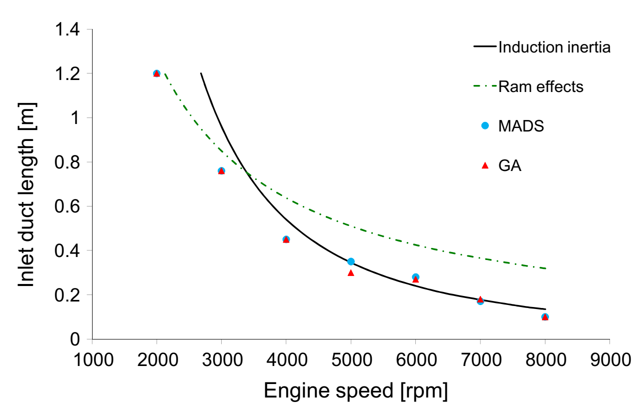

Fig. 3. Estimation of the optimal lengths on the basis of the induction inertia and ram effect, MADS method and by the GA.

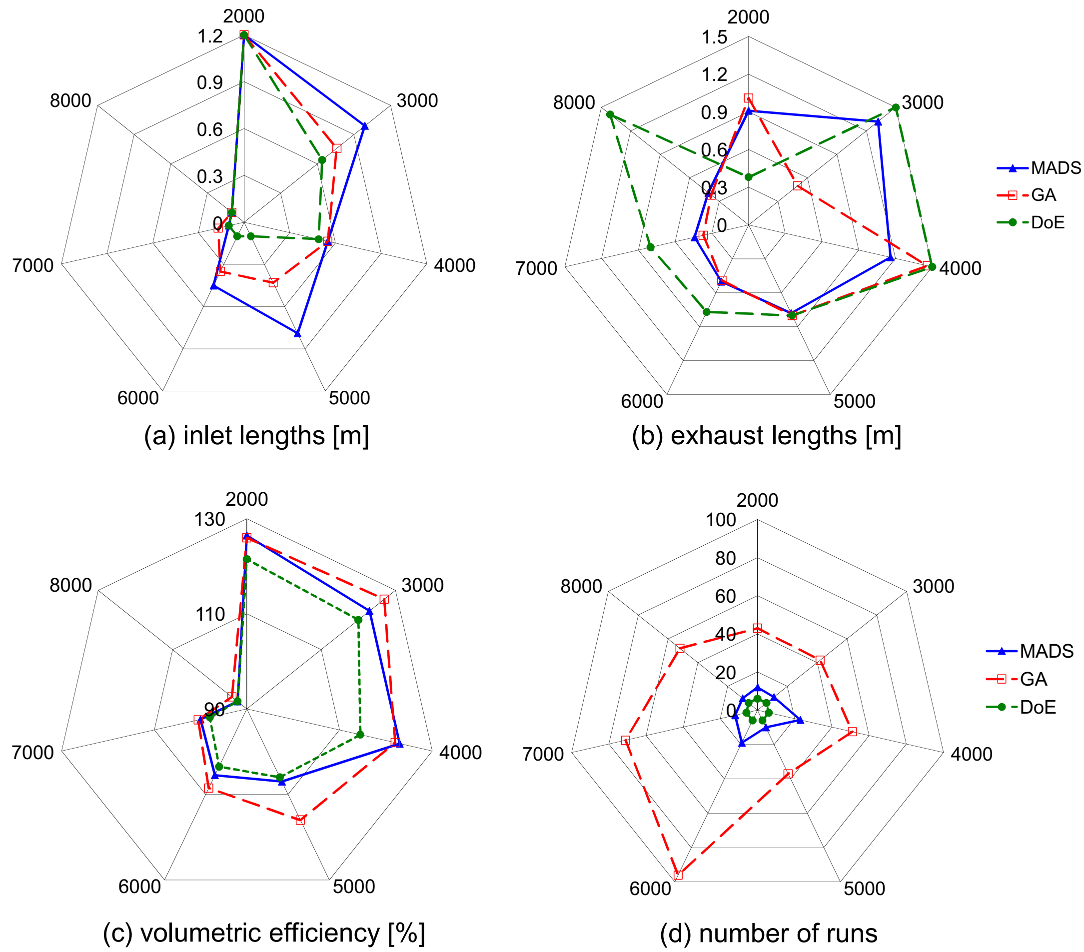

Fig. 4. (a and b) Optimal values of the inlet and exhaust ducts as function of the engine speed, (c) maximum values of volumetric efficiency determined by the two different optimization methods (MADS and GA) and DoE, (d) number of runs required by the two different optimization methods (MADS and GA) and DoE.

It is worth to underline that for all conditions the optimization techniques found the absolute optimal solution. Additionally, the simplicity of this system allows an estimation of the optimal lengths on the basis of the induction inertia and ram effect theory [32], whose corresponding optimal values are also reported in Fig. 3. This comparison helps a better understanding of the distinct phenomena which are responsible for the features of the volumetric efficiency curve, indicating the wave effects as dominating after 4000 rpm, while, at lower speed, the best configuration exploits the extra charge effects, which are due to the inertia and elasticity of the charge in the inlet duct and in the cylinder. Moreover, because of the simplicity and hence reduced computation time required by each run, it was possible to verify convergence to the best performance determined by the 1D fluid dynamic code at each engine speed.

If we increase the number of variables, we could choose the inlet and exhaust ducts as variables to be optimized. Fig. 4a and b shows the optimal values as function of the engine speed. Also in this case, the results obtained by the MADS and GA are quite similar, whose maximum values are given in Fig. 4c. Overall the best results are achieved by the Genetic Algorithm. On the other hand, it requires a significant higher number of runs than the MADS method, as shown in Fig. 4d (i.e. 97 runs for the MADS and 392 for the GA). It is clear that this difference tends to increase exponentially with the number of variables to be optimized, and this aspect significantly limits the possibilities of applying the Genetic Algorithms for more complex configurations for similar single objective problems. Moreover, it can be remarked that the obtained optimal values for the inlet lengths do not differ much from the ones obtained in the previous case, suggesting a fundamental independent behavior of the two variables. However, this test case concerns an ideal single cylinder engine with no valve overlap, therefore this conclusion is not surprising. The dashed green[1] line with closed circles in Fig. 4a–d shows the results obtained after the DoE space screening (D-optimal method), where the optimal solutions were determined on the basis of the constructed surface response. Obviously these results are less good than the ones obtained after the finer optimization techniques. Nevertheless the reduced computational cost (six runs for each operating point) suggests to consider the idea of performing only a simple DoE as a viable possibility. Additionally, the problem so far is purely theoretical and the range of variability of the two design parameters is wider than it would be in a real problem. Hence, a real engine configuration was investigated, considering a single cylinder spark ignition engine for motorbike application, to allow a fast computational time, without any loss of generality of the results. The main specifications of this engine are reported in Table 1.

The intake and exhaust systems are rather simplified to reduce the total weight and dimensions. The requirement of keeping low production costs does not allow to make use of expensive technical devices and the designer is asked to achieve good performances in terms of drivability and maximum power with a fixed duct and

[1] For interpretation of color in Fig. 4, the reader is referred to the web version of this article.

Table 1
Main engine specifications.

Engine type	Spark ignition
Number of cylinder	1
Total displacement	573
Bore (mm)	98
Stroke (mm)	76
Compression ratio	11:1
Air metering	Naturally aspirated
Injection system	PFI

valve timing configuration. The engine has a primary runner, which is connected to the ambient by the air-box. The used cam profile is suitably contoured to provide a smooth rise and fall for the predicted motion of the inlet and exhaust valves, which remain open respectively for 306 and 335 crank angles degrees.

The first step consisted in building an accurate model of the entire engine system in its baseline configuration, giving as input all geometrical and operating information. In Fig. 5 the comparison between the experimental and the computed brake torque curves at full load is shown (labeled as measured and computed). The baseline configuration consists of a total inlet length equal to 0.22 m, mainly because of the strong constraints imposed by the engine application on a motorbike, which requires compact dimensions and minimum weights for the used components. Similarly, the length of the exhaust system is strongly affected by the upper limits.

Again here, the engine performances are chosen as objective function. Typically, a 1D model of an engine consists of a series of numerous elements, where each 1D duct is defined by its inlet and outlet diameters, length and curvature. For a mathematical point of view, each of these dimensions should be considered as an independent variable. Obviously, this would include unrealistic configurations defined by a series of tapered pipes, separated by sudden diameter variations. Even imposing some constraints on the diameter continuity, if we treat each single piece of duct as a variable, the dimensions of the problem would become unacceptable. For these reasons, the optimization tool developed in this work, OptimICE, required a significant customization to be applied for optimization problems of intake and exhaust engine system configurations. Specifically, the possibility of grouping multiple variables (diameters and/or lengths) was introduced. This significantly reduces the number of geometric variables to be optimized, so that each group of ducts can be scaled with respect to its radial (diameter) or axial (total length) dimension. Additionally, in a multi-cylinder engine configuration, the designer might be interested in considering, for example, all the primary ducts as a unique variable to be varied within certain constraints. In the considered single cylinder engine layout, the variables to be optimized correspond to the total length of the primary duct and its radial dimension. This approach causes to change the angles of the tapered pipes, but, considering the limited range of geometric variations allowed in a real engine configuration, it is probably the most convenient and efficient strategy, at least for a preliminary stage of design and optimization. Similarly, the mean exhaust diameter and its length were considered for optimization. The head duct geometry, as the cylinder head and valve seats, was not varied. The MADS and Genetic Algorithm procedures, together with a DoE analysis, were applied to determine the optimal solution for the engine torque. The two mentioned duct radial dimensions and lengths and the inlet and exhaust valve timings were chosen as input variables. Table 2 shows the range of variability of each variable; all data are given relative to the baseline configuration and the range of variability was chosen according to the existing geometrical constraints.

Fig. 6 shows very similar results obtained by the GA (1040 runs) and MADS (138 runs) methods, while the optimal points, determined on the basis of the quadratic response surface built after the DoE screening, produce poor results. This is due to the high number of simultaneous input variables which were considered. In fact the DoE indications could have been more successful if a step-by-step optimization procedure had been adopted (duct lengths, duct diameters and valve timing treated separately). However, the results shown in Fig. 6 are meaningful only for a mathematical point of view. In fact it is the task of the designer to analyze the contribution of each variable and choose the feasible optimal configuration. The authors propose as possible optimal choice to increase the inlet and exhaust duct length respectively of 0.04 m and 0.2 m and to increase both radial dimensions of 1 mm, while for what concerns the valve timing, a two-step variable valve timing could be introduced, choosing as optimal solution an IVO value of 50° BTDC up to 5000 rpm and 40° BTDC from 5500 rpm and an

Table 2
Range of variability for the chosen input data of the single objective test case.

Variable	Range
Inlet duct length (m)	−0.02/+0.11
Exhaust duct length (m)	−0.06/+0.21
Inlet radial dimension (m)	−0.003/+0.007
Exhaust radial dimension (m)	−0.003/+0.002
IVO (°BTDC)	58/31
EVO (°BBDC)	107/116

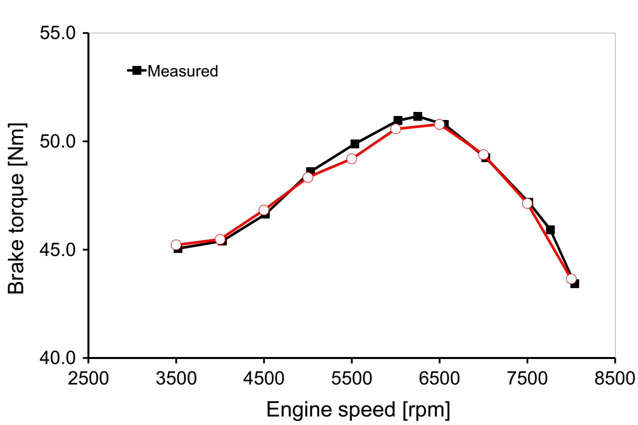

Fig. 5. Comparison between experimental and computed brake torque for the baseline model of the single cylinder engine test case.

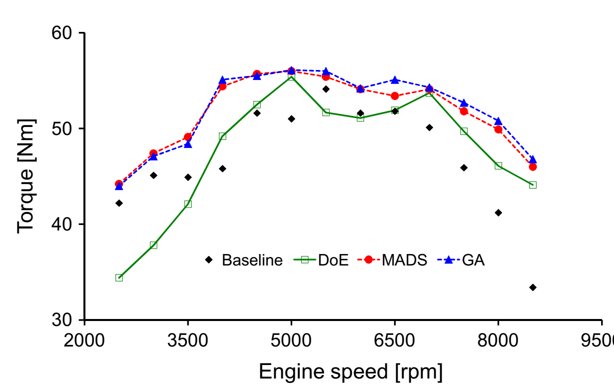

Fig. 6. Baseline and optimal brake torque computed with the different tested algorithms.

EVO value of 107 °BBDC.n The resulting brake torque curve is plotted in dashed line in Fig. 5 to show the predicted increase with respect to the baseline configuration.

We can conclude that both MADS and GA methods were successful, but the Genetic Algorithms always required a higher number of runs than MADS to determine the optimal solution. This is due to the fact that in a real configuration the variability of the geometrical and operating parameters which govern the fluid dynamic behaviour of the engine is rather limited, therefore the employment of evolutionary algorithms as fine optimization techniques is not the best choice. This analysis has been proposed as an example of a performance optimization of a real engine configuration. In a previous work [12], the authors have been investigating more extensively this type of problem by considering also multi-cylinder configurations, confirming the same conclusions which were drawn in this case. Hence it becomes matter of interest to apply the discussed methodologies for the optimization of multi-objective problems whose demands and characteristics might result rather different from the ones considered so far.

4.2. Simple multi-objective problem

A very common multi-objective problem occurring in the definition of the operation of an internal combustion engine is the trade-off between the NO_x emissions and the brake specific fuel consumption (BSFC). Therefore such a case was chosen as first multi-objective test case. In fact, while the most common single-objective problem in the design of an engine, regards the optmization of its performances at full load conditions and is mainly a fluid dynamic problem, multi-objective problems play a major role for all other operating conditions, where the reduction of the pollutant emissions and fuel consuptions are fundamental prerequisites. The variables chosen for this test case are the inlet valve opening (IVO), the exhaust valve opening (EVO) and the throttle angle. The complete details of the test problem are specified in Table 3.

To evaluate the convergence of the algorithms, a map of the objective space was created, with a relative coarse grid (two crank angle degrees for the valve timing and 2° for the throttle valve), which was possible to build because the time required by each run is very low due to the simplicity of the chosen test case. The mapped objective space and the Pareto-solutions are presented in Fig. 7, while the corresponding optimal configurations are shown in Fig. 8, which basically include the complete range of loads (throttle positions) and suggest erly inlet valve opening and late exhaust valve opening.

Before undertaking the analysis, it is important to briefly introduce the employed methodology. The optimization process creates an archive of all the tested configurations and returns a series of points which are the Pareto solutions: these configurations will be called *algorithm solutions*.

In addition, a control on all the tested solutions is performed and the nondominated ones are pointed out: these solutions will

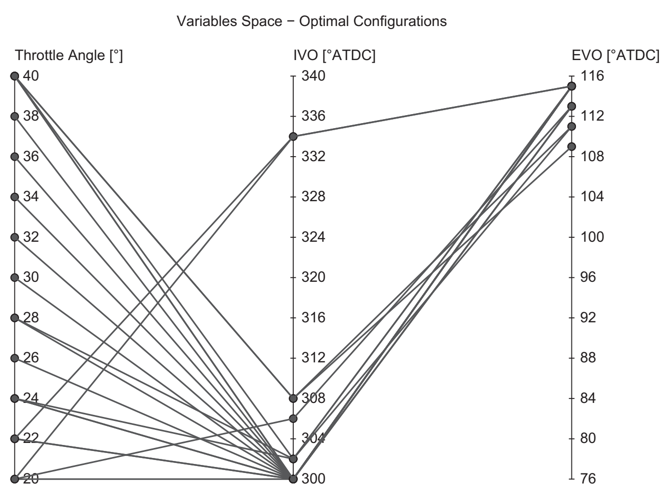

Fig. 8. Optimal configurations (throttle angle and valve timing) for the test problem.

Table 3
Details of the test problem.

Objectives	Max or Min		Accuracy
BSFC (g/kW h)	Min		1
NO_x (ppm)	Min		20
Variables	Min	Max	Accuracy
IVO (°)	300	340	0.5–2
EVO (°)	78	115	0.5–2
Throttle angle (°)	20	40	0.5–2
Configurations	249,075–4620		
Regime	4000 rpm		

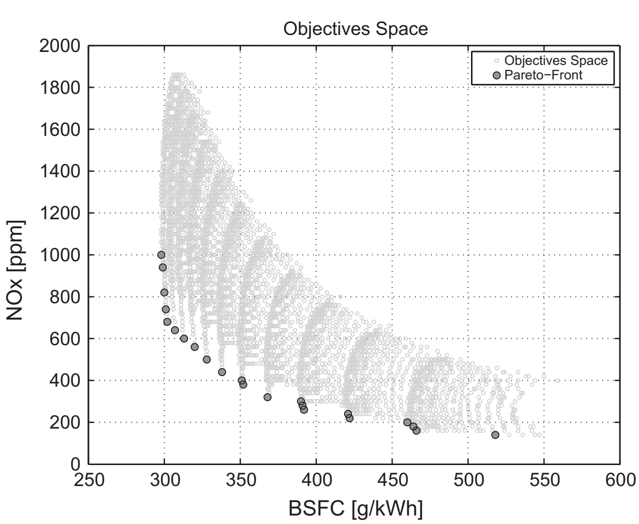

Fig. 7. Objective space and NO_x-specific fuel consumption Pareto-frontier of the test problem.

Table 4
Settings adopted in the optimization problems.

Algorithm	Parameters	Accuracy		
		2	1	0.5
ε-Constraints	No. of problems	30	30	30
NSGA-II	Population size	14	20	26
	No. of generations	15	15	15

Table 5
Number of tests executed in the optimization problems.

No. of tests		No. of tests		
Algorithm	Accuracy	Total	Real	Cache
ε-Constraint	2	1.202	243	959
NSGA-II	2	268	185	83
ε-Constraint	1	1.340	329	1.011
NSGA-II	1	364	265	99
ε-Constraint	0.5	1.372	417	955
NSGA-II	0.5	469	386	83

be called *archive solutions*. This procedure requires a very small amount of additional time, particularly if compared to the time required by the fluid dynamic code for a run. The control over the tested solutions is very useful because of the following reasons:

- The ε-constraint algorithm searches for a single pareto-solution at a time, and in this way it is possible that, during the research of one of the following solutions, a point which dominates a previous solution is found.

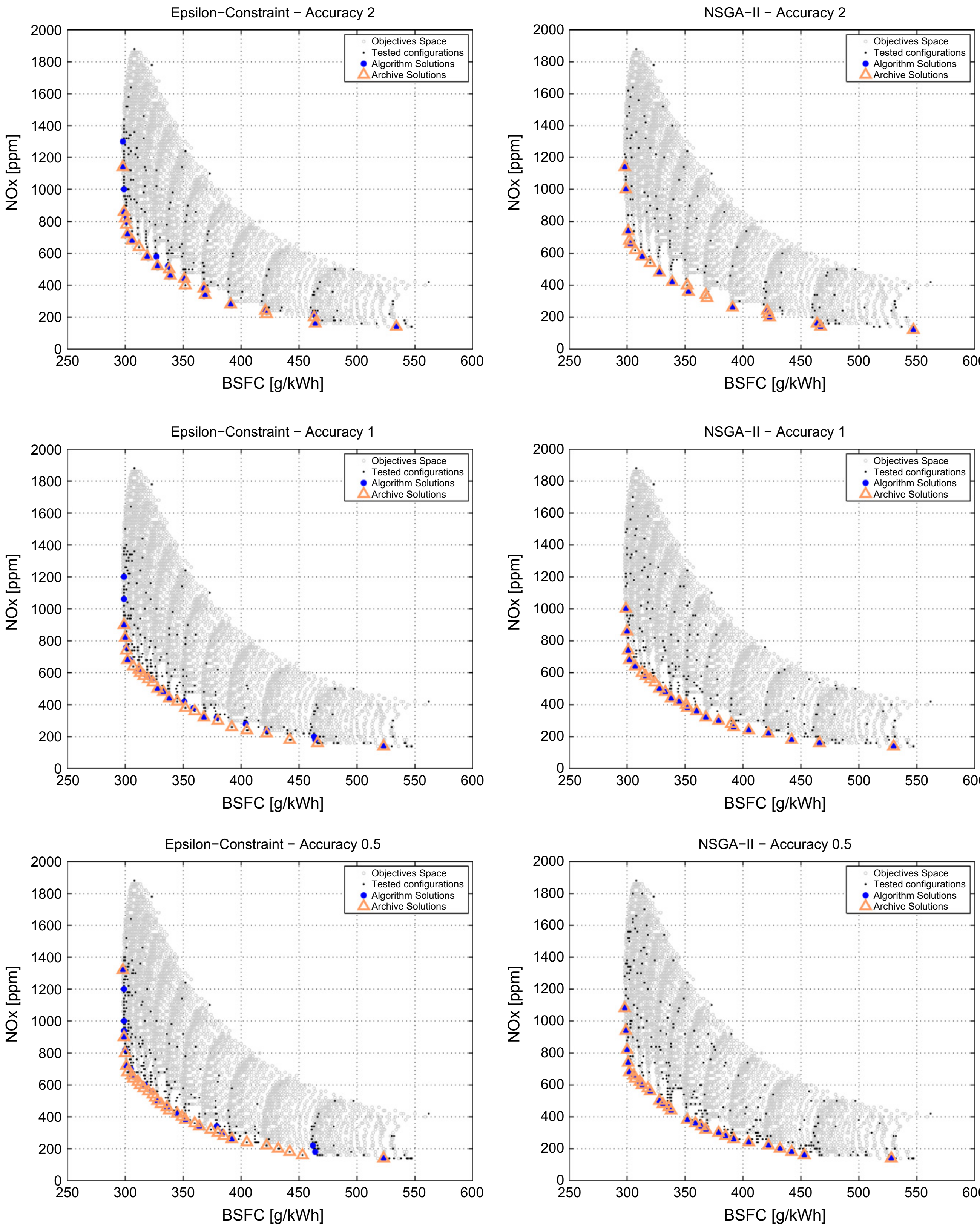

Fig. 9. Algorithm solutions and archive solutions found by the ε-constraint and the NSGA-II algorithms with different variable accuracy (2–1–0.5) for the NO_x-specific fuel consumption optimization problem.

- The NSGA-II algorithm returns a number of solutions equal to the population size, but during the optimization process a greater number of good points may have been tested: in this way it is possible for the user to recognize these solutions.

The capabilities of the two algorithms and of their coupling with the 1D fluid dynamic code were assessed, using three different values for the accuracy of the considered variables. The analyzed number of problems for the ε-contraint algorithm, the chosen population size and number of generations for the NSGA-II are shown in Table 4. It is clear that the total number of possible configurations varies greatly with the selected accuracy, whose labels (0.5, 1, 2) correspond to the range of values which is indicated in Table 3. The results of the performed analysis in terms of required runs are shown in Table 5, in which the real and cache tests are expressly indicated. A cache test consists in a set of variables which was already simulated, whose call is still time consuming since it requires an exchange of information between the optimization method and the stored values. Hence, when a cache test is executed, the fluid dynamic code is not called by the optimization method. A real test is instead a case with a new combination of variables, which will be processed as input by the fluid dynamic code. The sum of real and cache test is here referred as total. In Fig. 9 the algorithm solutions and archive solutions found by the two different methods are shown for all the different set of accuracies. Finally, for the sake of synthesis, only for the lower accuracy cases, in Fig. 10 and in Fig. 11 the algorithm solutions, for the ε-constraint and NSGA-II algorithms respectively, are proposed in the parallel charts.

The analysis of the shown results outlines two main limits of the ε-constraint algorithm with respect to the NSGA-II method:

- it sometimes fails in the search of a Pareto-solution, in fact some of the algorithm solutions are dominated by the archive solutions;
- even if the ε-constraint searches for 30 Pareto-solutions, only a smaller number is found each time, because many subproblems lead to the same solution.

The first problem is due to the adopted single-objective algorithm (MADS), which may have some difficulties in finding a solution in constrained problems. The NSGA-II does not have this kind of problem, in fact all the algorithm solutions are also archive solutions (and it means that they are not dominated by any configuration tested during the optimization process). Moreover, the NSGA-II has a small amount of repeated solutions, thanks to the crowding distance parameter used to preserve diversity. These considerations prove that the Genetic Algorithm is the best choice to solve multiobjective optimization problems and this is also confirmed by the number of required tests (see Table 5): the NSGA-II obtains better results and requires a smaller number of simulations. On the other hand, the Genetic Algorithm has no stop-criteria of its own: the optimization process ends when the number of generations set by the user is analyzed. In this study, the population size and the number of generations were set to obtain a number of simulations similar to the one performed by the ε-constraint algorithm. For this reason the number of tests required by the two algorithms are very similar, but it is important to note that the NSGA-II is also capable of identifying the Pareto-solutions with a smaller amount of simulations.

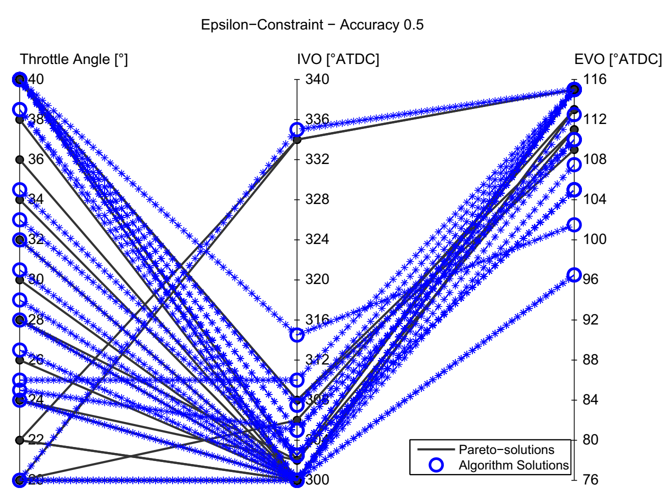

Fig. 10. Configurations corresponding to the algorithm solutions found by the ε-constraint algorithm (accuracy = 0.5).

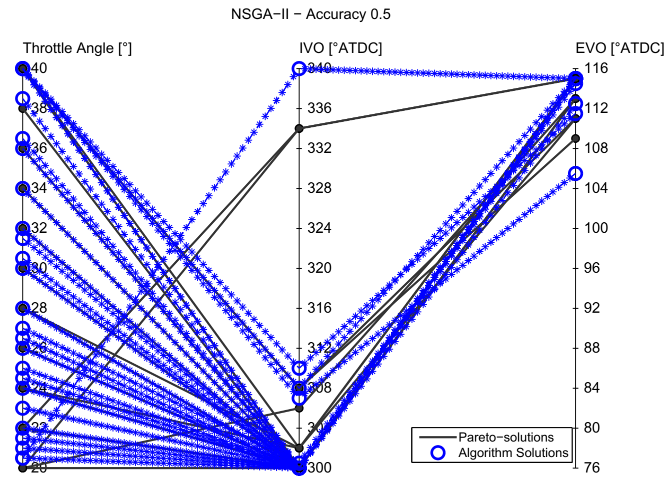

Fig. 11. Configurations corresponding to the algorithm solutions found by the NSGA-II algorithm (accuracy = 0.5).

In conclusion, it can be stated that the NSGA-II algorithm is able to identify the Pareto-solutions and performed better than the ε-constraint algorithm. The ε-constraint algorithm has proven not to be fully reliable for this kind of application: even if the archive solutions found in this analysis are on the Pareto-frontier, if no map of the objective space is available it is impossible to know a priori if the archive solutions lie on the Pareto-frontier. In fact the archive solutions are conceived to supply an additional information, but the only reliable solutions are the algorithm solutions.

4.3. Complex multi-objective problem

If we add one variable (spark advance) and one objective (BSFC), the dimensions of the problem increase significantly, in terms of possible configurations to test and solutions to analyze. On the basis of the previous conclusions, only the NSGA-II algorithm will be applied to this new test case, which considers four variables (IVO, EVO, throttle angle, spark advance) and three objectives (torque, BSFC and NO_x). All the details are given in Table 6. Because of the added dimension, the objective space will be represented as a bubble chart, in which torque and NO_x are shown on the axis, while the brake specific fuel consumption (BSFC) is given by the color of the bubble.

For this problem a detailed analysis was made with the NSGA-II algorithm, setting high values both for the population size and for the number of generations. Because of the high number of performed simulations, the algorithm solutions of this analysis can be taken as a good approximation of the exact Pareto-frontier

Table 6
Details of the complex problem.

Objectives	Max or Min		Accuracy
BSFC (g/kW h)	Min		1
NO_x (ppm)	Min		20
Torque (Nm)	Max		1
Variables	Min	Max	Accuracy
IVO (°)	300	340	1
EVO (°)	78	115	1
Throttle angle (°)	20	40	1
Spark adv. (°)	−36	−20	1
Configurations	556.206		
Analyzed regime	4000 rpm		

(see Figs. 12 and 13). The comparison of this analysis with the previous one shows a wider range optimal exhaust valve opening timings, which are due to the introduction of one additional objective (torque) and one variable (spark timing). Hence the correlations among the chosen variable increase, since the objective results depend on the trapped residual gases and wave motion along the duct systems. Then a second analysis with much coarser parameters was launched, in order to verify the convergence and the efficiency of the algorithm. The chosen settings in the optimization problems are given in Tables 7.

This analysis, whose number of executed tests is given in Tables 8, shows that the genetic algorithm is capable of achieving good re-

Table 7
Settings adopted in the optimization problems.

Parameters	Analysis	
	Detailed	Coarse
Population size	50	40
No. of generations	100	30

Table 8
Number of tests executed in the optimization problems.

NSGA-II analysis	No. of tests		
	Total	Real	Cache
Detailed	5.175	2.916	2.259
Rough	1.365	1.040	325

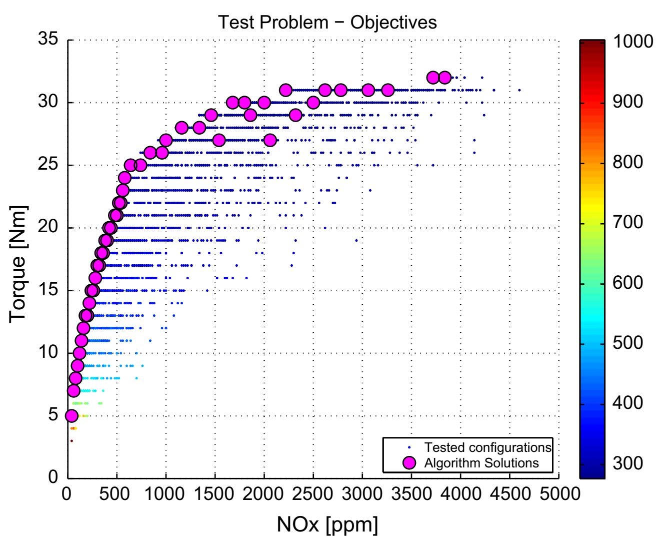

Fig. 12. Tested configurations and Pareto-solutions in the detailed analysis.

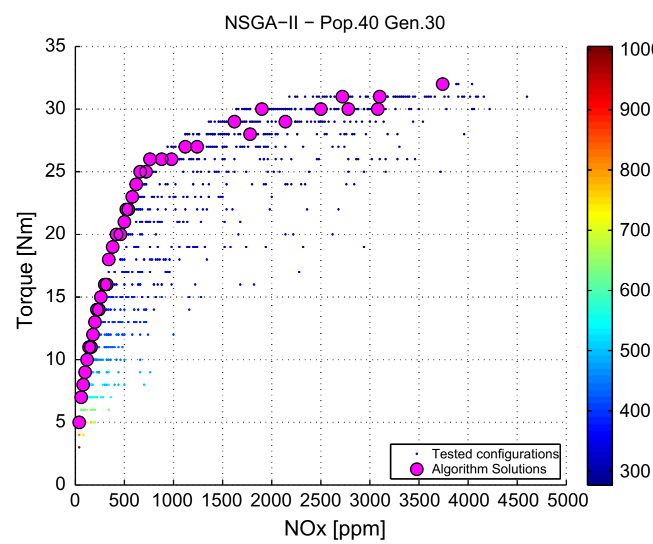

Fig. 14. Algorithm solutions found by the NSGA-II algorithm.

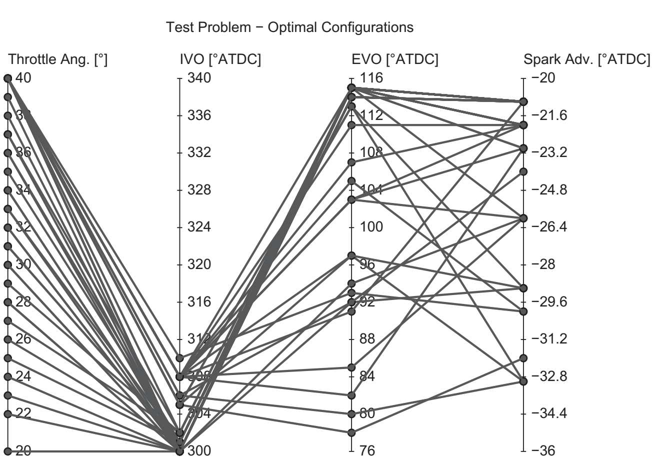

Fig. 13. Optimal configurations of the detailed analysis.

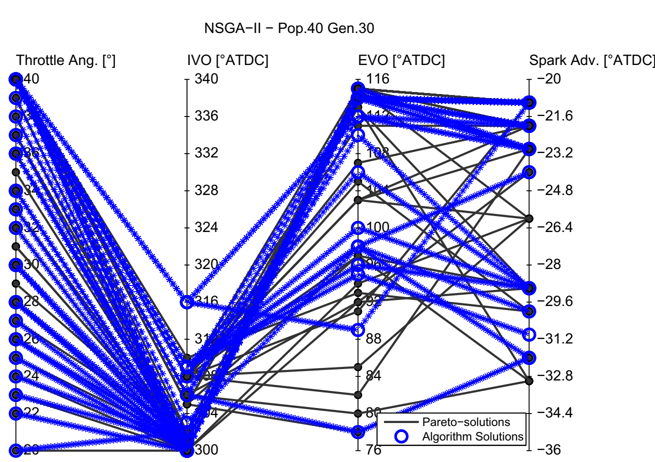

Fig. 15. Configurations corresponding to the algorithm solutions found by the NSGA-II algorithm.

sults even with a small number of simulations.. In fact, using coarse parameters, only 0.245% of the total number of configurations was tested (1.365 out of 556.206).

Figs. 14 and 15 show that most of the objective functions and corresponding configurations are identical, or very close, to the ones obtained in the detailed analysis, proving that good results can be obtained also with a reduced number of simulations. This aspect is very important when the evaluation of the objective functions is time consuming, as it happens in all cases in which very complex engine geometries are under investigation.

5. Conclusions

The high number of interacting parameters, which need to be defined, from a very early stage of design of a modern engine, entails the use of efficient optimization strategies together with advanced fluid dynamic simulation models. In the present work single-and multi-objective methodologies were assessed to define a best practice technique, depending on the complexity of the problem to be dealt with, on the number of design parameters, objective variables and constraints. A single cylinder spark ignition engine, used in a motorbike application, was chosen as rapresentative test case, to allow reduced computational times, without any loss of generality of the results. The analysis evaluated the convergence and efficiency of each methodology. The achieved goal was not the definition of an ever valid mathematical strategy, but here focus was given on the parallel application of a detailed fluid dynamic code and automated optimization techniques. Conclusions arising from the proposed investigation can be summarized as follows:

- Direct search methods should be preferred to evolutionary algorithms to solve single objective problems, such as the performance optimization at full load, since fewer runs are required to reach the solution; the higher is the number of variables the stronger is the relevance of this choice.
- When multi-objective problems are considered (i.e. NO_x emissions and specific fuel consumption), the NSGA-II algorithm generally performs better than the ε-constraint with the MADS method. The latter might fail in the search of a Pareto-solution and requires a higher number of runs because many subproblems lead to the same solution.
- When the dimensions of the problem increase, the use of coarser grids, as shown in the analyzed test cases, is preferable, since results are very close to the ones obtained with a detailed but much more time consuming analysis.
- The NSGA-II method is more efficient if the creation of the first generation is done on the basis of preliminary DoE techniques. The thermal and fluid dynamic response of the studied systems is such that the extremities of the Pareto-frontier can already be identified in the first generation.

Appendix A

A.1. The ε-constraint with the MADS method

The ε-constraint method [17] creates a single-objective subproblem for each nondominated solution we want to find. The optimization procedure is very simple, and it starts finding the best values (and thus the corresponding optimal set of variables) for each objective. In this way the extremities of the Pareto-frontier are defined, and so its extent is defined. Now it is possible to impose the constraints on the objective functions, one for each nondominated solution, and solve the corresponding single-objective optimization. For example, if we want to optimize the function f_2, constraints must be imposed on the function f_1. So, for the generic ith solution, we impose the ε_i inequality constraint on f_1 and solve the following problem:

$$\min_{\mathbf{x}} f_2(\mathbf{x}) \quad \text{t.c.} \quad f_1(\mathbf{x}) \leqslant \varepsilon_i \tag{A.1}$$

The Mesh Adaptive Direct Search (MADS) algorithm [29,30] was chosen among the class of direct search methods to solve single-objective problem within the ε-constraint method. The MADS method was firstly proposed by Audet and Dennis [29] and it is a generalization of the pattern search algorithms. As the name implies, this method generates iterates on a tower of underlying meshes on the domain space, but it performs an adaptive search on the meshes controlling their refinement, using a random selection of vectors to define the mesh, instead of the fixed direction vectors typical of previous GPS schemes [29]. Direct search methods are designed to only use function values. They do not compute nor attempt to approximate derivative information. These methods treat the optimization problem as a black box, which, given an input value x, it computes and returns an objective function value $f(x)$ together with a measure of its feasibility or infeasibility with respect to the domain Ω [31]. The MADS algorithm is designed for non-smooth optimization problems; the convergence analysis validates that MADS works on smooth problems. In fact, the strength of the convergence results are proportional to the smoothness of f and Ω.

It treats any optimization problem as a black box, i.e. given an input value x, it computes and returns an objective function value $f(x)$ together with a measure of its feasibility or infeasibility with respect to Ω (feasible region) [31]. Each iteration consists in three steps: SEARCH, POLL and UPDATE. Given an initial mesh and a feasible initial point the SEARCH (optional) step employs some finite strategy to try to identify a better feasible point on the mesh. If the SEARCH failed, the objective function must be evaluated at neighboring trial points in the POLL step. If the step was successful, then the mesh size is increased according to the mesh size parameter in the UPDATE step and the algorithm restarts from improved point. The algorithm is constructed in such a way that any trial point generated by a step is required to lie on the current mesh, the coarseness of which is governed by a mesh size parameter $\Delta_k \in \mathbb{R}_+$, where at iteration k, the mesh is formally defined as [30]:

$$M_k = \{x \in S_k\} \cup \left\{x_k + \Delta_k^m Dz | z \in \mathbb{N}^{n_D}\right\} \tag{A.2}$$

where S_k is the set of points where the objective function f had been evaluated by the start of iteration k, and D are the directions. The goal of each iteration is to find a trial mesh point with a lower objective function value than the current incumbent value. Such a trial point represents an improved mesh point, and the iteration is successful. If no improved point is found, the iteration is said to be unsuccessful [30].

A.2. The NSGA-II method

Among the wide class of Genetic Algorithms, the NSGA-II method was employed in this work. It is a multiobjective algorithm which searches for all the Pareto-optimal solutions simultaneously. In a Genetic Algorithm, the optimal solutions are obtained simulating an evolutionary process. Each individual of a population represents a set of variables, and all the individuals are ranked based on the values of their objective functions, their relative position in the objective space and the constraint's violation. In this way, the best individuals are recombined to create a new generation. This process is repeated for each generation, until the desired number of generations has been analyzed.

Details about the procedure adopted by NSGA-II can be found in [23]. In a generic generation, we start with a population of n individuals, which are called *parents*. The objective functions are evaluated for each individual, and the population is ranked according to some fitness parameters the *nondominated rank* and the *crowding distance*). A subset of the population is then chosen randomly, and the best individual is chosen to create a new population. This procedure is called *binary tournament* and is repeated until the desired amount of individuals is extracted. These individuals are then recombined (*crossover*) to create *n* new individuals, called *children*. In this passage, the *mutation* operator is also applied. Then objective functions and constraints are evaluated for the children population, and the parents and children population are merged, thus creating a single *intermediate* population made of 2*n* individuals. This population is ranked according to the fitness parameters and then only the best *n* individuals are chosen to create the new parents population of the subsequent generation. In this way *elitism* is achieved: it means that the best individuals of the previous generation are kept in the new generation. This feature is very important for Genetic Algorithm because it improves the convergence rate.

The fitness parameters mentioned above are very important for the optimization process. The first one is the *nondominated rank*. It specifies to which nondominated-frontier the individual belongs. To evaluate this information, all the individuals are compared, and the best nondominated rank is assigned to the nondominated individuals. The same procedure is then performed on the dominated individuals, identifying a second nondominated-frontier, to which a different rank is assigned. It is repeated until the nondominated rank has been assigned to all the individuals of the population. During the optimization process it is also important to preserve diversity, that is to prevent individuals to be crowded in some parts of the Pareto frontier. This feature is important to prevent premature convergence and to create a good quality approximation of the Pareto frontier. For these reasons a second parameter is taken into account when comparing two individuals: the *crowded distance*. If we consider a generic individual, the crowded distance is calculated as the average size of the cuboid side length built on the individuals next to it. In the binary tournament, the choice of the best individual is driven by the two described fitness parameters, but it also depends on the violation of the constraints when comparing two individuals.

A last important issue is the creation of the first generation. If the choice is good, a smaller number of generations will be required to drive the population on the Pareto frontier. This population is tipically created randomly, but more efficient methodologies can be adopted [1]. The authors decided to implement two different options for this task. In both of them, some preliminary tests are made with *Design of Experiment* techniques. In this way, a quadratic model of the responses is created, and it is possible to predict the nondominated set of variables. The first technique uses the predicted nondominated solutions as the initial population. Obviously, this technique will be more effective if the quadratic model correctly approximates the response. The second approach consists of an optimization of each objective involved in the MOP problem and the nondominated solution found in this first step are taken as initial population. In such a way, more function evaluations are needed to obtain the initial population, but its quality does not depend on the quadratic model and the extremities of the Pareto-frontier are already identified in the first generation.

References

[1] Fateh N, Parashar S, Silvestri J. Game theory approach to engine performance optimization. SAE 2008-01-0871.
[2] Macek J, Vtek O, Polsek M, Valsek M, Sika Z. Global and local optimization of transient diesel engine operations by combined one and three dimensional simulations. Thiesel 2008; 2008. p. 501–14.
[3] Rask E, Sellnau M. Simulation-based engine calibration: tools, techniques, and applications. SAE 2004-01-1264.
[4] Shi Y, Reitz RD. Optimization study of the effects of bowl geometry, spray targeting and swirl ratio for a heavy-duty diesel engine operated at low-and high-load. Int J Engine Res 2008;9(4):325–46.
[5] Ganapathy T, Murugesan K, Gakkhar RP. Performance optimization of Jatropha biodiesel engine model using Taguchi approach. Appl Energy 2009;86:2476–86.
[6] Saerens B, Vandersteen J, Persoons T, Swevers J, Diehl M, Van den Bulck E. Minimization of the fuel consumption of a gasoline engine using dynamic optimization. Appl Energy 2009;86:1582–8.
[7] Gumus M, Atmaca M, Yilmaz T. Efficiency of an Otto engine under alternative power optimizations. Int J Energy Res 2009;33(8):745–52.
[8] Atashkari K, Nariman-Zadeh N, Golcu M, Khalkhali A, Jamali A. Modelling and multi-objective optimization of a variable valve-timing spark-ignition engine using polynomial neural networks and evolutionary algorithms. Energy Convers Manage 2007;48(3):1029–41.
[9] Onorati A, Ferrari G, D'Errico G. 1D unsteady flows with chemical reactions in the exhaust duct-system of S.I. engines: predictions and experiments. SAE 2001-01-0939, 2001. SAE Trans J Fuel Lubricants; 2002.
[10] D'Errico G, Ferrari G, Onorati A, Cerri T. Modelling the pollutant emissions from a S.I. engine. SAE 2002-01-0006. SAE Trans J Fuel Lubricants; 2003.
[11] D'Errico G. Prediction of the combustion process and emission formation of a bi-fuel si engine. Energy Convers Manage 2008;49(11):3116–28.
[12] D'Errico G, Cerri T. Application of derivative-free search algorithms for performance optimization of spark ignition engines. SAE paper 2008-01-0354; 2008.
[13] Nocedal J, Wright SJ. Numerical optimization, Springer series in operations research. New York: Springer-Verlag; 1999.
[14] Abramson MA. Handbook of evolutionary computation. PhD thesis, Rice University, Houston; 2002.
[15] Back T, Fogel DB, Michalewicz Z. Pattern search algorithms for mixed variable general constrained optimization problems. New York: Institute of Physics Publishing and Oxford University Press; 1997.
[16] Kim IY, De Weck O. Adaptive weighted sum method for bi-objective optimization. Struct Multidiscip Optim 2005;29(2):149–58.
[17] Haimes YY, Lasdon LS, Wismer DA. On a Bicriterion formulation of the problems of integrated system identification and system optimization. IEEE Trans Syst Man Cybern 1971;1(3):296–7.
[18] Das I. Nonlinear multicriteria optimization and robust optimality. Houston Texas: Rice University; 1997.
[19] Das I, Dennis JE. An improved technique for choosing parameters for pareto surface generation using normal–boundary intersection. In: Proceedings of the 3rd world congress on structural and multidisciplinary optimization; 1999.
[20] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: Proceedings of the 1st international conference on genetic algorithms; 1985. p. 93–100.
[21] Fonseca CM, Fleming PJ. Genetic algorithms for multi-objective optimization: formulation, discussion and generalization. In: Proceedings of the 5th international conference on genetic algorithms; 1993. p. 416–23.
[22] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. Evolut Comput 1994;2:221–48.
[23] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitism multi-objective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 2002;6:182–97.
[24] Zitzler E, Thiele L. An evolutionary algorithm for multiobjective optimization: the strength pareto approach. Technical report 43, Computer Engineering and Communication Network Lab, Swiss federal Institute of Technology, Zurich; 1998.
[25] Knowles J, Corne D. The pareto archived evolution strategy: a new baseline algorithm for multiobjective optimization. In: Proceedings of the 1999 congress on evolutionary computation; 1999. p. 98–105.
[26] Horn J, Nafpliotis N, Goldberg DE. A niched pareto genetic algorithm for multiobjective optimization. In: Proceedings of the 1st IEEE conference on evolutionary computation, vol. 1; 1994. p. 82–7.
[27] Ghosh A, Dehuri S. Evolutionary algorithms for multi-criterion optimization: a survey. Int J Comput Inform Sci 2004;2(1).
[28] Reyes-Sierra M, Coello Coello CA. Multi-objective particle swarm optimizers: a survey of the state-of-the-art. Int J Comput Intell Res 2006;2(3):287–308.
[29] Audet C, Dennis JE. Analysis of generalized pattern searches. SIAM J Optim 2003;13:889–903.
[30] Audet C, Dennis JE. Mesh adaptive direct search algorithms for constrained optimization. SIAM J Optim 2006;17(1):188–217.
[31] Bchard V, Audet C. Robust optimization of chemical processes using a MADS algorithm, Les Cahiers du GERAD, G-2005-16, Montreal; 2005.
[32] Winterbone DE, Pearson RJ. Design techniques for engine manifolds. London: Professional Engineering Publishing; 1999.