MULTI-FIDELITY CONCURRENT AERODYNAMIC OPTIMIZATION OF ROTOR BLADES IN HOVER AND FORWARD FLIGHT

Enric Roca León, enric.roca_leon@onera.fr, Airbus Helicopters (France)
Jean-Antoine Désidéri, jean-antoine.desideri@inria.fr, INRIA (France)
Arnaud Le Pape, arnaud.lepape@onera.fr, ONERA The French Aerospace Lab (France)
David Alfano, david.alfano@airbus.com, Airbus Helicopters (France)

Abstract

Complementary multi-objective strategies adapted to the aerodynamic optimization of helicopter rotor blades in hover and in forward flight are developed. A first competitive strategy is based on Nash Games from game theory, where the objective functions are minimized by virtual players involved in a non-cooperative game. A method is presented to split the design vector into two sub-spaces, defined to be the strategies of the players in charge of the minimization of the primary and the secondary objective functions respectively. This split of territory allows the optimization of the secondary function while causing the least possible degradation of the first one. Alternatively, a cooperative approach based on a generalization of the steepest-descent method to multiple objectives is presented. These methodologies are applied to the optimization of the twist distribution of the model rotor ERATO, with the aim to maximize the Figure of Merit in hover while minimizing the rotor torque coefficient in forward flight. The optimizations are performed in a framework based on high-fidelity evaluations. A multi-fidelity model is proposed and tested, creating a bridge function between the low and high fidelity models. The results demonstrate the potential of these techniques to obtain rotor designs realizing interesting trade-offs.

NOMENCLATURE

c	Mean chord $[m]$
$\bar{C} = \frac{100C}{\frac{1}{2}\rho_\infty S\sigma R(R\Omega)^2}$	Rotor torque coefficient
$FM = \frac{\bar{Z}^{3/2}}{20\bar{C}}\sqrt{\sigma}$	Figure of Merit
N_b	Number of blades
R	Rotor radius $[m]$
$S = \pi R^2$	Rotor disk surface $[m^2]$
$\bar{Z} = \frac{100Fz}{\frac{1}{2}\rho_\infty S\sigma(R\Omega)^2}$	Rotor thrust coefficient
$\mu = V_\infty/(\Omega R)$	Advance ratio
Ω	Rotor rotational speed $[rad/s]$
$\sigma = \frac{N_b c}{\pi R}$	Rotor solidity

1 INTRODUCTION

The design and aerodynamic optimization of helicopter rotor blades involves multiple objectives which are often antagonistic in nature. Such is the case for maximizing the rotor efficiency in hover and minimizing the rotor shaft torque in forward flight. In addition, the rotor flow field is complex, featuring strong Fluid-Structure Interaction (FSI) which must be taken into account to correctly predict the blade aerodynamic loads. Accurately modeling the rotor flow field implies the use of Computational Fluid Dynamics (CFD) and Computational Structural Dynamics (CSD), which remain still computationally intensive. In this context, the use of high fidelity information in rotor blades optimization process is a topic of active interest in the engineering community. The use of the adjoint formulation has successfully been applied in steady and unsteady cases in single objective optimizations making an efficient use of gradient based algorithms (Dumont [1] and Choi [2]). Alternatively, the use of surrogate models based on CFD simulations has been used in various works by Allen [3], Glaz [4], Leusink [5], Johnson [6] and Massaro [7]. However, the building of such surrogate models remains computationally costly when using high fidelity CFD-CSD evaluations. In order to alleviate this problem, multi-fidelity strategies combine information of multiple models of varying accuracy in order to approximate the high fidelity model using less samples. Such is the case in the work presented by Collins [8], which uses a correction metamodel to create a bridge function between a comprehensive code and CFD-CSD, and also in the works of Imiela and

Wilke [9, 10], which make use of the so-called Hierarchical Kriging (proposed by Han [11]), where the low fidelity model is used as a trend for the high-fidelity kriging surrogate.

Moreover, multiple optimization strategies are used to solve the multi-objective problem. An usual approach is to use some form of weighted function agglomerating the functionals into a single one (Refs. [6] and [12]) as to optimize a single point, or instead to search for the complete Pareto front using stochastic algorithms (Refs. [5] and [7]). Alternatively, two complementary optimization algorithms are presented in this article. The paper is structured as follows: first an introduction to the proposed multi-objective algorithms is given. Next, the numerical framework used in the optimization applications is described, including comparisons of the results of different fidelity models in forward flight. The multi-fidelity approach is then presented and validated. Finally, the application of the proposed multi-objective algorithms are applied to the twist optimization of the ERATO rotor, and the results are discussed and analyzed.

2 MULTI-OBJECTIVE OPTIMIZATION STRATEGIES

In this section we introduce a brief theoretical basis on the proposed alternative algorithms for the treatment of multi-objective problems, focusing on the bi-objective case. Two complementary approaches are presented, namely competitive and cooperative optimizations. The competitive phase is adapted to those cases where the treated objective functions are strongly antagonistic, that is when improvements in one criterion lead to degradation of the other function. This problem can be treated using Nash Games, introduced in the next section. On the other hand, a cooperative optimization phase is possible whenever all the criteria can be optimized at the same time. The Multi-Gradient Descent Method (MGDA), a generalization of the steepest-descent method, is adapted to an efficient resolution of this kind of problem.

2.1 Nash Games With Territory Splitting

Nash games where first introduced by Nash in the 1950s [13] in the framework of game theory. In this approach, the system is modeled as a game in which each optimization objective is adjusted by a virtual player trying to improve its value. All the players simultaneously try to improve their objectives taking into account the other players' strategies, which are considered fixed. Let us consider a game with 2 virtual players or disciplines, namely J_A and J_B (as such is the case for the rotor optimization in hover and forward flight). Both disciplines share the design vector $Y = (Y_A, Y_B)$, that can be split in two subvectors. During the game, each subvector Y_i is assigned to its respective player J_i, who tries to improve its objective using its portion of the design territory while taking into account the actions of their opponent (i.e. the other discipline's subvector). A Nash equilibrium $\hat{Y}$ is reached when no further improvements are obtained for each player if their opponent keeps its strategy unchanged, verifying:

$$(1) \qquad \begin{cases} \hat{Y} & = & (\hat{Y_A}, \hat{Y_B}) \\ \hat{Y_A} & = & \arg\min_{Y_A}\ J_A(Y_A, \hat{Y_B}) \\ \hat{Y_B} & = & \arg\min_{Y_B}\ J_A(\hat{Y_A}, Y_B) \end{cases}$$

Where $\arg\min_{Y_i}$ stands for argument of the minimum, meaning the point Y_i for which J_i attains its minimum value. How the design variable vector is split is the crucial step, as this determines the solution. Rather than splitting directly the original design vector, which introduces some arbitrariness, Désidéri [14] proposed a method to split the variables territory based on sensitivity analysis. In this framework a hierarchy of two disciplines is considered with one of them being regarded as principal or fragile (J_A). In our work the hover efficiency (Figure of Merit, or FM) is considered the principal discipline J_A, whereas in forward flight J_B is the rotor torque coefficient $\bar{C}$. The main idea is that, after a successful single objective optimization in hover (maximizing FM), a multi-objective competitive optimization is conducted to optimize $\bar{C}$ in forward flight, assuring that the gains obtained on the first optimization are not degraded in excess. The proposed split of the design vector (of length N) assures the sub-optimality of the principal criterion and is defined as follows:

$$(2) \qquad Y = Y(U, V) = Y_A^{\star} + S \begin{pmatrix} U \\ V \end{pmatrix}$$

$$(3) \qquad U = \begin{pmatrix} u_1 \\ \vdots \\ u_{N-p} \end{pmatrix} \quad V = \begin{pmatrix} v_1 \\ \vdots \\ v_p \end{pmatrix}$$

In which $Y_A^{\star}$ corresponds to the previously found optimum of the principal discipline alone and S is an invertible NxN matrix henceforth referred to as *splitting matrix*. The U and V subvectors correspond to the design vectors of the principal and secondary functionals respectively. The splitting matrix is defined such as small perturbations of the parameters about $Y_A^{\star}$ that affect the secondary criterion cause the least possible degradation to J_A. The intended approach consists in conducting the optimization of the secondary discipline (i.e. J_B) as a small perturbation from the original optimum $Y_A^{\star}$. The solution corresponds to a Nash equilibrium between the two disciplines.

At the end of the first optimization, access to $J_A^\star$, its gradient $\nabla J_A^\star$ and the Hessian matrix $H_A^\star$ via direct evaluation or surrogate models is supposed. Assuming that $Y_A^\star$ is a local or global unconstrained minimum of the functional J_A, then $\nabla J_A^\star = \overrightarrow{0}$ and the Hessian matrix $H_A^\star$ is real, symmetric and positive definite (hence all its eigenvalues h_i are real and positive). In this case, the Hessian can be diagonalized obtaining the matrix Ω_H formed by the eigenvectors ω_i of the Hessian.

$$H_A^\star = \Omega_H \Lambda_H \Omega_H^T \tag{4}$$

$$\Lambda_H = Diag(h_i) \tag{5}$$

$$\Omega_H = \{\overrightarrow{\omega_i}\} \tag{6}$$

The territory of the secondary functional should be taken to be the span of *p* eigenvectors of the Hessian matrix associated with the smaller eigenvalues h_i. The eigenvalues are thus ordered as a monotone decreasing sequence verifying Eq. 7 and the splitting matrix S is obtained reordering the eigenvectors according to the sorted eigenvalues (Eq. 8).

$$h_1 \geq h_2 \geq \ldots \geq h_N \tag{7}$$

$$S = (\overrightarrow{\omega}_1 \quad \overrightarrow{\omega}_2 \quad \ldots \quad \overrightarrow{\omega}_N) \tag{8}$$

The search of the Nash equilibrium could be applied directly to the problem as defined in Eq. 1, but a more convenient redefinition of the multi-objective problem is proposed by Désidéri [14]:

$$\begin{cases} \min\limits_{U \in \mathbb{R}^{N-p}} J_A[Y(U,\hat{V})] \\ \min\limits_{V \in \mathbb{R}^p} J_{AB}[Y(\hat{U},V)] = \frac{J_A}{J_A^\star} + \epsilon\left(\theta\frac{J_B}{J_B^\star} - \frac{J_A}{J_A^\star}\right) \end{cases} \tag{9}$$

Fixed subvectors are represented by $\hat{U}$ and $\hat{V}$, ϵ is a continuation parameter comprised between 0 and 1 that allows a gradual introduction of the antagonism between disciplines and θ represents a relaxation factor which in practice is set to 1.

Three main theoretical results are obtained using this formulation. First, the optimality of the chosen S matrix to reflect the hierarchical sensitivities of J_A. Secondly, a Nash equilibria exists at the optimum of the principal discipline $Y_A^\star$ ($\epsilon = 0$), which is consistent with the single-optimization results. Moreover, by continuity a smooth continuum of Nash equilibria exists starting from $\epsilon = 0$ throughout $\epsilon = 1$ (which corresponds to a pure minimization of J_B). Finally, given a Nash equilibrium for a certain ϵ ($\hat{Y}_\epsilon$), the principal criterion J_A is 2^{nd} order insensitive with respect to variations of ϵ. This is due to the fact the the Hessian-based split is developed considering the Taylor's expansion of J_A about $Y_A^\star$ in the direction of a unit vector w:

$$J_A(Y_A^\star + \epsilon w) = J_A(Y_A^\star) + \epsilon \nabla J_A^\star w + \frac{\epsilon^2}{2} w H_A^\star w + \mathcal{O}(\epsilon^3) \tag{10}$$

And in consequence, provided that $Y_A^\star$ is an optimum, the gradient at this point is null, thus verifying:

$$J_A(\hat{Y}_\epsilon) = J_A^\star + \mathcal{O}(\epsilon^2) \tag{11}$$

The Nash equilibrium at each given ϵ can be obtained via an iterative process in which each discipline runs a small number of optimizing iterations before exchanging information with the other discipline, until convergence (or a relaxed convergence) is obtained. In pseudo-code, this process can be described as:

1. Initialize the design vector: $Y := Y(U^0, V^0)$
2. Loop while $Y(U^{i+1}, V^i) \neq Y(U^i, V^{i+1})$:
   (a) *Player A*: $Y = Y(U, V^i)$
   Perform K_A iterations for the optimization problem $min\ J_A(U, V^i)$ to obtain U^{i+1}.
   (b) *Player B*: $Y = Y(U^i, V)$
   Perform K_B iterations for the optimization problem $min\ J_B(U^i, V)$ to obtain V^{i+1}.
   (c) Update the design vector: $Y := Y(U^{i+1}, V^{i+1})$

The convergence of such an algorithm for an arbitrary split is not assured. However, the formulation proposed by Désidéri guarantees the existence of Nash Equilibria in the neighborhood of the optimum of the principal objective. Hence in practice additional measures to accelerate the algorithm's convergence ratio are not needed, at least in the author's experience (if necessary, Attouch et al. [15] note that the convergence rate can be improved by using damping mechanisms in the criteria). In practice, the convergence termination criterion is relaxed. Numerically, convergence is assumed attained when either the L-2 norm of the squared difference of the scaled design vectors is smaller than a certain positive δ or when a maximum number of exchanges between disciplines is reached (Eq. 12). Scaling of the design vectors for the test is necessary in order to account for the difference of magnitudes between the design variables.

$$\begin{cases} \parallel Y(U^{i+1}, V^i) - Y(U^i, V^{i+1}) \parallel^2 & \leq & \delta \\ i & \geq & N_{max} \end{cases} \tag{12}$$

The described formulation in this section is valid for unconstrained problems, but constrained problems can also be treated, in which case additional steps are needed in the computation of the splitting matrix (see Desideri [14] for further details).

2.2 Multi-Gradient Descent Algorithm

In contrast with the Nash Game competitive approach, the general problem of unconstrained minimization can be treated so that all the criteria are simultaneously minimized. In other words, consider a cooperative strategy beneficial to all the objectives. In the case where the gradients can be computed at an initial design point, it can be shown that a direction along which all the criteria are minimized can be found in the convex hull of the gradients. This is the basis of the Multi Gradient Descent Algorithm (MGDA) [14], which is, as noted above, a generalization of the classical steepest-descent method to multi-objective optimization. The principles and formulation of MGDA are presented in this section, based on the developments of Désidéri presented in detail in [16, 14]. The context is the simultaneous optimization of n smooth criteria (or disciplines) $J_i(Y)$ (where Y is the design vector, $Y \in \mathbb{R}^N$). The cooperative optimization improving all criteria is started at a initial point Y^0 that is not Pareto optimal (i.e. not in the Pareto Front). In practice, the functions are assumed to be of class C^1 and locally convex. We note the family of the function gradients at the initial point as $\{u_i^0\} = \{\nabla J_i(Y^0)\}$ for $i = 1, \ldots, n$.

In this setting, consider the set $\mathcal{U}$ formed by the strict convex combinations of these vectors:

$$(13) \quad \mathcal{U} = \left\{ w \in \mathbb{R}^N \,/\, w = \sum_{i=1}^{n} \alpha_i u_i^0; \quad \alpha_i \geq 0 \,(\forall i); \quad \sum_{i=1}^{n} \alpha_i = 1 \right\}$$

and $\bar{\mathcal{U}}$ the convex hull of the family.
Then, there exists a unique element $\omega \in \bar{\mathcal{U}}$ of minimum norm, and:

$$(14) \qquad \forall \bar{u} \in \bar{\mathcal{U}} \quad : \quad (\bar{u}, \omega) \geq (\omega, \omega) = \parallel \omega \parallel^2$$

Let us introduce the concept of Pareto stationarity at a point Y^0. The point Y^0 is said to be Pareto stationary if there exists a convex combination of the gradients u_i^0 that is equal to zero:
(15)

$$\exists \alpha = \{\alpha_i\} \in \mathbb{R}^n / \alpha_i > 0 \, \forall i; \quad \sum_{i=1}^{n} \alpha_i = 1; \text{ and } \sum_{i=1}^{n} \alpha_i u_i^0 = 0$$

Then, two situations are possible, regarding ω :

1. Either $\omega = 0$, and the criteria $J_i(Y)\,(\forall i)$ are Pareto-stationary;
2. Or $\omega \neq 0$, and $-\omega$ is a descent direction common to all the criteria. Additionally, if $w \in \mathcal{U}$ the scalar product $(\bar{u}, \omega)$ is equal to $\parallel \omega \parallel^2$ for all $\bar{u} \in \bar{\mathcal{U}}$.

Considering a case with just two criteria, three configurations of the objective function gradients are possible, as presented in Figure 1. In this particular case, ω can be expressed explicitly. Besides the trivial case where $u = v = \omega$, the convex hull is represented by the segment $\bar{uv}$ connecting the extremities of both vectors, which share the same origin O (this is acceptable as only the norm and the angle between gradients is of importance). Consider the vector $\omega^\perp$ with origin at O whose extremity is the orthogonal projection of O onto the line containing the segment $\bar{uv}$. If the vector $\omega^\perp$ is in the convex hull (i.e. the extremity of the vector lays in the $\bar{uv}$ segment) then $\omega = \omega^\perp$. Otherwise, ω is equal to the criterion gradient of smallest norm. Explicitly:

$$(16) \qquad \omega = (1 - \alpha)u + \alpha v$$

This convex combination is orthogonal to $\bar{uv}$ for a certain $\alpha^\perp$:

$$(17) \qquad \alpha^\perp = \frac{(u, u - v)}{(u - v, u - v)}$$

Thus, if $\alpha^\perp \in [0, 1]$, then $\alpha = \alpha^\perp$. Otherwise, $\alpha = 0$ or 1 depending on whether $\alpha^\perp < 0$ or > 1.

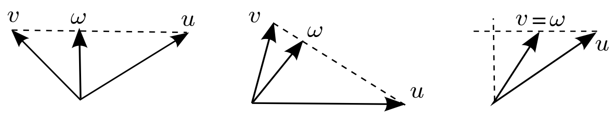


Figure 1: Various possible configurations of the two gradients-vectors u and v and the minimal-norm element ω (extracted from [17]).

In the case where more than two objectives are considered, the determination of ω is done via a minimization of the following constrained quadratic form:

$$(18) \qquad \begin{array}{ll} \text{Minimize:} & \min\limits_{\alpha \in \mathbb{R}^n} \left\| \sum\limits_{i=1}^{n} \alpha_i u_i^0 \right\|^2 \\ \text{subject to:} & \alpha_i \geq 0 (\forall i), \quad \sum\limits_{i=1}^{n} \alpha_i = 1 \end{array}$$

Once the common descent direction ω is found for all the criteria, a new candidate point is searched in this direction, that is:

$$(19) \qquad Y^{k+1} = Y^k - \rho \omega(Y^k)$$

Where Y^{k+1} is the new candidate, Y^k the current point, k is the MGDA iteration number and ρ is a positive step length. Once a new candidate is identified, after a linesearch in this direction the ω at this point is re-computed and the process is repeated until a convergence criterion is met. In single objective optimization this stopping criterion is a sufficiently small (close to zero) gradient norm. For multiple objectives, the optimization process is stopped when the Pareto stationarity condition is met. In practice, Pareto stationarity is guaranteed when at least one of the criteria gradients is of zero norm (which following our formulation implies that $\| \omega \|$ is also zero, if no gradient scaling is applied). For this reason, in practice the convergence criterion is met when $\| \omega \| \leq \delta$ for a user-defined small positive δ.

The optimization process can thus be resumed in the following pseudo-code:

- Initialize the design vector: $Y := Y^0$
- Loop while $\| \omega \| \leq \delta$:
  1. Compute $J_i(Y), \quad (1 \leq i \leq n)$
  2. Compute $\nabla J_i(Y), \quad (1 \leq i \leq n)$
  3. Identify ω
  4. Perform a line-search in the direction $-\omega$: determine optimal ρ
  5. Update the candidate: $Y := Y - \rho\omega$

3 HIGH-FIDELITY AERODYNAMIC MODELS: ERATO APPLICATION

The high-fidelity aerodynamic models used in the computation of the rotor performance in hover and in forward flight along with the numerical details of the implementation are presented in this section, with special emphasis on the forward flight case. This numerical framework was validated on the ERATO rotor. This rotor, developed in a joint program between Eurocopter, ONERA and DLR was designed to reduce noise emissions [18]. It features a 2.1m radius, a mean chord of 0.14m and a linear aerodynamic twist of $-10°/R$. The blade planform has forward and backward sweep as well as a non-optimized straight tip.

3.1 Hovering Computations

High-fidelity evaluations solving the RANS equations is employed in hover (assuming a rigid blade), using a numerical framework already presented in a previous work [19]. A brief summary of the numerical parameters is given, for further details refer to Roca [19]. The RANS equations were solved employing elsA [20], the Computational Fluid Dynamics (CFD) code developed at ONERA.

A Roe MUSCL 2^{nd} order scheme (with a Van Albada limiter) was used to discretize the space. The choice of the Roe scheme was imposed in order to assure consistency/compatibility with the available adjoint solver formulation [21]. The turbulence is modeled using the $k - \omega$ Kok model with a shear stress transport (SST) correction.

The mesh (a quarter of rotor) was generated using an in-house analytic meshing tool, which yields structured meshes. The blade tip is simplified by a degenerated section. The retained size is of 0.81 million points. Even if this is a relatively coarse grid it is deemed adequate for optimization purposes. The main objective being to accurately model the global performances in hover at the lowest-possible cost. The required (cumulated) CPU time for each simulation was of approximately 9.3 hours using Xeon 5500 cores.

3.2 Forward Flight Computations

In forward flight, the blade was assumed to be elastic, and time-marching computations were carried out using a loose coupling approach [22]. The blade dynamics and deformations were provided by the rotor comprehensive code HOST [23], developed by Airbus Helicopters. In this code, the blade dynamics is modeled by a 1D Euler-Bernoulli beam model, coupled with a simplified aerodynamics model based on lifting-line theory. The aerodynamic coefficients are determined via interpolation of 2D airfoil lookup tables, using the computed rotor trim information in order to evaluate the local Mach numbers and angles of attack.

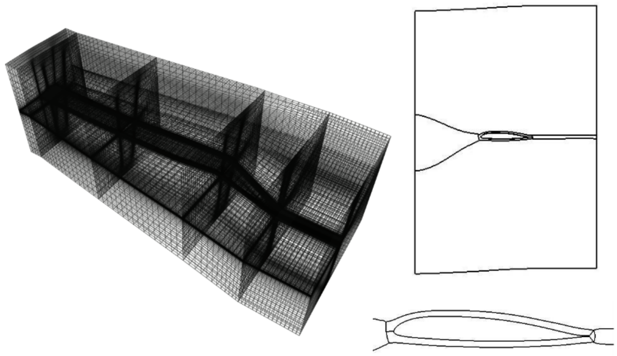
The aerodynamic loads where computed by elsA. In this approach, assuming a periodic solution, the data is exchanged between HOST and elsA. An initial search for the rotor equilibrium is done by HOST, obtaining the estimated rotor dynamics and the simplified aerodynamic loads F^0_{2D}. Subsequently, the rotor blade deformations and movements are fed to the flow solver. The mesh is deformed accordingly and elsA recomputes the aerodynamic loads W^0 which are decomposed in a Fourier series, obtaining F^0_{3D}. The difference between the aerodynamic loads of HOST and elsA is thus computed and added to the HOST aerodynamic loads. At this point, the global equilibrium is no longer verified, and thus HOST recomputes the blade dynamics, which is then exchanged with elsA. This process, repeated iteratively, can be expressed in general:

$$F^n = F^n_{2D} + (F^{n-1}_{3D} - F^{n-1}_{2D}) \quad (20)$$

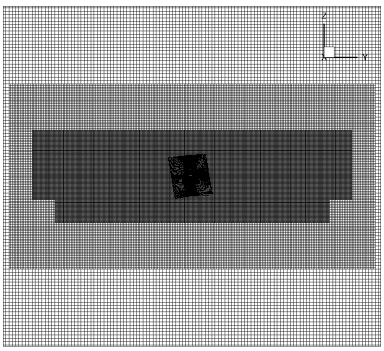
The HOST simplified aerodynamic forces and moments at the n_{th} iteration F^n_{2D} are corrected by the difference between the precise elsA results F^{n-1}_{3D} and the simplified aerodynamics loads F^{n-1}_{2D} of the previous iteration . The process converges when:

$$F_{3D}^{n-1} - F_{2D}^{n-1} \approx 0 \tag{21}$$

In our computations, the four blades were meshed and a Chimera approach was retained, using near-body refined grids spinning along the rotor blade and a non-rotating Cartesian background mesh. The background meshes were generated as an Octree. The mesh topology (based on a O-H structure) and a section of the background mesh are shown in Figures 2a and 2b respectively.



(a) Near-body grid and topology.



(b) Background grid section.

Figure 2: ERATO mesh for forward flight high-fidelity evaluations.

A relatively coarse body-grid of 766K points per blade was used, aiming to reduce the computational cost in view of the optimization framework, using a background grid of approximately 10 million points. The turbulence was modeled using $k - \omega$ Kok with a shear stress transport (SST) correction and the Zheng limiter. A time step of 1 degree was used along with a Gear scheme. Finally, the Jameson scheme was used to discretize the fluxes in space.

Precisely, the chosen design point for the ERATO rotor was at $\mu = 0.344$, $\bar{Z} = 12.5$ and a fuselage drag $C_x S/S\sigma = 0.1$ for a rotational tip Mach number $M_{\Omega R} = 0.617$ and imposing zero flapping for rotor trim (i.e. $\beta_{1c} = \beta_{1s} = 0$). At this design point experimental data was available. The CPU cost of a rotor revolution was of approximately 425 hours. Four coupling iterations where necessary in order to obtain convergence of the trim angles, thus requiring 1700 CPU hours in total (using Xeon 5500 cores).

3.2.1 Numerical Validation

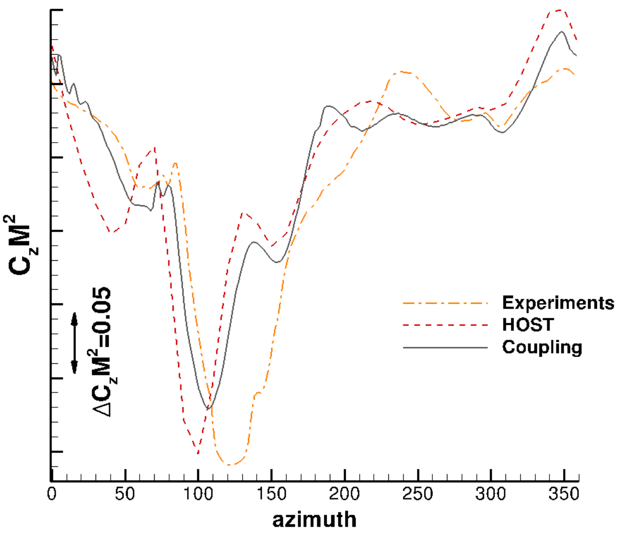
The ERATO rotor was computed using the proposed numerical framework and the results were compared to available experimental data and to HOST (using a prescribed wake model). These results are presented here in order to illustrate the need to take into account the rotor trim and the fluid-structure interaction in forward flight computations. The sectional loads at the $0.97R$ station of the ERATO blade are shown in Fig. 3.

The consideration of the elastic blade significantly improves the sectional loads prediction. The negative lift peak at the tip of the advancing blade is better captured in the elastic coupled simulation, with a better phase accuracy than for HOST. The differences are much more evident in the sectional pitching moment values, which are not well predicted by simplified aerodynamic models, especially near the blade tip. The oscillations in the pitching moment appearing around $\psi = 0°$ are due to the interaction with the rotor trailing wake.

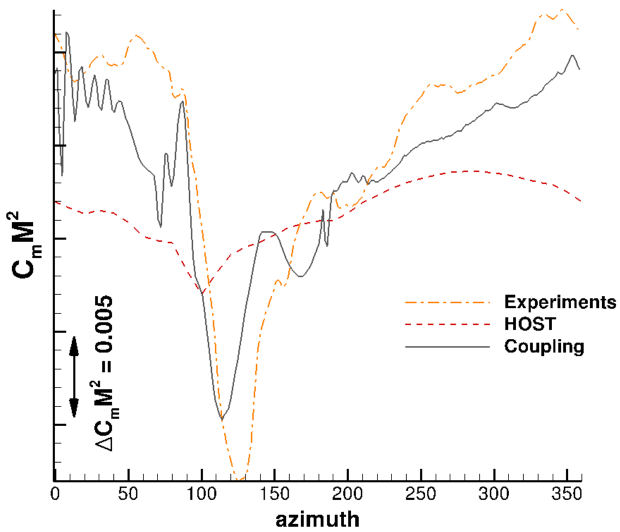
4 SURROGATE-BASED OPTIMIZATION FRAMEWORK

The use of high-fidelity evaluations directly in the optimization loop is expensive, specially for time-marching computations in forward flight. For this reason, a surrogate-based approach was used in order to reduce the cost of the optimization runs. The metamodel type chosen in this work is the popular Kriging technique. Kriging (or Gaussian Process) is an interpolation technique first developed in the geostatistics community [24]. It interpolates the value of a random field at an unobserved location using the information of observations at nearby locations. The construction of a Kriging model basically requires the choice of a trend function and a function to estimate the correlation parameters (i.e. correlation function). In this work, the trend function was assumed to be a constant (ordinary Kriging), along with a Gaussian correlation function for all the applications.

In the case of the evaluation of the Figure of Merit in hover, a standard Kriging metamodel is built. A training data set is obtained via a Design of Experiments (specifically Latin Hypercube Sampling). The training set is computed using the high-fidelity framework



(a) Sectional loads $C_z M^2$.



(b) Sectional loads $C_m M^2$.

Figure 3: Blade sectional pitching moment and lifting loads at the $0.97R$ station as computed via HOST, loose coupling and for experimental measures.

and an initial surrogate is trained. Subsequently, the metamodel is improved iteratively by adding points to the database at interesting locations. The Efficient Global Optimization [25] methodology is used to select promising points. In this algorithm, a metric known as Expected Improvement (EI) is used to guide where the sampling should take place. This metric depends on the uncertainties in the metamodel and the potential to improve the objective in the metamodel. The EI is high at sparsely sampled regions where the metamodel predicts low function values. In our particular case, the metamodel was built in preparation of Nash Games, and thus a true optimum of the objective function in hover was already known. For this reason, in addition to adding points with high EI to the database, local optimizations starting from the known optimum were carried out in order to ensure that the optimum of the metamodel converged to the true hover optimum. This iterative process improving the metamodel was terminated either when a maximum number of iterations was reached or when the estimated error was sufficiently small (i.e. mean errors of 1 point of Figure of Merit or less).

In the case of forward flight, a multi-fidelity approach was tested, using loose-coupling computations as the top fidelity level and HOST for the low fidelity level. The details are presented in the following section.

4.1 Multi-fidelity Approach

The main idea of multi-fidelity strategies is that coarser models still contain valuable information that can be used to diminish the number of evaluation calls to higher fidelity models. Some surrogate modeling strategies can integrate directly the hierarchy of models, such as Co-kriging [26], or Hierarchical Kriging [11]. Alternatively, another strategy consists in the correction of global low-fidelity models with high-fidelity samples. This technique presents the advantage that it is independent of the number of design variables, but the accuracy of the corrected model requires a well-behaved low-fidelity model. The question arises as how to choose the best correction for the low-fidelity model to rapidly converge towards the true model. A detailed review of possible correction methods is presented by Eldred [27]. This work is mainly concerned with global corrections: data global fits are used to model the relationship between the low and high levels at a limited set of points. This follows the approach adopted by Collins [8], where a correction metamodel was used to create a bridge function between a comprehensive code and CFD-CSD for the purposes of optimization.
Hence, we use a Kriging model to describe the objective function error between HOST and the loose-coupling computations instead of modeling directly the high-fidelity level. The hypothesis behind this reasoning is the assumption that the relationship between the criterion results yielded by HOST and those yielded by elsA/HOST is more easily modeled (i.e. less non-linear) than the criterion itself. Two types of correction were tested, additive (A_{err}) and multiplicative (M_{err}), defined for the $\bar{C}$ coefficient as:

$$\begin{cases} \bar{C}_{elsA/HOST} = A_{err} + \bar{C}_{HOST} \\ \bar{C}_{elsA/HOST} = M_{err} * \bar{C}_{HOST} \end{cases} \quad (22)$$

The additive and multiplicative corrections are obviously a function of the chosen low fidelity model. In the particular case of HOST, a parametric study was performed in order to choose the inflow model best reproducing the trends obtained by the high-fidelity computations. Precisely the induced velocity models of Meijer-Drees, METAR [28] and FiSUW [29] were tested to compute each correction. Subsequently, each evaluation of the torque coefficient $\bar{C}$ at a design vector was obtained by evaluating the point with the correction metamodel and HOST and using one of the above bridge functions.

5 OPTIMIZATION OF THE ERATO ROTOR

As stated above, the objective of the hover optimization is to maximize the Figure of Merit, representing the rotor efficiency in hover. In order to treat the function as a minimization problem the equivalent criterion is used:

$$J_A = 100(1 - FM) \tag{23}$$

The FM is a function of the thrust and torque coefficients, which in turn depend on the rotor solidity. In order to obtain comparable FM values between rotors, iso-solidity must be enforced. In particular, thrust-weighted solidity is forced to be constant throughout the optimization run (as recommended by Bingham [30] and Dumont [21]). This geometric requirement is directly integrated in the blade deformation module as an implicit constraint.
In forward flight, the required rotor torque $\bar{C}$ coefficient is the chosen function to be minimized:

$$J_B = \bar{C} \tag{24}$$

In this case, the problem was unconstrained (except for variable boundaries). The blade is optimized in hover for a rotor with a tip Mach number of $M_{\Omega R} = 0.617$ and a Reynolds tip number of $1.93 * 10^6$ ($Re_{\Omega R}$). In forward flight, the flight point presented in section 3.2 is used, namely $\mu = 0.344$, $\bar{Z} = 12.5$, $CxS/S\sigma = 0.1$ and $M_{\Omega R} = 0.617$, imposing zero flapping.

5.1 Variables and Parametrization

The rotor blade is parametrized using only parameters controlling geometric laws such as twist, chord and sweep along the blade span without modifying airfoil shape. The structural properties are assumed to be frozen throughout the optimization. The distribution of each geometric parameter is piloted by Bézier or cubic Splines using a fixed control point distribution, presented in Figure 4. For the case of the presented multi-objective optimizations, only the twist distribution was considered, thus using 5 optimization variables corresponding to the Bézier points.

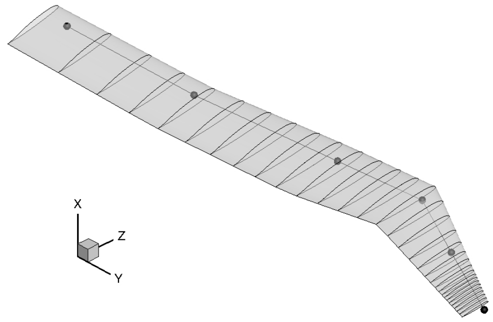


Figure 4: Control point locations for the ERATO blade optimization.

New blade geometries were generated by mesh deformation using an in-house mesh deformation tool (whenever CFD computations were involved). In the case of HOST computations, a Python module was developed in order to generate deformed blade geometry files in consistency with the mesh deformation program. In order to better condition the optimization process all the variables were normalized with respect to their respective bounds as to be comprised between zero and one.

5.2 Preliminary Hover Optimization

The application of the Nash Game formulation requires the starting optimization point to be an optimum of the principal discipline (maximum of Figure of Merit in hover). For this reason, an optimization is needed as a previous step ensure that the starting point is on the Pareto Front. In a previous work by the authors [19], Nash Games were already presented in a low-fidelity framework for the ERATO rotor. A single-objective optimization of the Figure of Merit was already carried out, using the adjoint formulation to perform the gradient-based optimization. We use this already optimized configuration in hover as a starting point for the high-fidelity multi-objective optimization.

For completeness, we include the geometric laws yielded by the single-objective optimization in Figure 5. The optimized rotor increased its maximum FM by 6.1 points, as observed in Figure 6 along with its load capacity.

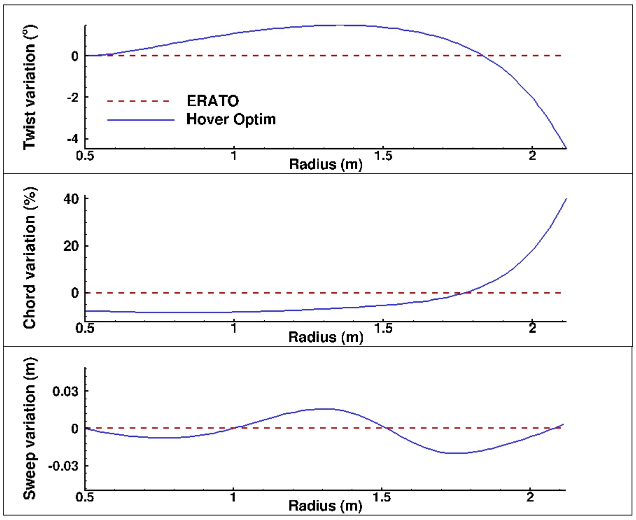


Figure 5: Geometric twist, chord and sweep law variations of the hover optimum (continuous line) with respect to the baseline rotor (dashed line), from [19].

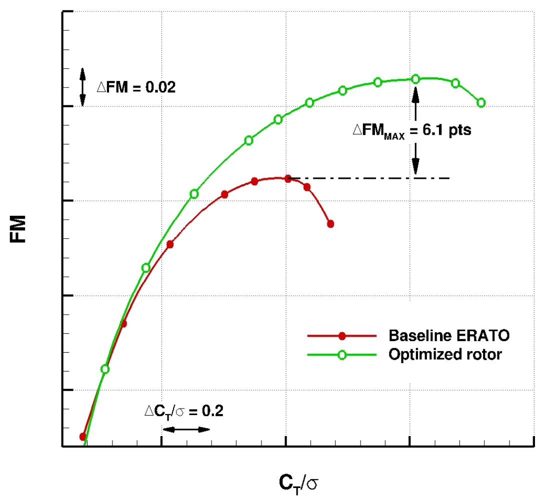


Figure 6: Maximum FM gain for the optimum with respect to the baseline rotor, from [19].

5.3 Validation of the Multi-Fidelity Model in Forward Flight

A multi-fidelity model was built following the previously described approach in preparation of the multi-objective twist optimization. In order to model the corrections, an initial Latin Hypercube Sampling plus the hover optimum were evaluated using the loose coupling strategy. Multiple correction models were subsequently built (additive/multiplicative and depending on the chosen HOST wake model). The best correction was chosen using the error estimator provided by the leave-one-out method: the error estimation for the i^{th} individual is the result of training the metamodel with all the points but the i^{th} one, and evaluating the error of the metamodel at this location. Figure 7 shows these errors for each one of the individuals conforming the metamodel training database. Only the errors of the standard kriging (without multi-fidelity) model and the results yielded by best correction are presented. The additive correction using the FiSUW inflow model appears to minimize the error.

The resulting prediction error is very satisfactory for

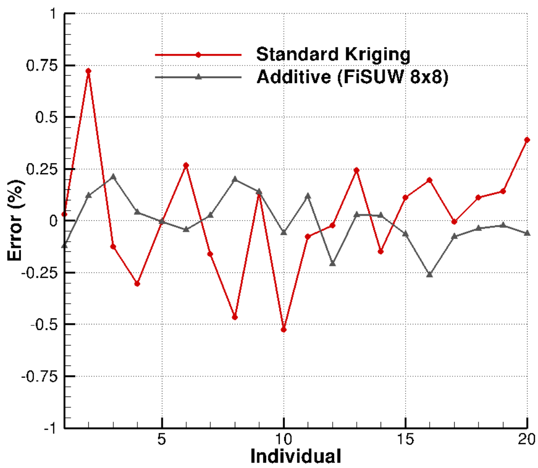


Figure 7: Error of the multi-fidelity metamodel compared to standard kriging

the FiSUW additive correction, namely lower than 0.25%. However, it should be noted that small variations of the $\bar{C}$ (typically of the order of 0.01) can have a significant impact on the consumed power. As an example, a variation of $\Delta\bar{C} = 0.01$ with respect to an initial value of $\bar{C} = 1.11$ represents a 0.9% variation, which is numerically small but physically significant. Despite this fact, the precision of the metamodel was deemed already sufficient for optimization purposes and thus no additional points were added to the training database in order to further improve the metamodel precision.

5.4 Nash Game Application

An unconstrained Nash Game was carried out optimizing the twist distribution, using the multi-fidelity metamodel in forward flight to compute the $\bar{C}$ (that is, combining HOST evaluations with the kriging model of the error). The Figure of Merit in hover was also obtained via a metamodel, as noted previously. Precisely, a kriging metamodel of 57 points (including 10 update points) was built in hover and was used to compute the Hessian based split (the splitting matrix).

Two variables were assigned to the primary discipline (hover performance, J_A) and 3 to the secondary

$(\bar{C}, J_B)$. Up to 10 function evaluations and 5 gradient evaluations where allowed at each search of the Nash Equilibria, authorizing as much as two exchanges of information between players. The optimization algorithm for the search was CONMIN [31], included in the DAKOTA library [32]. Results are shown in Fig. 8 for a complete run of the game, discretized in 5 Nash equilibria. In total, J_A was evaluated 86 times, ∇J_A 12 times, J_{AB} 92 times and J_{AB} 25 times.

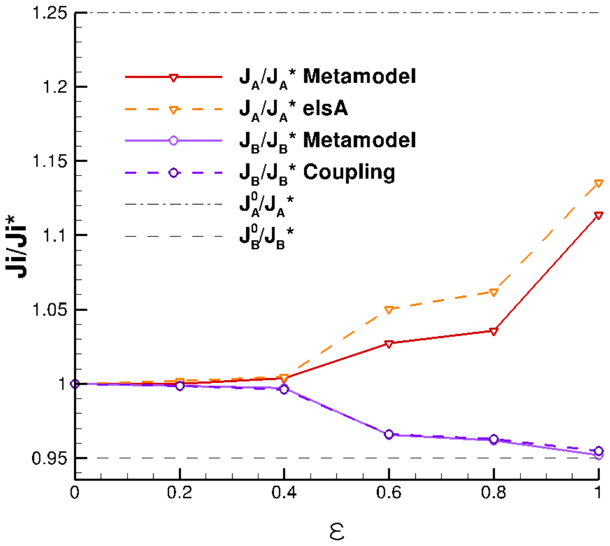


Figure 8: High-fidelity Nash Game involving the twist distribution.

The obtained Nash Equilibria where subsequently evaluated using higher fidelity methods. The multi-fidelity surrogate appears to predict with practically zero error the forward flight performance, whereas the kriging metamodel in hover slightly under predicts the degradation of the the J_A criterion for $\epsilon \geq 0.6$. However, the trends are well reproduced by the metamodel. The twist delta distribution at the last equilibrium is presented in Figure 9, remarkably similar to the results obtained in the analogous low-fidelity Nash Game presented in [19]. At this point, a 3.1 points increase of FM with respect to ERATO is obtained for practically the same rotor shaft torque (0.05% increase with respect to ERATO).

In this case, the use of higher-fidelity methods appears to have a relatively small impact in the optimization results. This can be explained by the fact that HOST predicts rather accurately the rotor performance as a function of the twist (via 2D lookup tables and including a correction for the sweep effect). Indeed, this is a relatively easy case in comparison, for example, to sweep optimization, where the fluid-structure-interaction is presumably of much greater importance and the relationship between HOST and elsA/HOST more non-linear.

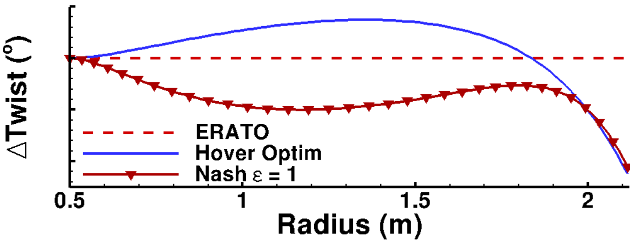


Figure 9: Variations of the twist law with respect the ERATO rotor for the initial hover optimum and the Nash Equilibrium at $\epsilon = 1$

5.5 Genetic Algorithm Comparison

Genetic algorithms are a popular strategy to treat multi-objective problems in the literature. Desirable properties of this approach are their global character and the fact that they yield an array of solutions among which the practitioner can choose (a Pareto Front). However, these advantages come at the cost of many function evaluations, which limits its direct use. In order to palliate this problem objective function evaluations are very usually carried out on a metamodel. Given that surrogate models were already available for both hover and forward-flight cases, a genetic optimization was conducted on these metamodels in order to compare the obtained results. The Non-Sorting Genetic Algorithm II, developed by Deb [33] was chosen as it is a good representation of the state-of-the-art genetic algorithms. A Python implementation of the algorithm (included in the PyOpt [34] library) was used to carry out the optimization. Therefore, the Figure of Merit was obtained with a kriging metamodel, while the multi-fidelity approach was used in the computation of the torque coefficient.

The optimization was run using 40 individuals per generation while monitoring the convergence of the solution. The determination of the Pareto Front required at least 400 evaluations for each discipline (the final results are shown for 30 generations for increased diversity of the Pareto set). The converged Pareto Set, formed by 40 individuals, was then recomputed using elsA as the evaluator in hover. Due to the good precision of the multi-fidelity metamodel, the performances were not recomputed in forward flight. However, 5 points were used to further test the results of the multi-fidelity approach, obtaining a mean error of 0.138%, which further confirmed the validity of the model. The recomputed Pareto Front is presented in Figure 10 along with the one obtained with the metamodels. The performances of the baseline ERATO rotor are presented as well in dashed line. The region where both disciplines are improved simultaneously is rather small (corresponding to the inferior left quadrant) thus illustrating the antagonism between the hover and forward flight disciplines.

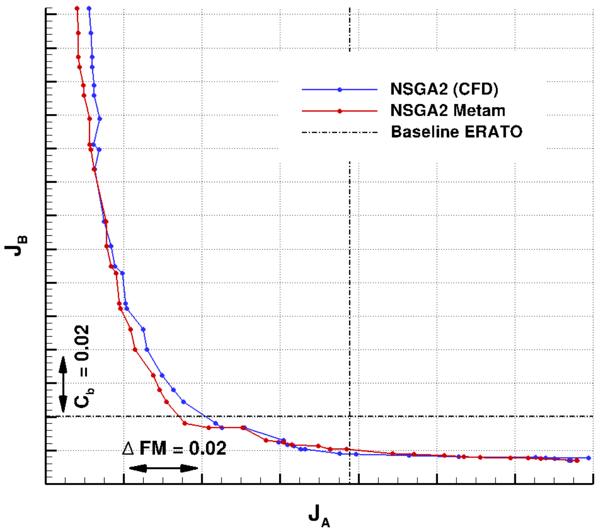


Figure 10: Comparison of the Pareto Front points obtained using a metamodel for J_A w.r.t. their CFD evaluations. The ERATO baseline performances

The results of the high-fidelity Nash game carried out on the metamodels is presented in Figure 11. As expected, the Nash game starts from a non-dominated point on the Pareto Front and the continuum of equilibria extends tangentially to the front. In this particular case the continuum of equilibria remains very close to the Pareto Front, even at $\epsilon = 1$. The verification of the Nash Equilibria with CFD computations (elsA in hover and elsA/HOST in forward flight, even if the agreement with the multi-fidelity model is already excellent) is presented next in Figure 12.

The high-fidelity Nash Equilibria dominate sections of the recomputed Pareto Front yielded by NSGA-II thus indicating that in the high-fidelity space this front is not yet converged and would require further iterations. Furthermore, the error between the Nash Equilibria obtained using metamodels and the high-fidelity values appears to be lower (at least for $\epsilon \leq 0.8$) than in the case of the NSGA-II Pareto Front. This can partly be explained by the fact that the Equilibria remain in the proximity of the initial optimum in the variable space. Indeed, by construction, the region around the initial optimum is better sampled and thus metamodel evaluations near to this initial optimum tend to be more accurate. It should be noted as well that the existence of a continuum of Equilibria implies that sufficiently close equilibrium points are close in the space of the objective functions but as well in the design space.

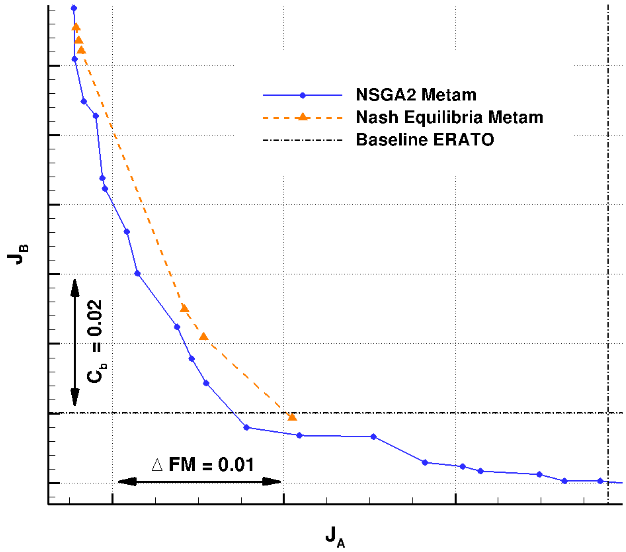


Figure 11: Pareto Front yielded by NSGA-II and Nash Equilibria obtained using metamodels for both disciplines.

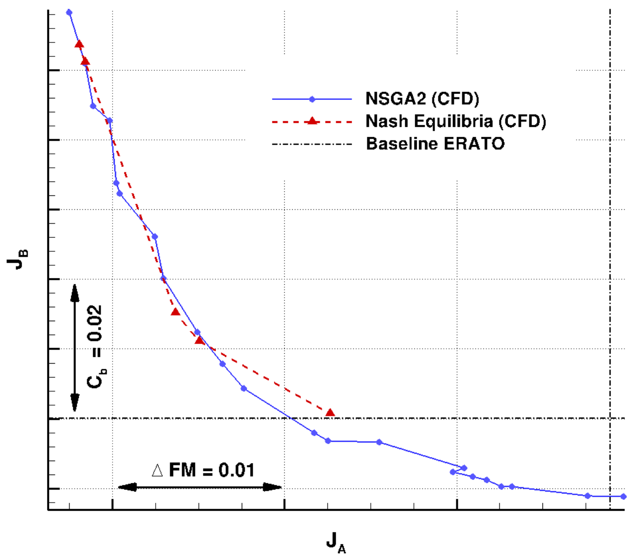


Figure 12: Pareto Front yielded by NSGA-II (recomputing J_A with CFD) and Nash Equilibria (recomputing both J_A and J_B with CFD).

The Nash Game formulation is thus validated for the high-fidelity optimization, requiring less than half function evaluations than NSGA-II for this particular case, even if only a segment of the Pareto Front is approximated and the number of evaluations on the metamodel is not a significant issue. More importantly, solutions of the Nash game remain in the vicinity of the initial optimum. In our application, for the equilibrium at $\epsilon = 1$ the torque coefficient in forward flight is improved to the level of the initial baseline ERATO rotor, while at the same time conserving gains on the

Figure of Merit of about 3.1 points (out of the approximately 6.1 points improvement obtained for the single-objective optimization in hover).

5.6 MGDA Application

Nash Games and MGDA represent complementary approaches to the multi-objective optimization problem and can be combined in competitive and cooperative optimization phases. In the case where the starting point is not on the Pareto Front, MGDA can be employed to converge to a Pareto stationary point simultaneously improving all the criteria. Subsequently, a Nash game can be carried out, obtaining a continuum of equilibria tangent to the Pareto Front that gradually departs from this front. At the end of a Nash game (or when a user-defined maximum degradation of the principal criteria is reached during the Nash Game) MGDA can be applied again, using as starting point the last Nash Equilibria, attaining the Pareto Front. The iterative alternation of cooperative and competitive phases can yield a discretization of the Pareto Front. Such an application was demonstrated by Minelli in the aero-acoustic shape optimization of a business jet [35].

In the case of the high-fidelity twist optimization presented in the previous section, the continuum of Equilibria remains very close to the Pareto Front up to $\epsilon = 1$. It could be interesting to apply MGDA at the last equilibrium point and subsequently run a new Nash game starting from the newly obtained optimum. Alternatively, in our work we opted to apply MGDA to each one of the obtained Nash Equilibria in order to investigate whether this procedure would produce better results than a genetic algorithm at an efficient cost. This was motivated by the fact that the obtained equilibria appear to be already very close to the predicted Pareto Front. Additionally, this approach allows us to take advantage of high-fidelity models to compute the gradient in hover.

Precisely, all the evaluations in hover were directly evaluated using CFD computations and the gradient of the Figure of Merit with respect to the design variables was obtained using the adjoint formulation available in elsA. In forward flight the previously presented multi-fidelity model was employed. The optimizations were limited to a maximum of 3 MGDA iterations (that is, to the computation of 3 gradients for each criterion in addition to the linesearch evaluations). The results of the optimization are presented in Figure 13, along with the NSGA-II Pareto Front and the Nash Equilibria (both recomputed with CFD).

In this case, the gradient was evaluated 11 times, and the J_A and J_B functions were called 49 and 44 times respectively, considering all the runs of the algorithm at the 5 starting points. Given the fact that the starting points are relatively close to the Pareto Front the convergence is fast, all runs converge with a maximum of 3 MGDA iterations. Furthermore, significant gains are obtained almost exclusively at the first iteration, at the end of which the candidates are already practically converged on the Pareto Front. For instance, considering only the first iteration, only 5 gradient evaluations were required along with 18 and 17 function evaluations for J_A and J_B.
Small gains in both disciplines are obtained, especially for the optimization starting at the last Nash Equilibrium. At this last point, a gain of 3.4 points of FM is retained with respect to ERATO along with a reduction of the necessary rotor-shaft of 0.16%. However, the Pareto Optimal point resulting from the last optimization appears to coincide with the Pareto Front yielded by NSGA-II (recomputed with CFD): it appears that the hover metamodel is in fact quite precise.

These results validate our implementation of MGDA combined with a high-fidelity approach. The MGDA approach appears to be quite efficient, as a single iteration already provides near optimal solutions in this case.

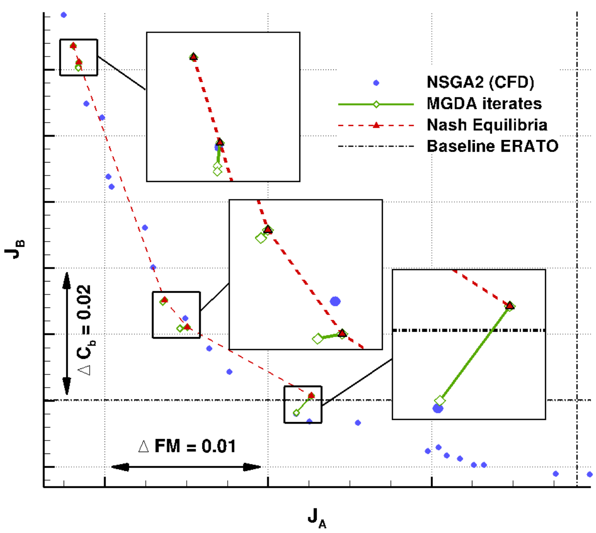


Figure 13: Results of the adjoint-assisted MGDA optimizations starting at the Nash Equilibria.

6 CONCLUSIONS

In this work, two algorithms (i.e. Nash games and MGDA) adapted to the resolution of multi-objective optimization problems were presented and applied to the optimization of the ERATO model rotor. Both algorithms are complementary in the sense that Nash games are competitive whereas MGDA is cooperative in nature. The treated optimization problem consisted in maximizing the Figure of Merit (J_A) in hover and minimizing the consumed power (or equivalently the torque coefficient value, J_B) in forward flight. Starting from an optimized configuration in hover, the competitive algorithm allowed the optimization of the forward flight performance without degrading in excess the gains already obtained in hover. This procedure has been applied to the planform shape optimization of the ERATO rotor model employing both low- and high-fidelity models for the evaluation of rotor performance in forward flight. An adjoint-based optimization in hover was thus presented in order to prepare the Nash games. Subsequently, Nash game multi-objective optimizations were successfully carried out. However, the optimization results for the cases where only sweep was considered showed that higher-fidelity models (and possibly the introduction of structural constraints) are needed in order to obtain realistic solutions.

A procedure was thus built to introduce higher-fidelity information in the optimization loop, using a multi-fidelity approach. The presented multi-fidelity framework, based in the modeling of error functions mapping evaluations from the lowest to the highest level, is able to predict very accurately the high-fidelity evaluations in the case of twist distributions. However, this approach needs further validation using more challenging geometric functions (sweep variables for instance), as the differences in the trends provided by HOST and the loose coupling of elsA/HOST will surely be more different. In this case, multiplicative corrections might perform better than the retained additive correction, as this allows for skewing (i.e. scaling in addition to rotation and translation) more adapted to the modeling of non-linearities.

The results of the Nash game involving twist distributions are similar to those obtained using a low fidelity model (HOST), which highlights the fact that HOST is able to correctly predict the high-fidelity trends in this particular case. Again, it would be interesting to explore other variables in future works. A high-fidelity optimization using a genetic algorithm (NSGA-II) was also carried out for comparison purposes. The Nash Game formulation required less than half function evaluations than NSGA-II, even if only a segment of the Pareto Front was approximated. It should be noted that the number of evaluations in this context is not an issue (as surrogate models were employed). However, the fact that the Nash game yields solutions in the vicinity of the original starting point presents an additional advantage with respect to global genetic algorithms in general, as the metamodel needs to be only locally accurate, whereas in the case of a global search the metamodel should be accurate throughout the complete search space.

A final cooperative optimization phase was finally presented using the Nash game results. The high fidelity application, using the adjoint formulation in hover and the multi-fidelity approach in forward flight demonstrates the efficiency of the algorithm, as very few evaluations are needed to obtain significant improvements of all the objective criteria. A final optimum was thus obtained, retaining a gain of 3.4 points of the Figure of Merit in hover and slightly reducing the rotor-shaft torque value in forward flight by 0.16% with respect to the baseline ERATO rotor.

COPYRIGHT STATEMENT

References

[1] Dumont, A., Le Pape, A., Peter, J., and Huberson, S., "Aerodynamic Shape Optimization of Hovering Rotors Using a Discrete Adjoint of the Reynolds-Averaged NavierStokes Equations," *Journal of the American Helicopter Society*, Vol. 56, (3), 2011, pp. 1–11.
doi: 10.4050/JAHS.56.032002

[2] Choi, S., Potsdam, M., Lee, K., Iaccarino, G., and Alonso, J. J., "Helicopter rotor design using a time-spectral and adjoint-based method," *AIAA Paper*, 2008.
doi: 10.2514/6.2008-5810

[3] Allen, C., S.Rendall, T., and Morris, A., "Computational-Fluid-Dynamics-Based Twist Optimization of Hovering Rotors," *Journal of Aircraft*, Vol. 47, (6), 2010, pp. 2075–2085.
doi: 10.2514/1.C000316

[4] Glaz, B., Friedmann, P. P., and Liu, L., "Helicopter Vibration Reduction throughout the En-

tire Flight Envelope Using Surrogate-Based Optimization," *Journal of the American Helicopter Society*, Vol. 54, (1), 2009, pp. 12007–1–12007–15. doi: doi:10.4050/JAHS.54.012007

[5] Leusink, D., Alfano, D., Cinnella, P., and Robinet, J.-C., "Aerodynamic rotor blade optimization at Eurocopter-a new way of industrial rotor blade design," 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2013.
doi: 10.2514/6.2013-779

[6] Johnson, C., Woodgate, M., and Barakos, G., "Optimising Aspects of a BERP-like Rotor Using Frequency-Domain Methods," 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2013.
doi: 10.2514/6.2013-639

[7] Massaro, A. and Benini, E., "Multi-Objective Optimization of Helicopter Airfoils Using Surrogate-Assisted Memetic Algorithms," *Journal of Aircraft*, Vol. 49, (2), 2012, pp. 375–383.

[8] Collins, K. B., Sankar, L. N., and Mavris, D. N., "Application of Low-and High-Fidelity Simulation Tools to Helicopter Rotor Blade Optimization," *Journal of the American Helicopter Society*, Vol. 58, (4), 2013, pp. 1–10.

[9] Imiela, M. and Wilke, G., "Passive Blade Optimization and Evaluation in Off-Design Conditions," 39th European Rotorcraft Forum, 2013.

[10] Wilke, G., "Multi-Objective Optimizations in Rotor Aerodynamics using Variable Fidelity Simulations," 39th European Rotorcraft Forum, 2013.

[11] Han, Z.-H. and Görtz, S., "Hierarchical kriging model for variable-fidelity surrogate modeling," *AIAA journal*, Vol. 50, (9), 2012, pp. 1885–1896.

[12] Imiela, M., "High-fidelity optimization framework for helicopter rotors," *Aerospace Science and Technology*, December 2011.
doi: 10.1016/j.ast.2011.12.011

[13] Nash, J., "Non-cooperative games," *Annals of mathematics*, Vol. 54, (2), 1951, pp. 286–295.

[14] Désidéri, J.-A., Duvigneau, R., and Habbal, A., "Multi-objective design optimization using Nash games," *Computational Intelligence in the Aerospace Sciences, AIAA Special Issue*, edited by V. Becerra and M. V. Eds., AIAA, In press.

[15] Attouch, H., Redont, P., and Soubeyran, A., "A new class of alternating proximal minimization algorithms with costs-to-move," *SIAM Journal on Optimization*, Vol. 18, (3), 2007, pp. 1061–1081.

[16] Désidéri, J.-A., "Multiple-gradient descent algorithm (MGDA) for multiobjective optimization," *Comptes Rendus Mathématique*, Vol. 350, (5), 2012, pp. 313–318.
doi: 10.1016/j.crma.2012.03.014

[17] Zerbinati, A., Désidéri, J.-A., and Duvigneau, R., "Comparison between MGDA and PAES for multi-objective optimization," Research Report 7667, INRIA, June 2011.

[18] Prieur, J. and Splettstoesser, W. R., "ERATO - An ONERA-DLR Cooperative Programme On Aeroacoustic Rotor Optimization," 25th European Rotorcraft Forum, Rome, Italy, September 1999.

[19] Roca, E., Le Pape, A., Désidéri, J.-A., Alfano, D., Costes, M., *et al.*, "Concurrent Aerodynamic Optimization of Rotor Blades Using a Nash Game Method," AHS 69th Annual Forum, 2013.

[20] Cambier, L. and Veuillot, J., "Status of the elsA CFD software for flow simulation and multidisciplinary applications," 46th AIAA Aerospace Science Meeting and Exhibit, Vol. 664, 2008.

[21] Dumont, A., *Calculs de gradients pour l'optimisation des performances aérodynamiques d'un rotor d'hélicoptère en vol stationnaire*, Ph.D. thesis, Université de Poitiers, Faculté des Sciences Fondamentales et Appliquées, April 2010.

[22] Servera, G., *Développement d'une Méthodologie de Couplage Dynamique/Aérodynamique pour les Rotors d'Hélicoptère*, Ph.D. thesis, Université d'Orléans, April 2002.

[23] Benoit, B., Dequin, A., Kampa, K., Von Grünhagen, W., Basset, P., and Gimonet, B., "Host, a General Helicopter Simulation Tool for Germany and France," American Helicopter Society 56th Annual Forum, Virginia Beach, USA, May 2000.

[24] Krige, D., "A statistical approach to some basic mine valuation problems on the Witwatersrand," *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, 1951.

[25] Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, Vol. 13, (4), 1998, pp. 455–492.

[26] Forrester, A. I., Sóbester, A., and Keane, A. J., "Multi-fidelity optimization via surrogate modelling," *Proceedings of the royal society A: mathematical, physical and engineering science*, Vol. 463, (2088), 2007, pp. 3251–3269.

[27] Eldred, M., Giunta, A., Collis, S., Alexandrov, N., and Lewis, R., "Second-order corrections for surrogate-based optimization with model hierarchies," 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, USA, August 2004.

[28] Toulmay, F., Benoit, B., and Arnaud, G., "Improvements to the aerodynamic model of the r85 helicopter rotor code: Validation and applications," 28th ISL Applied Aerodynamics Symposium, 1991.

[29] Pierre-marie, B., Heuze, O., Prasad, J. V. R., and Hamers, M., "Finite State Rotor Induced Flow Model For Interferences and Ground Effect," *Aerospace Engineering*, 2001.

[30] Bingham, G. J., "The Aerodynamic Influences of Rotor Blade Taper, Twist, Airfoils and Solidity on Hover and Forward Flight Performance." Technical report, DTIC Document, 1982.

[31] Vanderplaats, G. N., "CONMIN: A FORTRAN program for constrained function minimization," Technical report, NASA, NASA-TM-X-62282, August 1973.

[32] Adams, B., Bohnhoff, W., Dalbey, K., Eddy, J., Eldred, M., Gay, D., Haskell, K., Hough, P., and Swiler, L., "DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5.0 User's Manual," Technical report, Sandia National Laboratories, TR SAND2010-2183, December 2009.

[33] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, Vol. 6, (2), 2002, pp. 182–197.

[34] Perez, R. E., Jansen, P. W., and Martins, J. R., "pyOpt: a Python-based object-oriented framework for nonlinear constrained optimization," *Structural and Multidisciplinary Optimization*, Vol. 45, (1), 2012, pp. 101–118.

[35] Minelli, A., *Optimisation aéro-acoustique de forme d'un aéronef supersonique d'affaire*, Ph.D. thesis, Université de Nice-Sophia Antipolis, November 2013.