DOI: 10.13009/EUCASS2019-432

Use of ROM Based Methods in the Determination of Gas Turbine Engines Health Condition.

*Sánchez de León, Luis; Vega, José Manuel; Montañés, José Luis**
** ETSIAE-UPM.*
Plaza del Cardenal Cisneros, 3.
28040, Madrid (Spain).
luis.sanchezdeleon@upm.es, josemanuel.vega@upm.es, joseluis.montanes@upm.es.

*Rodrigo, José***
*** GE Power, Aeroderivative Gas Turbines Services. Conversions, Modifications and Upgrades.*
Calle Josefa Valcarcel, 26.
28027 Madrid (Spain).
jose.rodrigo@ge.com.

Abstract

Reduced Order Models (ROMs) have recently found great application within the field of aerothermodynamics, for the study of highly complex physical systems, such as the analysis of aeroelastic effects in transonic flight of modern jet fighters, heat transfer effects in separated flows, etc. Given the great computational time reduction that can be achieved by means of implementing this sort of methodologies, the aim of this paper is to assess whether ROMs could find similar usage within the engine health management system of modern turbofans. For this purpose, an engine model is created in PROOSIS®, with the ability to simulate degraded performance of the aeroengine. The model is coupled with an optimisation algorithm developed in MATLAB® to solve the inverse problem, in which the actual state of degradation of the individual components of the engine are worked out from the data gathered by the engine sensors. The capabilities of ROMs to speed up the above calculations are explored, with the final idea in mind of facilitating the development of a health management tool that can be deployed on-board, running in real-time and constantly assessing the engine degradation state, both to improve decision-making process of maintenance actions to be taken, as well as for on-the-fly update of the control system of the aeroengine based on its actual degradation state at every point during the mission. The impact that such a tool could have in both the direct and indirect operating costs of aeroengines is expected to be very relevant, especially for civil commercial aviation.

Nomenclature

BFGS: Broyden–Fletcher–Goldfarb–Shanno.
EGT: Exhaust Gas Temperature.
EKF: Extended Kalman Filter.
EPR: Engine Pressure Ratio.
GA: Genetic Algorithm.
GPA: Gas Path Analysis.
HOSVD: Higher-Order Singular Value Decomposition.
HPC: High-Pressure Compressor.
HPT: High-Pressure Turbine.
LPC: Low-Pressure Compressor.
LPT: Low-Pressure Turbine.

DOI: 10.13009/EUCASS2019-432

NH: N (rotational speed) of the High-pressure shaft.
NL: N (rotational speed) of the Low-pressure shaft.
NLGPA: Non-Linear Gas Path Analysis.
ROM: Reduced Order Model.
SI: Spark Ignition.
SQP: Sequential Quadratic Programming.
SVD: Singular Value Decomposition.
TIT: Turbine Inlet Temperature.

Symbols

N: Rotational speed (rpm).
Pt: Total pressure (Pa).
Tt: Total temperature (K).
Wc: Corrected mass flow rate (kg/s).
Wf: Fuel mass flow rate (kg/s).

1. Introduction

The operational life of the conventional commercial aircraft is getting longer and estimated currently in about 25 years. A big portion of the pressure, in terms of responsibility for the operability of the aircraft, is clearly transferred to the aircraft engine manufacturers. A similar situation occurs with gas turbines inside cogeneration plants or combined cycles.

Prognostics are directly related to the risk analysis and reparation costs due to unexpected system failures as it is exposed in [1]. Some of the economic impacts that a good embedded prognostics system could imply, per [2] from a maintenance perspective, include (but are not limited to):

1. Reduction in the workload.
2. Minimisation of reparation times.
3. Reduction of life cycle costs.

The present document addresses the detailed study, evaluation, analysis and in-depth interpretation of the enormous amount of information that can be obtained, during the monitoring of the different types of jet engines, through the readings taken of the instrumentation installed in them. The main concept that arises in this article is the obtaining of information of quality through the analysis of the measures taken from the different sensors and instrumentation that conventional jet engines such as civil 2-spool turbofan engines usually have mounted. These engines count with a numerous and variated set of sensors that log constantly the condition of the machine during its operation. Those readings are sent to the engine control system to detect faults or problems, as well as their storage for later processing and incorporation into the fleet database.

Health monitoring, performance evaluation, prognosis analysis and even regime control of aircraft engines and gas turbines, on real time, could be attempted by using adequate mathematical, numerical and computational tools such as the methods based on Reduced Order Models (ROMs) with the data retrieved on real time from the engines' instrumentation. One of its most attractive benefits is related to the resulting computational efficiency (in terms of computational time). The methods based on ROMs allow to reduce the computational cost in the determination of design cycles, where must be known the effect that, on the quality variables, have diverse parameters. A recent example of the potential of these methods applied to aeronautical engineering can be found in Benito et al. [3], where the authors apply this method to a SI engine.

Thus, it's clear these methods have been already used in the study and analysis of alternative engines. But could they be applied to the case of aircraft engines, such as 2-spool civil turbofan engines and gas turbines, which count with an amount of quality parameters and operation variables considerably higher? The present work wishes to answer this question.

DOI: 10.13009/EUCASS2019-432

The engines that will be considered for the analysis are widely used in the aviation industry nowadays: 2-spool civil turbofans (like the CFM-56 5A, GE90, etc.) that are defined, in addition to size, by design parameters such as the by-pass ratio, fan compression ratio, overall compression ratio or the turbine inlet temperature. Also, these engines are characterised by the quality parameters of their components and their respective evolutions. Among them, it's necessary to emphasise the role of the mass flow at the entrance of the compressor, in the combustion chamber or in the ducts of hot and cold flows; other quality parameters are, obviously, the efficiency of the compressors, the efficiency of turbines or the efficiency of the combustion chamber. It is convenient to keep in mind that, even in selection problems, calculations cannot be performed in a single flight condition. This makes the problem very complicated when it is necessary to make compatible the results obtained in different conditions of flight where the characteristics of lever (regulation of the regime of the engine) also play an important role.

Before arriving to the tremendous applicability of the ROMs and their validity in the quick obtaining of solutions to complex, non-linear, inverse problems, different methodologies were used, different theories were explored, and the valuable previous work of different investigators was consulted, with the clear intention of reaching the best possible result.

In [4], Ogaji et al., clearly stated that it is necessary some sort of measurements to assess failures in one engine. With the proper data gathered from the engine, it is possible to improve the maintenance plan, increasing the availability of the machine. Depending on which parameter is used to fix the power output value, the measurements to diagnostic failures in an engine could vary (here Tt4 and MW are the parameters employed). In that work, the NLGPA (Non-Linear Gas Path Analysis) was used to find out the set of instruments needed, so it's possible the optimisation when selecting the set of applicable sensors, resulting unnecessary the traditional redundancy. In [4] it was also highlighted the impact in maintenance hours that phenomena like creep or thermal fatigue would mean.

Lu et al., developed in [5] an integrated method based in non-linear on-board calculation techniques to elaborate diagnostics, working with measurements from installed sensors. Their work led to a control architecture that employed engine dual-non-linear models:

- One of them was a non-linear, adaptive, real-time performance model with an Extended Kalman Filter (EKF), to avoid lacks in accuracy.
- The other was a non-linear model of the on-board baseline. It provided the updated reference during the flight.

The work exposed in [5] established a new strategy to set up failure thresholds for sensors, as well as for noise levels. The engine model couldn't be used as a reference for the baseline forever. The authors considered some powerful simplifications as zero- dimensional flow, negligible influence of the Reynolds number or lack of combustion delays. Having a nice model for the engine and a good technique to establish the operational thresholds are key elements of the model exposed in [5].

Stamatis et al. defined the parameters and measurements to employ in [6] when monitoring an engine, to guarantee the operational safety on it. The key factors were engine configuration, available measure instruments and the required accuracy. In this work, a quick selection procedure was developed based on the Singular Value Decomposition (SVD). A value of the uncertainty in the results was given, so this method could be used to take decisions. The main objective of the study was to develop a Health Monitoring Expert System.

In [6], authors provided a very interesting method as well to evaluate changes in several engine components, to detect measure failures or perturbations, in the whole system, when introducing 1% variations in specific engine parameters.

Sarkar et al. in [7] followed a very similar procedure when defining a proper engine model. In this case, the Modular Aero-Propulsion System Simulation (C-MAPSS) developed by NASA was employed to obtain data. Then, a semantic framework for multi-sensor data interpretation and fusion was used in front of two different scenarios:

- S1: Engine progressive deterioration.
- S2: Sudden failure.

Then, it was stablished a hierarchical selection criterion based on the frequency of appearance of the different failures.

Ogaji et al. in [8] tried the diagnosis of failures in the gas path of gas turbines by means of the minimisation of differences in between observed data and simulated data. Gas Path Analysis (GPA) method was limited by the number and accuracy of the measures taken from the engine.

DOI: 10.13009/EUCASS2019-432

The GPA needed several measures, at least equal to the number of parameters considered, which would be, as usual, the efficiency and flow parameters of each component. The conclusions were somewhat short.

In reference [9], Li developed a new method for the calculation of performances of an engine, to predict and control the behaviour of the same. It was based on Newton-Raphson's Influence Coefficients Matrix and algorithm (it's used an ICM to solve by NR later). This method employed Gas Path measurements. It seemed it didn't need the maps of the components, which reduced the calculation times. However, there were 30 seconds per prediction point. Very high time for what was expected.

In [10], Najjar et al. continued with the topic, as many other researchers have done in the last decade, getting a classical optimisation problem to solve. In this sense, it was good to remember that Kurzke established in his classic paper [11], there were two possible ways to follow when looking for optimal values:

1. Parametric studies: Not useful when the number of parameters is high.
2. Numerical optimisation: Based in the gradient method.

The big issue of the techniques based in the GPA is the lack of precision caused by the very few measures available and because the errors inherent to those. The authors in [12] highlighted (and compared) different techniques, based in the GPA, that appeared sequentially since 1970, meaning a continuous increment in complexity and fusion with other methods that were appearing through the years.

On the other hand, Tumer et al. explained in [13] how a recurrent problem nowadays with the Engine Health Monitoring Systems was the transfer of a huge amount of information, after a single flight, to the ground maintenance personnel. Several commercial programs, based on expert software systems, were started in the past years to manage this outstanding amount of data. The problems they found were the big concentration of false alarms, the few valuable data obtained and the expensive systems to manage complete flight data logs. It looks managing big sets of data will be less expensive every day, it will provide a more accurate and reliable feedback in the next few years to come and, very likely, it will be the best way to assess and validate the condition in which a turbine engine is. In this sense, the management of big sets of data has been treated by different researchers in the last years, with successful results, reducing the computational time needed. Vega et al. developed, in [14], a method to create aerodynamic data bases with important savings in CPU time. In that work, model equations were replaced by a ROM based on HOSVD method that provided simplified global descriptions of multidimensional databases with enough accuracy for engineering applications.

A possible way to create big databases is characterizing statistical distributions with the attributes that real data have, as it is explained in [15], getting this way a big set of synthetic data that closely matches the characteristics of the raw data, reducing this way the cost of obtaining these data.

In commercial aircraft engines and gas turbines, the applicability of these methods will dramatically depend on the Return on Investment perceived by the end customers. Results leading to longer major maintenance intervals, longer life in components, controlled deterioration of the materials or more careful operational strategies could justify the applicability of the mentioned methods. Some other authors like in [16] introduced the concept of cost factor to determine, in a more immediate way, which anomaly should be fixed first.

It's necessary to clarify that the interest of these studies was not only for aircraft propulsion purposes. The performance, diagnostics and prognostics are concepts that are experiencing an impressive increase in terms of interest inside the aeroderivative gas turbines world and in inside heavy-duty gas turbines forums. In regard to the industrial sector, another interesting reading is the work presented by Ogbonnaya et al. in [17]. Furthermore, GE Digital is a new division created by General Electric to address the demands from operators and end customers to get real time data from their assets. This level of knowledge could allow to make sensible decisions that could prevent eventually unscheduled outages or catastrophic failures in key components inside a conventional power plant. This means the interest from gas turbine operators in the topic commented so far is increasing with the years.

As a summary of what has been exposed so far, the aim of the present work, once reviewed the literature on the matter, is to explore the potential of ROMs to speed up the calculations in solving fully non-linear inverse problems. Therefore, the main objectives will be establishing the lacks and advantages of solving linear vs. non-linear inverse problems; establishing the pros and cons of several global vs. local optimisation methods (namely: genetic algorithms vs. gradient-based methods) to solve fully non-linear inverse problems; and, finally, establish the potential of ROMs being applied to all the above, to reduce the required amount of computational time.

DOI: 10.13009/EUCASS2019-432

2. Methodology

The methodology that will be established hereafter will have the main goal, therefore, of solving the inverse problem for the performances of a given aeroengine. This means, for a known set of sensor measurements at any given operating point, to work out the actual operating regime —this is, the turbine inlet temperature (TIT)— and the degradation state of the individual turbomachinery components —which will be calculated as a percentage deviation from the nominal conditions (or “clean engine” state)— of said aeroengine at that particular operating point.

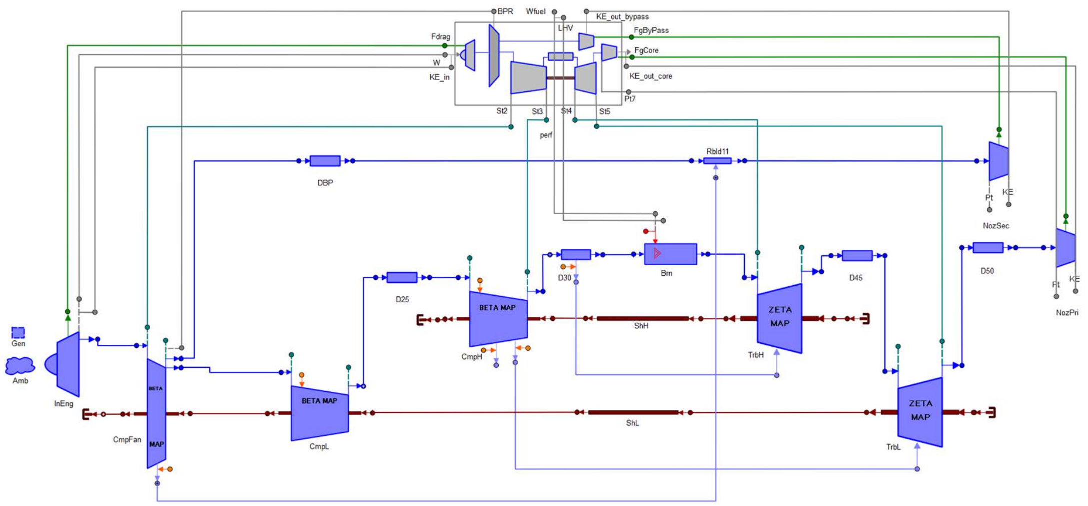

Figure 1: PROOSIS® model for the two-spool unmixed-flow high-bypass turbofan

For this study, sensor measurements will be replaced by numerical simulations of the performances of the aeroengine carried out in PROOSIS®. Once the set of sensor measurements for a given operating condition is known, it is fed into an optimiser developed in MATLAB®, which tries to minimise the quadratic error between the sensor outputs from the guessed degradation state of the aeroengine and the set of known target measurements. When sufficient agreement between the two sets is achieved, the optimisation ends, and this guessed degradation state is assumed to be the solution of the inverse problem.

2.1 Engine model

For the purposes of this research, a numerical model of a two-spool unmixed-flow turbofan was developed in PROOSIS® and aligned against publicly available data of a typical modern high-bypass turbofan (namely, the General Electric CF6-80E1A2, see [18]) at design point. PROOSIS® is a well-known state-of-the-art tool for advanced simulation of gas turbine engine performances [19], and provides the TURBO® library, which includes a complete catalogue of gas turbine components that allows the modelling and simulation of virtually any sort of imaginable gas turbine cycle [20].

Both the PROOSIS® simulation platform and the TURBO® toolkit have been widely used and extensively validated in studies related to the performances of gas turbine cycles —ranging from the typical current civil turbofan configurations to the most innovative cycle arrangements or even counter-rotating open-rotor configurations (see, for instance, [21] and [22], among many others)— which makes the aforementioned tools very suitable for the intended purpose of this research. The complete diagram of the engine being considered for this study is depicted in Figure 1.

DOI: 10.13009/EUCASS2019-432

2.2 Optimisation algorithms

As previously stated, solving the inverse problem means dealing with an undetermined system. There are 11 unknowns, namely:

- The TIT, which is an indicator of the actual operating regime of the engine.
- The degradation, both in efficiency and in corrected mass flow rate, of each of the five turbomachinery components of the engine (fan; low-pressure compressor, LPC, or booster; high-pressure compressor, HPC; high-pressure turbine, HPT; and low-pressure turbine, LPT).

For which there is only a limited set of 10 sensor measurements —i.e. equations—available. This set of sensors has been chosen in accordance with typical specifications of the CFM 56-5A turbofan, as found in [23]. The full list of variables —i.e. unknowns, the actual operating and degradation state of the engine— and data —i.e. equations, sensor measurements available for the inverse problem— are presented in Table 1.

Table 1: Equations and unknowns for the inverse problem

State variables (*unknowns*)		**Data available (*equations*)**	
Parameter	**Units**	**Sensor**	**Units**
Fan efficiency	(-)	Pt_{13}	(Pa)
Fan Wc	(kg/s)	Pt_{25}	(Pa)
LPC efficiency	(-)	Pt_3	(Pa)
LPC Wc	(kg/s)	Tt_{25}	(K)
HPC efficiency	(-)	Tt_3	(K)
HPC Wc	(kg/s)	Tt_{45}	(K)
HPT efficiency	(-)	Tt_5	(K)
HPT Wc	(kg/s)	NL	(rpm)
LPT efficiency	(-)	NH	(rpm)
LPT Wc	(kg/s)	Wf	(kg/s)
TIT	(K)		

Note that it has been chosen to leave the TIT as an unknown for this inverse problem, meaning that the actual operating regime of the engine is indeed unknown beforehand. It must be stated that, typically, the TIT (or the actual net thrust level) of an aeroengine is worked out by means of indirect measurements (such as the EGT, the EPR, etc.) for control purposes. Once degradation starts taking place, the levels of degradation of the different components of the aeroengine will affect how these indirect measurements relate to the TIT, and the actual TIT may indeed be unknown. In this way, the presented methodology opens also the possibility for new control strategies for aeroengines: if the inverse problem can be solved quickly enough (by means of ROMs, for instance) so that real-time calculations can be achieved, it could be possible to work out, from the set of available sensor measurements at any given point of any flight, the actual TIT required to meet the specified performances with the current state of degradation of the engine (the more degraded the engine, the hotter it will have to be run, in order to achieve the same levels of net thrust that a "clean" engine would provide).

In order to deal with the lack of equations to solve the inverse problem, a multi-point strategy can be made use of. By making the only hypothesis of the degradation levels of the turbomachinery components being constant in all operating conditions for at least a moderately short period of time (the slow degradation of the engine means that the variations in these degradation parameters will only be noticeable, typically, across several flights), data can be gathered from (at least) two different operating points within the same flight (for instance, the start of the take off and the beginning of the cruise phase, very close in time), so that both sets of sensor measurements (i.e. 20 equations) will be made compatible with a single state of degradation of the engine for two different operating regimes (i.e. 10 degradation unknowns + 2 TIT unknowns).

DOI: 10.13009/EUCASS2019-432

The inverse problem, posed this way, requires no a priori information to be fed into the model: no previous knowledge of the TIT at which the engine is running (or the net thrust provided by the engine), so that the “baseline” case for the nominal sensor measurements could be calculated in order to assess deviations of the actual sensor measurements from these nominal conditions, is required whatsoever. All the information needed is actually contained within the sets of sensor measurements provided at (at least) a couple of different operating points, and this amount of information is just enough (20 equations, 12 unknowns, as previously stated) to work out the whole state of the engine in both operating conditions.

MATLAB® has been chosen as the developing tool to implement the solver for the aforementioned inverse problem, being one of the main advantages of this MATLAB® + PROOSIS® simulation platform the easiness for coupling both tools (so that MATLAB® can be used as the “solving platform”, that will make calls to PROOSIS®, the “engine-simulation platform”) thanks to the native MATLAB® libraries that, for this purpose, PROOSIS® includes within its default installation packages.

The fact that, when considering a multi-point strategy for the inverse problem, there are more equations than there are unknowns —and especially if sensor uncertainties are accounted for— makes the system of equations for the inverse problem incompatible, indeed. Therefore, an optimising strategy that tries to minimise the quadratic errors between the sets of sensor outputs that the guessed operating and degradation state of the engine would produce and the sets of target sensor measurements gathered “in-flight”, becomes necessary. For this task, several MATLAB® optimisers have been tested, which will be briefly described hereafter, while their pros and cons will be established in the following section.

2.2.1 Genetic Algorithm

The first optimiser tested is the genetic algorithm. It is indeed a global optimisation method; this is, if properly set up (with appropriate fractions of crossover, for convergence, and mutation, for thorough exploration of the search space), and given enough time, it will find the global optimum of any objective function within a given domain, without getting trapped in other local optima that may appear in this domain.

In addition to this, it does not require the evaluation of the derivatives of the objective function (sometimes very costly to evaluate, even more so when this involves making calls to an external solver such as PROOSIS®), so the algorithm is somewhat sped up by this fact. However, the lack of information of the gradient of the objective function, along with the randomness of the search, makes this algorithm inherently very slow, especially to reach convergence in the final stages of the optimisation, when refinement of the most promising candidates must be accomplished to ensure that an optimum is actually found.

Moreover, since the genetic algorithm optimiser does not evaluate the derivatives of the objective function, and hence has no information on the Jacobian matrix of the objective function, it cannot guarantee that any solution found by this method is an optimum at all (not even a local optimum), not even when convergence has been reached.

2.2.2 Gradient-based Method

Gradient-based methods rely on calculating the Jacobian matrix at each candidate point, to gather information about the gradient and hence predict where better solutions (in the proximity of said candidate) may appear. In particular, a sequential quadratic programming (SQP) method was chosen for this study. The algorithm solves a quadratic sub-problem at each iteration, for which knowing the Hessian matrix (or an approximation) is necessary. In this particular case, a quasi-Newton method which uses the BFGS formula to update the estimate of the Hessian was chosen.

Convergence, therefore, is much faster than with the genetic algorithm, since it does not perform random (but guided, instead) search. However, the fact that relies only on the gradient of the objective function to predict where better solutions could lie, also means that it will only search for local optima, since the method has no means to escape from a function basin to search in further areas of the solution space.

To avoid this issue, enough exploration of the search space can be ensured by means of implementing a multi-start strategy: if enough initial points are explored (and each of them is locally optimised by means of this SQP method), the chances of the whole algorithm missing the global optimum solution become really slim.

DOI: 10.13009/EUCASS2019-432

However, the inherent cost to this strategy is that it greatly increases the amount of computational time required to perform the optimisation: since each initial candidate must be locally optimised, the more initial points chosen for exploration, the better the chances of the multi-start algorithm finding the best solution, but also the more time it requires.

2.2.3 Linearized inverse problem

To assess in the fairest comparison the increase in calculation speed that linearizing the inverse problem can provide, the multi-start algorithm is kept the same, with the same multi-point strategy, as well as the same objective function to minimise the quadratic errors. Model calls to PROOSIS® are replaced, however, by a matrix of linearized coefficients (or, more precisely, one matrix per operating point) for the direct problem.

Replacing model calls to an external solver by matrix and vector products and sums speeds up the calculations immensely but has one inherent major drawback: since linearization can only work well when changes in the independent variables (when compared with the "baseline" case) are small, the linearized inverse problem requires some a priori information to be fed into the model. Namely, the TIT of each operating condition must be known (or, at the very least, a good approximation of this value, so that the actual operating condition can be linearized in the surrounds of this known approximation) in order to calculate the matrices of linearized coefficients at each operating condition as deviations from the "baseline" case in each of those conditions.

2.2.4 Reduced Order Models

The main tool to solve the direct problem via ROMs is a combination of higher-order singular value decomposition (HOSVD) of the fully non-linear problem, with a series of one-dimensional interpolations. This methodology has been already used successfully in different contexts (see, for instance, [3] and [24]). HOSVD [25] is an extension to tensors (namely, multiarray, containing multidimensional data) of standard SVD [26].

The resulting ROM requires a fairly large number of computations using the full PROOSIS® model to be properly set up, but these computations are performed offline and only once. The online computations, instead, are extremely fast, since they only need a series of algebraic operations and interpolations. The accuracy, on the other hand, is comparable to that of the full model. It must be noted that this ROM is fully non-linear. In other words, it does not need any linearization of the problem at hand.

3. Results and discussion

3.1 Genetic Algorithm

Three degradation cases are initially chosen to test the capabilities of the genetic algorithm to solve the inverse problem and work out the actual value of all the degradation parameters in each case, namely:

1. Moderate degradation levels in both the fan efficiency and fan Wc, whilst keeping all other degradation parameters "clean".
2. Moderate degradation levels in the core components (this is, both the HPC and HPT), degrading both their efficiency and Wc simultaneously, whilst keeping all other degradation parameters "clean".
3. Wide range of degradation levels across the board: every single component of the engine is degraded at the same time (with no "clean" parameters whatsoever), and the degradation levels of each parameter will take values ranging from relatively low to very high, to really put to test the capabilities of this method to solve the inverse problem in the most demanding (and relatively unlikely) situations. In particular, heavier degradation is considered within the hot segment of the engine.

The performances of the genetic algorithm dealing with these three cases were exceptionally good —reaching convergence within tolerances well below 1.E-04 in the residuals for all the three test cases— albeit extremely slow —taking up to 4 days to complete the calculations for a single run on a modern i7 desktop PC.

DOI: 10.13009/EUCASS2019-432

The convergence graph for the calculations of case 3 (which expectedly showed the slowest convergence of all the three test cases considered) mentioned above, for instance, is presented in Figure 2, where it can be seen the norm of the residuals of the inverse problem plotted against millions of function evaluations. The fact that each function evaluation is essentially a model call to an “external” engine-simulation platform (namely PROOSIS®) makes these function evaluations very costly, and this is indeed where most of the time required by the genetic algorithm is spent.

It is also worth noticing the exponential increase in the computational time required to achieve convergence when the demanded accuracy becomes more stringent: a reduction of the norm of the residuals from 1.E-02 to 1.E-03 can be achieved in roughly 2.5 million function evaluations, while further reducing the norm of the residuals from 1.E-03 to 1.E-04 requires approximately 5.5 million additional function evaluations (this is, more than a two-fold increase). This showcases the inherent issue with the genetic algorithms: although the initial stages of each run are very efficient, and the most promising region (where the global optimum is most likely to be found) is usually identified very quickly, at the later stages, refining that initial search to achieve higher accuracies, becomes a very slow process due to the random nature of the search process, which has to explore the surrounds of each promising candidate without any information that can be exploited to speed up the search.

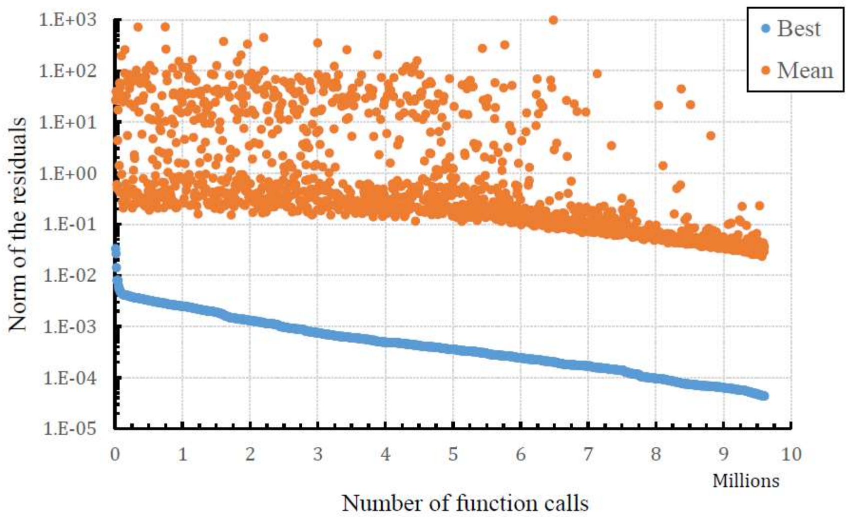

Figure 2: Convergence graph for the inverse problem solved by means of the genetic algorithm —norm of the residuals vs. millions of function evaluations (test case 3)

Just to give an idea of the accuracy achieved by this method in solving the inverse problem, the final value of the norm of the residuals for the aforementioned case 3, according to Figure 2 and just after roughly 9.6 million function evaluations, is equal to 4.34E-05, which translates into relative errors below 0.5% in all calculated degradation parameters. These relative errors for the whole set of ten degradation parameters (as per Table 1) are represented in Figure 3. As shown, the greatest inaccuracies appear when trying to calculate the 7th, 9th and 10th parameters; which correspond to the HPT efficiency degradation, the LPT efficiency degradation and the LPT Wc degradation, respectively.

DOI: 10.13009/EUCASS2019-432

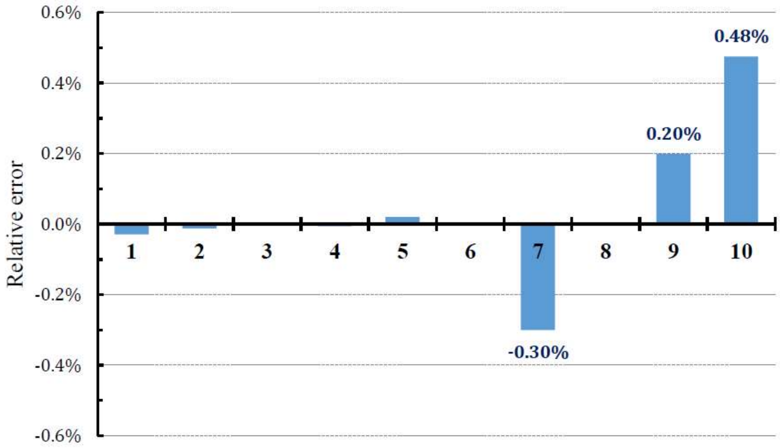

Figure 3: Relative errors on the whole set of calculated degradation parameters after 9.6 million function evaluations by the genetic algorithm (test case 3)

This fact will have an impact on the implementation of the ROMs for the inverse problem later on: since these three parameters, for which the solution incurs in the greatest inaccuracies, represent those which will be tougher to predict (based on the available data gathered from the engine sensors), they will therefore require a more dense sampling (i.e. increased "resolution" of the mesh) for the initial tensor with which the ROM (by means of a HOSVD, applied to said tensor, plus an interpolation strategy) will be set up. Higher resolution of the mesh means that a greater number of "modes" (i.e. "information") will be available for the HOSVD, hence the ROM will be able to capture finer details on the impact of these parameters on the degraded performances of the engine, and thus a solution with a higher degree of fidelity may be found for the inverse problem.

3.2 Gradient-based Method

The three test cases considered above are used to verify the performances of the gradient-based SQP optimiser method. In addition to these, two more test cases (hereafter referred to as cases 4 and 5) are defined, too, whereby two additional scenarios with a wide range of degradation levels across the board (similarly to the aforementioned case 3) are considered. These two additional test cases, which are included within this study for the mere purpose of increasing the variety of scenarios in which the SQP solver is being tested, are defined as follows:

4. High degradation levels for both the HPC efficiency and HPC Wc. Low to moderate values of degradation level for every other engine component (both in efficiency and in Wc).
5. High degradation levels for both the fan efficiency and fan Wc. Low to moderate values of degradation level for every other engine component (both in efficiency and in Wc).

The accuracy achieved by the SQP optimiser in all the five cases considered is remarkable, resulting in relative errors in the calculated degradation parameters an order of magnitude lower than those achieved by the genetic algorithm. As a broad comparison of their relative capabilities to solve the inverse problem being considered herein, Figure 4 provides the relative errors in the whole set of the ten calculated degradation parameters, after convergence has been reached, for both the genetic algorithm (GA) and the SQP optimiser methods in test case 3.

DOI: 10.13009/EUCASS2019-432

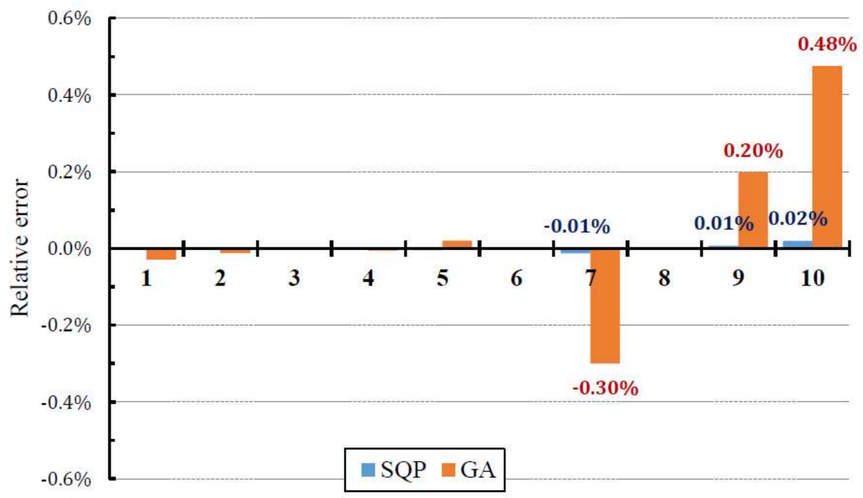

Figure 4: Relative errors on the whole set of calculated degradation parameters after convergence — comparison between the GA and the SQP optimiser methods (test case 3)

Once again, it is highlighted the struggle of the GA to refine the search around the global optimum and find the best solution within the vicinities of a promising candidate, a task in which gradient-based methods excel. On the other hand, the fact that the gradient-based methods require the calculation of the derivatives of the objective function (or, at least, an estimate by means of finite difference formulation), necessarily imposes on the SQP optimiser method a limitation on the maximum achievable accuracy: given that the estimate of the derivatives will be performed via calls to an external solver (namely, PROOSIS®) which, as such, will have a given convergence tolerance, will inevitably make the objective function exhibit some sort of "ripple" effect, so that the finite differences must be taken in large enough intervals so that this "ripple" effect does not interact with the calculations of the derivatives themselves, thus unavoidably losing accuracy on the estimates of the derivatives.

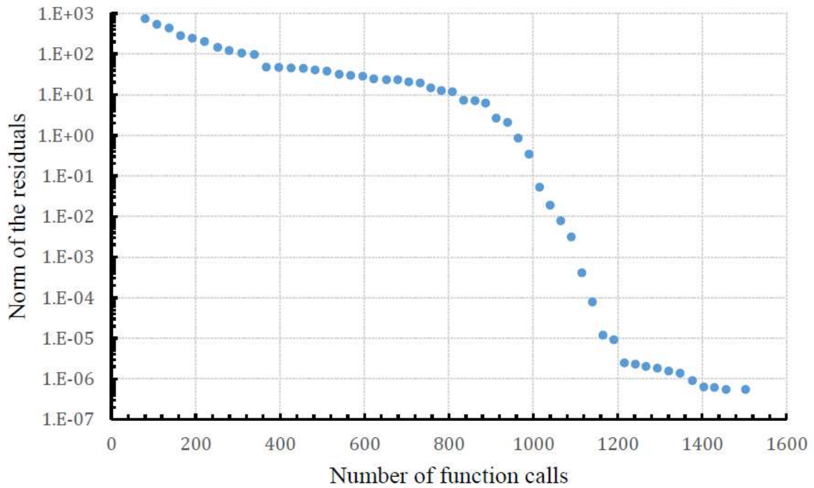

Figure 5: Convergence graph for the inverse problem solved by means of the SQP optimiser —norm of the residuals vs. number of function evaluations (test case 3)

As for the performances of the SQP optimiser goes, convergence is reached extremely quickly when compared with the genetic algorithm. A lot fewer function evaluations are required by gradient-based methods, thanks to the exploitation of the information contained within the derivatives of the function.

DOI: 10.13009/EUCASS2019-432

Figure 5 shows the convergence graph for the SQP method applied to the aforementioned case 3 (so that fair comparison can be made against the convergence graph shown in Figure 2 for the GA), where it can be seen that convergence is reached, with a final value of 5.47E-07 for the norm of the residuals, in barely above 1500 function evaluations. This represents three orders of magnitude fewer evaluations than the GA required, and explains on itself the reduction in computational time observed: only about 5–6 minutes are required for the SQP optimiser to converge.

Still, Figure 5 shows the convergence graph for the SQP optimiser when the initial value tried for the iterative process is such that it actually converges to the global minimum of the objective function. However, if not a priori information is available to be fed into the model and guarantee that the initial guess is good enough to ensure convergence to the global minimum, most of the times, the local optimum that a gradient-based method will find from an arbitrarily-chosen initial guess will not be the intended global optimum. As previously stated, this can be sorted out by simply implementing a multi-start strategy, but the associated cost is to increase the required computational time with each new initial guess that is explored by this means.

For the inverse problem being treated in this research, a value of around 200 initial guesses (randomly chosen from the whole solution space) was found out to be a good number to ensure enough exploration of the solution space so that the algorithm would never miss the global minimum. This being the case, the computational time required by the multi-start SQP optimiser was no longer 5–6 minutes, but rather around 15–20 hours (depending on how close the randomly-chosen initial guesses are from their local minima, this time can vary substantially). Nevertheless, this is still a great time reduction when compared with the performances of the genetic algorithm.

3.2.1 Sensor uncertainties

In order to ensure that the above methodology is robust, from a mathematical point of view, it is paramount to account for the uncertainties that the sensor measurements gathered in-flight unavoidably will carry. Some preliminary studies have been carried out in this direction, which will be summarised herein.

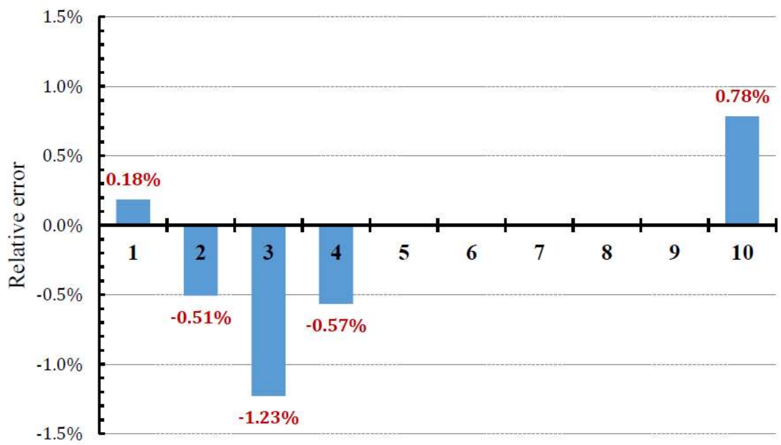

Figure 6: Relative errors on the whole set of calculated degradation parameters after convergence — SQP optimiser with ±1% uncertainty in all sensor measurements (test case 1)

For starters, all sensor measurements gathered from the two different flight conditions are affected by a random ±1% deviation (i.e. uncertainty), and the inverse problem is then solved, by means of the same multi-start, multi-point SQP optimiser, once again. Figure 6 shows, for the simplest test case 1, the relative errors in the set of calculated degradation parameters of the best solution to said problem found by the SQP method.

Notice that for the test case 1, as previously defined, only the fan efficiency and the fan Wc have been degraded (i.e. according to Table 1, parameters no. 1 and no. 2), and both of these are accurately predicted (with relative errors that are smaller than the 1% uncertainty of the sensor measurements provided to the optimiser).

DOI: 10.13009/EUCASS2019-432

However, non-negligible levels of degradation appear too in parameters no. 3, 4, and 10, which should be zero. This miss-identification of the degraded parameters is associated with the uncertainty of the sensor measurements, and hence unavoidable. The mathematical method can still be considered robust, since at least the errors found in the solution are of the same order of the "errors" found in the data provided to the solver.

This miss-identification becomes worse the greater the number of parameters are degraded simultaneously (it is already troublesome for the test case 2, and makes the solution of the test case 3 extremely inaccurate), but its effect can be mitigated by providing the solver with additional (redundant) "information": rather than taking sensor measurements from only two operating points, more data could be gathered from different flight conditions (during the whole mission/flight of an engine, the amount of points available for this multi-point algorithm is almost unlimited) to increase the number of residuals the solver has to minimise, thus reducing the uncertainty of the sensor measurements (as long as these are truly random and un-biased) by forcing them to verify as closely as possible the "constitutive equations" of the aeroengine.

3.3 Linearized inverse problem

The merits of solving the fully non-linear inverse problem vs. solving the linearized inverse problem will be assessed by carrying out a comparison in which the multi-start strategy, along with the multi-point algorithm, and even the objective function will be kept the same in both cases. Simply, in the latter case, model calls to PROOSIS® will be replaced by a matrix of linearized coefficients, whose columns are calculated as the "sensitivity vectors" (i.e. response of the system to a predefined 1% variation) associated to each degradation parameter, independently. This way, and assuming that linearization is indeed possible (in general, this will be true, as long as the deviations from the "baseline" case, in which the matrix of linearized coefficients has been calculated, are small), the variation in sensor outputs for a given degradation state of the engine can be calculated by merely multiplying the matrix of linearized coefficients by the column vector of degradation levels.

Replacing calls to an external solver by products of matrices and vectors incredibly speeds up the calculations, so that the same optimisation carried out with the fully non-linear methodology in around 20 hours can be completed by means of linearization in under 30 seconds. Again, speed comes at a price: linearization of the inverse problem inevitably leads to an accuracy loss in most of the cases. Only when the deviations from the "baseline" case are very small, the hypotheses made are verified and the linearized solution yields good results. In particular, it must be true that the system response to a linear combination of degradation parameters can be expressed as the same linear combination of the "modes" associated to a unitary degradation along each parameter. There are two main sources of accuracy loss, namely:

- Linear response: if degradation along a single parameter is considered, a k-fold relative variation in the degradation level will yield the same k-fold relative variation in the system response.
- Superposition: if degradation along several parameters is considered simultaneously, the overall system response can be expressed as the sum of the system responses to the degradation along each parameter independently.

If any of the above conditions are not met, the linearized methodology will yield poor results. To showcase this, two separated studies have been carried out. In the first one, degradation of a single parameter is considered and, beginning from a unitary value, the degradation level will be increased. In the second one, unitary degradations only will be considered and, commencing from degradation along a single parameter, the number of degraded parameters will be increased.

Figure 7 shows the perils associated to excessive extrapolation: if the actual degradation of any parameter is much higher than the unitary "reference" value, the effects of the non-linearities within the problem will have an impact on the accuracy of the predictions made by the linearized solver. It is also interesting to note that the greatest losses in accuracy do not necessarily have to happen for the degraded parameter, but rather small deviations can appear in the calculations of some of the "clean" parameters, that will be predicted as "slightly degraded" by the linearized solver.

Similarly, Figure 8 shows how the linearized solver fails to capture the interactions between different "modes": superposition of several degraded parameters at the same time cannot be truly expressed as the sum of the system responses to the degradation on each parameter independently, but instead second-order effects will take place that modify the overall system response, and these must be retained and accounted for if a high degree of accuracy is sought.

DOI: 10.13009/EUCASS2019-432

All in all, this behaviour makes the linearized inverse problem only useful for the analysis of a very narrow set of degradation states of the engine, failing to make good predictions for most cases in which the genetic algorithm or the SQP optimiser successfully calculated the degradation levels of every single component.

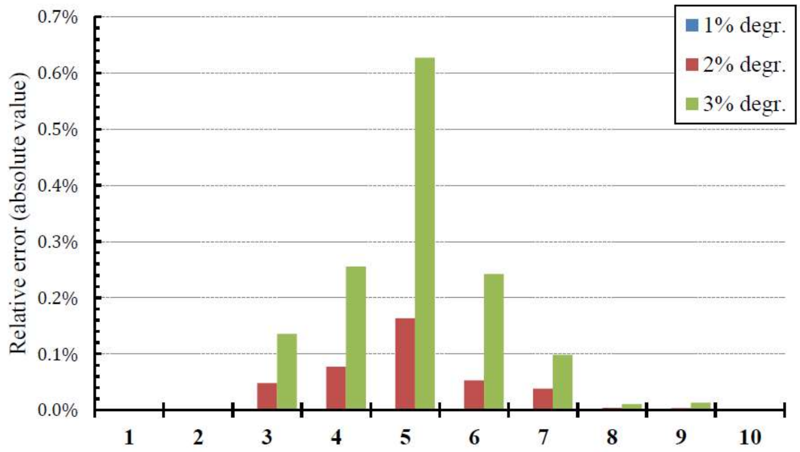

Figure 7: Relative errors on the whole set of calculated degradation parameters against increasing values of the degradation of the 3rd parameter (namely, HPC efficiency)

In sight of these results, it is straightforward to understand why the linearized solver works reasonably well for the test case 1 mentioned above (at least when the degradation levels are kept low, only minor accuracy losses are noticeable), starts to struggle for the test case 2 (even if the degradation levels are kept low, accuracy begins to drop and, more importantly, identification of the degraded parameters fails, for the solver predicts non-negligible levels of degradation for some of the "clean" parameters), and completely fails in any of the test cases 3, 4, 5 (throwing results that have little to no resemblance at all with the actual degradation state of the engine, see Figure 9).

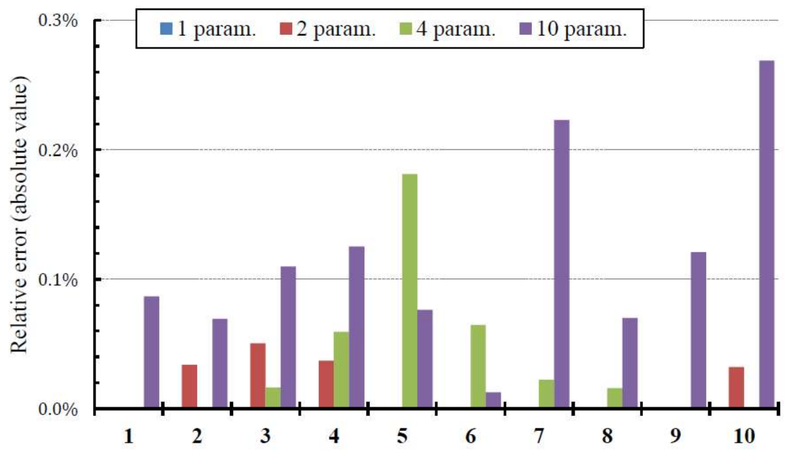

Figure 8: Relative errors on the whole set of calculated degradation parameters against an increasing number of actually degraded parameters

The fact that test case 1 is somewhat solvable by means of the linearized methodology (accuracy is not great, but at least the identification of the degraded parameters is correctly made), but test case 2, on the other hand, is not (the identification of the degraded parameter fails), already gives some clues on the shape of the matrix of linearized coefficients: even though the matrix is non-singular, the effective rank of the matrix must be between 2 and 4, most likely.

DOI: 10.13009/EUCASS2019-432

This is, a matrix that is able to solve a problem of rank 2 (test case 1 has 2 degraded parameters) but fails to solve a problem of rank 4 (test case 2 has 4 degraded parameters) must not provide enough information to solve the latter, and thus the effective rank of the matrix will most likely be lower than 4, and probably barely over 2. In other words, some of the "modes" that compose the columns of the matrix of linearized coefficients are almost linearly dependent on one another, and hence there is not enough information within the linearized matrix to be able to solve problems of a higher rank.

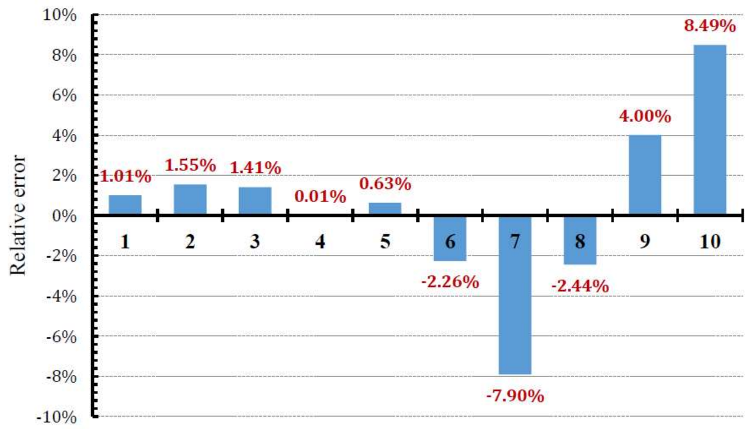

Figure 9: Relative errors on the whole set of calculated degradation parameters after convergence by the linearized solver (test case 3)

3.4 Reduced Order Models

The main goal is, thus, to achieve accuracies as close as possible to those of the fully non-linear gradient-based method, but in computational times of the order of those typical of the linearized solver.

To this end, a ROM based on a combination of HOSVD plus interpolation, similar to that developed in [3] and [24] has been constructed from a dataset constructed with the full PROOSIS® engine model. This dataset is computed offline, in a pre-process that may be fairly computationally expensive, but it is only performed once.

Using this ROM, a gradient-like method (namely, MATLAB® SQP) is used to minimise the same objective function as in the fully non-linear method described in Section 3.2. In other words, the online computations are completely similar to those in Section 3.2, except that the full engine model is replaced by the ROM. The attained accuracies are comparable to those reported above, in Section 3.2, but the required CPU time for each computation is much smaller, of the order of 10 CPU seconds, even using a non-optimised software. Thus, the online computations based on the ROM are suitable for real-time monitoring of the engine. These computations correspond to work in progress and will be presented elsewhere.

4. Conclusions and further work

In summary, the potential of ROMs to solve the inverse problem of the health monitoring of an aeroengine has been shown. As stated, their main feature is their capability to replicate a fully non-linear model of said aeroengine, with negligible deviations, but performing the calculations in just a fraction of the computational time, showcasing speeds close to those of the linearised inverse problems. This opens the field for further research on the topic, that may facilitate the on-board implementation of this sort of tools for real-time, accurate, health-monitoring systems.

As opposed to the fully non-linear methodology, the drawbacks of linearized modelling have been showcased, especially when the number of degraded parameters increases, given that both non-linearities (particularly if extrapolation takes place) as well as "non-linear superposition of degradation modes" effects, as discussed previously, result in unacceptable errors for most of the cases analysed.

DOI: 10.13009/EUCASS2019-432

Genetic algorithms are among the most robust optimisers available for the purposes of solving the inverse problem of an aeroengine degradation state but must be discarded due to their inherent extremely low speed, which makes them unfeasible for any sort of intended real-time calculations.

Gradient-based methods, on the other hand, have proven to be as effective as the genetic algorithms in finding the global minima of any system of equations when use is made of the multi-start strategy and enough initial conditions are explored, can be made equally robust from a mathematical standpoint if enough information is provided to the solver (i.e. different operating conditions at which sensor measurements are gathered) to be able to deal with sensor measurement uncertainties, and, in addition to that, can truly benefit from the speed increase that the usage of ROMs provide (as opposed to having to deal with a full-engine model).

4.1 Work in progress

Firstly, as previously stated, ROM simulations are still work in progress, and will be presented, once completed, at a later stage. The main goal is to be able to "refine" the number of modes along each degradation parameter that are necessary to retain the expected level of accuracy of the fully non-linear model, at the same time as reducing this number of modes to the minimum to speed up the calculations. Finding the best possible trade-off between accuracy and speed is the main task that should be accomplished before taking any further steps.

Secondly, uncertainties must be dealt with. So far, preliminary studies have been carried out to show that, upon imposing a random uncertainty of 1% in the sensor measurements provided to the optimiser, this uncertainty is propagated within the same order of magnitude in the solution of the inverse problem (and not amplified), which suggests that the methodology is robust enough, from the mathematical point of view, to be implemented. However, the inaccuracies in the solution (and the miss-identification of the degraded parameters) become unacceptable once the number of degraded parameters starts growing, and further steps will have to be taken in this direction to assess if these effects can be mitigated by providing the solver with sensor measurements gathered at more than just two flight conditions. The more points that are used as data for the optimiser, the longer it will take to find the optimal solution (the greater the number of model calls), so there is, once again, a fine trade-off to be found here.

The potential of ROMs is not merely to increase the computational speed of non-linear algorithms but can have further benefits and applications. Among others, there are mathematical methods that would be able to detect and isolate sensor failures on-the-fly, which would make the ROM solver even more robust than its fully non-linear counterpart; at the same time that it could be used to detect and identify sudden variations in just a few degradation parameters (as long as the assumption is made that not too many of them can suffer sudden variations simultaneously), so that the application of these methods could reach not only the calculations of the slow and steady degradation of all engine components, but also the quick and reliable diagnosis of failures and events that would suddenly affect engine performances, allowing for potentially better and quicker in-flight decision making.

Acknowledgements

The authors would like to sincerely thank EA International and, in particular, A. Barrasa, P. Cobas, F. Rodríguez, D. Castaño and the whole PROOSIS® development team, for their support in the use of said aeroengine modelling software and the licenses provided for the Universidad Politécnica de Madrid.

References

[1] Amin, M., A. Majid, R. K. Wassan and A. A. Mokhtar. 2014. Gas Turbine Risk Assessment Based on Different Repair Assumptions. *Journal of Applied Sciences.* Vol. 14, no. 17: 1966-1971.
[2] Greitzer, F. L., C. J. Hostick, R. E. Rhoads and M. Keeney. 2001. Determining how to do prognostics, and then determining what to do with it. *AUTOTESTCON Proceedings, IEEE Systems Readiness Technology Conference*. 780-792.
[3] Benito, N., J. R. Arias, A. Velázquez and J. M. Vega. 2011. Real time performance improvement of engineering control units via high order singular value decomposition: application to a SI engine. *Control Eng. Practice*. Vol. 19: 1315-1327.

DOI: 10.13009/EUCASS2019-432

[4] Ogaji, S. O., S. Sampath, R. Singh and S. D. Probert. 2002. Parameter selection for diagnosing a gas-turbine's performance-deterioration. *Applied Energy*. Vol. 73, no. 1: 25–46.
[5] Lu, F, Y. Chen, J. Huang, D. Zhang and N. Liu. 2014. An integrated nonlinear model-based approach to gas turbine engine sensor fault diagnostics. *Proc. IMechE Part G: J. Aerospace Engineering.* Vol. 228, no. 11: 2007-2021.
[6] Stamatis, A., K. Mathioudakis and K. Papailiou. 1991. Optimal Measurements and Health Indices Selection for Gas Turbine Performance Status and Fault Diagnosis. *ASME 1991 International Gas Turbine and Aeroengine Congress and Exposition, Orlando, Florida, USA, June 3–6, 1991*.
[7] Sarkar, S., S. Sarkar, K. Murkherjee, A. Ray and A. Srivastav. 2013. Multi-sensor information fusion for fault detection in aircraft gas turbine engines. *Proc. IMechE Part G: J. Aerospace Engineering*. Vol. 227, no. 12: 1988-2001.
[8] Ogaji, S. O., S. Sampath, L. Marinai, R. Singh and S. D. Probert. 2005. Evolution strategy for gas-turbine fault-diagnoses. Applied Energy. Vol. 81, no. 2: 222–230.
[9] Li, Y. 2015. Aero Gas Turbine Flight Performance Estimation Using Engine Gas Path Measurements. *Journal of Propulsion and Power*. Vol. 31, no. 3: 851-860.
[10] Najjar, Y. S. and S. F. Al-Sharif. 2006. Thermodynamic optimization of the turbofan cycle. *Aircraft Engineering and Aerospace Technology*. Vol. 78, no. 6: 467-480.
[11] Kurzke, J. 1999. Gas Turbine Cycle Design Methodology: A Comparison of Parameter Variation with Numerical Optimization. *J. Eng. Gas Turbines Power*. Vol. 121, no. 1: 6-11.
[12] Marinai, L., D. Probert and R. Singh. 2004. Prospects for aero gas-turbine diagnostics: a review. *Applied Energy*. Vol. 79, no. 1: 109-126.
[13] Tumer, I. and A. Bajwa. 1999. A survey of aircraft engine health monitoring systems. *35th Joint Propulsion Conference and Exhibit, AIAA*. 1-8.
[14] Lorente, L. S., J. M. Vega and A. Velázquez. 2008. Generation of Aerodynamic Databases Using High-Order Singular Value Decomposition. *Journal of Aircraft*. Vol. 45, no. 5: 1779-1788.
[15] Eklund, N. H. 2006. Using Synthetic Data to Train an Accurate Real-World Fault Detection System. *IMACS Multiconference on Computational Engineering in Systems Applications*. 483-488.
[16] Verda, V. 2004. Thermoeconomic Analysis and Diagnosis of Energy Utility Systems. From Diagnosis to Prognosis. *Int. J. Thermodynamics*. Vol. 7, no. 2: 73-83.
[17] Ogbonnaya, E. A. and K. Theophilus-Johnson. 2011. Optimizing Gas Turbine Rotor Shaft Fault Detection, Identification and Analysis for Effective Condition Monitoring. *Journal of Emerging Trends in Engineering and Applied Sciences*. Vol. 2, no. 1: 11-17.
[18] Igie, U., M. Goiricelaya, D. Nalianda and O. Minervino. 2016. Aero engine compressor fouling effects for short- and long-haul missions. *Proc. IMechE Part G: J. Aerospace Engineering*. Vol. 230, no. 7: 1312-1324.
[19] EA INTERNACIONAL, PROOSIS, PRopulsion Object-Oriented SImulation Software [Online]. Available: http://www.ecosimpro.com/products/proosis/. [Accessed Jan 2019].
[20] EA INTERNACIONAL, TURBO, Gas Turbine Performance Simulation [Online]. Available: http://www.ecosimpro.com/products/turbo/. [Accessed Jan 2019].
[21] Camilleri, W., E. Anselmi, V. Sethi, P. Laskaridis, A. Rolt and P. Cobas. 2015. Performance characteristics and optimisation of a geared intercooled reversed flow core engine. *Proc. IMechE Part G: J. Aerospace Engineering*. Vol. 229, no. 2: 269-279.
[22] Bellocq, P., I. Garmendia, V. Sethi, A. Patin, S. Capodanno and F. Rodriguez Lucas. 2016. Multidisciplinary Assessment of the Control of the Propellers of a Pusher Geared Open Rotor—Part II: Impact on Fuel Consumption, Engine Weight, Certification Noise, and NOx Emissions. *Journal of Engineering for Gas Turbines and Power- Transactions of the ASME*. Vol. 138.
[23] LUFTHANSA TECHNICAL TRAINING. 1999. Training Manual A319 / A320 / A321, Engine CFM56-5A, ATA 73-20.
[24] De Lucas, S., J. M. Vega and A. Velázquez. 2013. Surrogate model for viscous drag in aircraft empennage conceptual design. *Aerosp. Sci. Tech*. 31: 99-107.
[25] Kolda, T. G. and B. W. Bader. 2009. Tensor decompositions and applications, *SIAM* Rev. 51: 455-500.
[26] Golub, G. H. and C. F. Van Loan. 1996. Matrix Computations, Baltimore, MD: John Hopkins University Press.