MULTI-DISCIPLINARY ANALYSIS AND OPTIMIZATION JAVA TOOL FOR AIRCRAFT DESIGN

Vittorio Trifari*, Manuela Ruocco*, Vincenzo Cusati*, Fabrizio Nicolosi*, Agostino De Marco*
***Department of Industrial Engineering (DII) University of Naples Federico II**
Via Claudio 21, 80125 Napoli – ITALY
Ph: +39 0817683583 Fax: +39 081 624609

Abstract

The first stages of the aircraft design process require to carry out multi-disciplinary analyses as fast as possible, and with a certain grade of accuracy. During the conceptual and the preliminary phases, the goal is to search for the design that best fulfils the requirements.
This work presents a Java framework, named JPAD, developed at the University of Naples Federico II by the Design of Aircraft and Flight technologies research group (DAF) to perform multi-disciplinary analysis and optimization of transport aircraft.
This paper describes all the JPAD capabilities, focusing on the sensitivity analyses and optimization modules. At the end, a case study concerning the optimization of a regional turboprop aircraft model similar to the well-known ATR72 will be presented.

1 Introduction

Nowadays the preliminary design phase of an aircraft has becoming very challenging due to ever more demanding requirements. The goal of first design stages is to search for the configuration that best fit all requirements, among the results of a great number of multi-disciplinary analyses, as fast as possible, and with a certain grade of accuracy.

The continuous improvement of computer calculation capabilities over years has allowed the growth of a large family of software dedicated to aircraft preliminary design activities concerning also multi-disciplinary analyses, and optimizations [1], [2], [3], [4], [5], [6], [7].

A key feature that most of this software provide, is the possibility to parametrically define both aircraft components and complete aircraft configuration leading to a very fast and intuitive definition process of a generic aircraft model. With software and computer hardware currently available in aerospace industry, the design process has become very effective and employs, a very sophisticated and highly optimized chain of calculation tools [1], [2], [3], [4], [5], [6], [7].

A modern preliminary aircraft design tool should be characterized by a certain level of accuracy and reliability (although using fast and simple semi-empirical procedures), the capability to perform multidisciplinary analyses and optimizations, and reasonably short computational times for a complete analysis process. Because of the relevance of aircraft performance, noise and emissions levels, maintenance and operative costs in the commercial success of a transport aircraft, a modern software framework must be developed aiming at a multidisciplinary approach. Another important feature lies in the user-friendliness of the software allowing users to interact with the framework in an easy, fast, and efficient way.

To ensure longevity and to enrich future exploitation capabilities, the possibility to include in the software multiple fidelity analysis methodologies or to easily implement new semi-empirical models, is of primary importance.

One remarkable example is given by the possibility to easily generate and export the aircraft configuration CAD model in one or more standard formats and to execute high-fidelity analyses with external tools (i.e. Computational

Fluid Dynamics CFD or Finite Element Method FEM solvers).

Major aerospace companies have developed their own codes to estimate aero-structural characteristics and aircraft stability in the conceptual/preliminary design phase, as well as universities which have developed various codes for educational and research purposes like SUAVE [1] or CEASIOM [2].

Several commercial aeronautical software are available to perform a variety of aircraft aerodynamic and performance calculations. Among them, industry standard commercial software for preliminary aircraft analysis may be represented by AAA [3], RDS [4], Piano [5] and PACELAB [6]. This latter has quickly become one of the most used software for aircraft preliminary design phases thanks to a very smart software architecture, the possibility to perform fast multi-disciplinary sensitivity studies and the integration of dedicated analysis modules focused on systems architecture (also including hybrid-electric propulsion) and cabin configuration layout. Another comprehensive program that uses a multi-disciplinary approach for transport aircraft is FLIGHT [7], specialized in the prediction and modelling of fixed-wing aircraft performance.

The Design of Aircraft and Flight technologies (DAF) [1] research group of the University of Naples Federico II have been working since 2005 to the development of software and frameworks for aircraft design and they are expert users of most of the above-mentioned software, reaching a mature vision of what kind of features are expected from a modern multi-disciplinary analysis and optimization software. A first remarkable example is the ADAS software [8], developed mainly for teaching purposes, which has also been used in combination with CEASIOM [9].

Since 2013 the DAF group has been involved in the development on a complex open-source Java library named Java toolchain of Programs for Aircraft Design (JPAD), built as a modular framework, gathering all the lessons learned in the past few decades of tool development for aircraft design [10], [11]. This library is designed as a fast, reliable and user-friendly computational aid for aircraft designers in the conceptual and preliminary design phases.

In recent years, the DAF group has gained knowledge and experience in developing, testing and validating several approaches and methodologies concerning aircraft design field of application. For instance, an improved approach regarding the vertical tail plane design and sizing was accomplished by means of numerical and experimental analyses [12], [13], [14]. This methodology was also applied to size the vertical tail plane of a new twin-engine commuter aircraft [15], [16], then was validated through wind tunnel tests [17]. Past research activities have, also, focused on aerodynamic derivatives estimation on light and General Aviation (GA) aircraft [18]. Another methodology, regarding the design of the fuselage and the prediction of its aerodynamic characteristics, was developed through CFD-RANS calculations performed on several fuselage geometries suited for regional transport aircraft [19]. The research group have developed a deep experience as far as aircraft design [20], [21], [22] is concerned also for innovative technologies [23], such as for design and aerodynamic analysis of airfoil and high lift devices [24] and performance estimation of light aircraft with morphing devices [25]. Most of these knowledges have been included in the JPAD library using dedicated external databases.

2 JPAD overview

The JPAD library has been conceived to be used in an industrial environment across conceptual and preliminary design phases. In these phases a lot of different configurations should be analyzed, so the software has been developed to provide results in a short period of time; this need often requires relying on semi-empirical methods. A comprehensive study of the methods available in literature has been firstly carried out to improve the accuracy of the results: each method (produced in-house or drawn from literature) has been tested against experimental data so that statistical quantities (e.g., standard deviation) could be estimated

[1] www.daf.unina.it

either to find the best method currently available or to make a merger of different methods.

JPAD is completely written in Java; this programming language is a general-purpose, concurrent, class based and object-oriented. One design goal of the Java language is the portability, which means that programs written for the Java platform must run similarly on any combination of hardware and operating system with adequate runtime support.

To achieve an understandable input file organization, a considerable study has been done. The result is an input structure composed by different interconnected XML files aiming to provide the maximum level of flexibility to the user both in the generation of an aircraft model, both in the execution of one or more analyses.

In Fig. 1 the entire structure of the software is schematized. It is possible to clearly note that there are two main blocks: input and core. The input block is defined by two main parts: aircraft and analyses definitions. The first one defines a parametric aircraft model using a main file (Aircraft.xml) which collects all the components positions and the related xml file name (i.e. fuselage.xml, vtail.xml, and so on) which contains all geometrical data. This structure allows to generate different aircraft, or different configurations of the same model, by simply combining different components allowing to easily perform comparisons between these latter. The second one defines all necessary data for each analysis present inside the Core module.

The software is capable to automatically generate and export the aircraft CAD model in several formats (i.e. STEP, BRep, etc.). This can be easily imported in external tools like CAD, CFD or FEM suites. The CAD model is conceived to allow also the automatic creation of complex elements such as wing tips and fairings. The possibility to generate a CAD model gives to the user an immediate feedback about the data provided to the application and allows for an accurate estimation of the wet surface of each component. CAD models are created in JPAD using the Open CASCADE library [26], an open source software development kit, written in C++ and released by Open Cascade SAS.

Besides the input, the second main block shown in Fig. 1, is the Core which manages all the available analyses. This contains several independent modules, that deals with following application fields.

- **Weights:** estimates the aircraft weight breakdown starting from a first guess maximum take-off weight and some mission profile specifications. It evaluates each aircraft component mass using a mix of several semi-empirical equations [27], [28], [29], [30], [31], [32], [33], [34]. The module is designed to allow users to choose for each component one calculation method or an averaged weight estimation using all the available methodologies. In addition, the user is provided with a calibration module which allows to manage each component estimated mass value.
- **Balance:** estimates the center of gravity position related to each weight condition and draws the balance diagram. The module allows also to manage each aircraft system and equipment group positions to better estimate the center of gravity excursion. User can assign each group position or let the module to estimate them assuming typical positions as provided in [27].
- **Aerodynamics and Stability:** the aerodynamics module estimates all the aerodynamic characteristics concerning lift, drag and moments coefficients at different operating conditions both for the complete aircraft and each component (wing, tails, fuselage and nacelles). Whereas the stability module gives useful data about longitudinal and lateral-directional static stability of the whole aircraft considering non-linearity effects as well (i.e. pendular stability, non-linear downwash gradient, etc.).
- **Performance:** evaluates the most important aircraft performance producing several useful reports and charts such as the Payload-Range diagram, a detailed mission profile analysis report, the cruise flight envelope, climb and ground performance simulations [35] as well as the Specific Air Range (SAR) chart. JPAD implements a smart simulation-based approach to analyze both the complete mission profile, both each mission phase.
- **Costs:** estimates the Direct Operating Costs (D.O.C.) breakdown. A detailed explanation of this module is provided in [11].

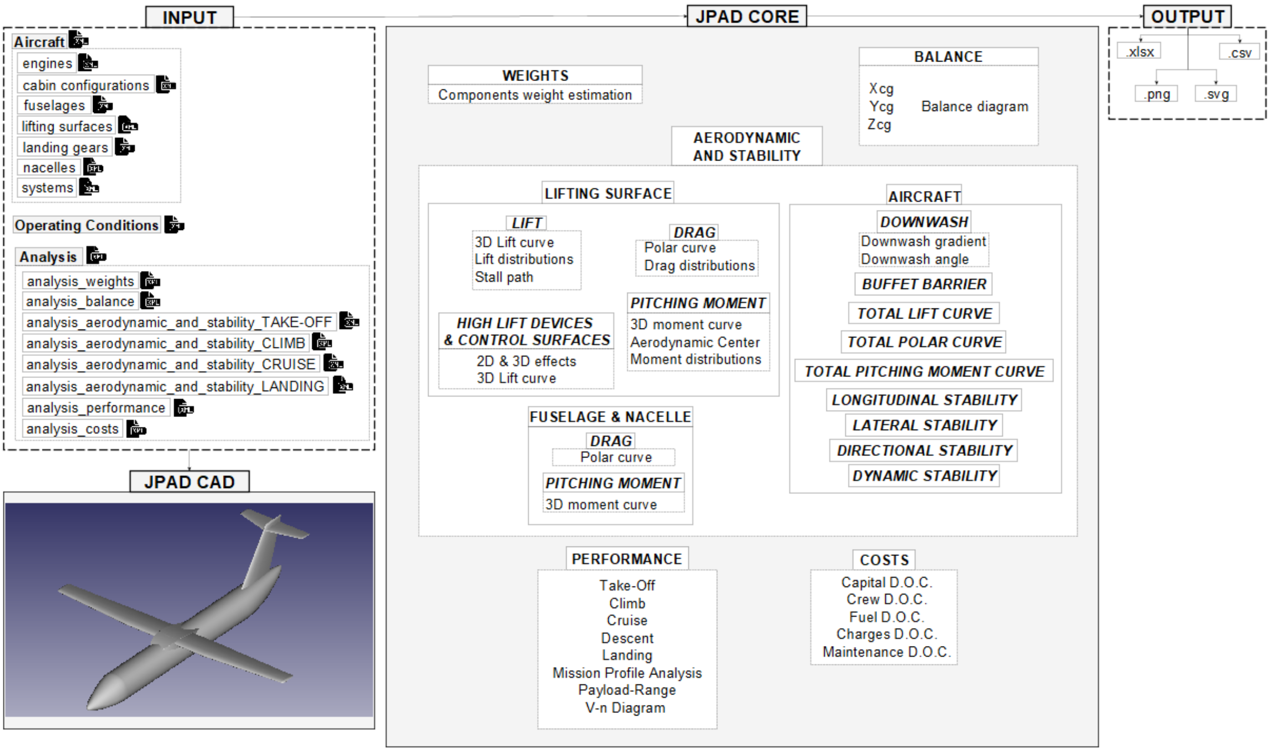

Fig. 1. JPAD – Main features

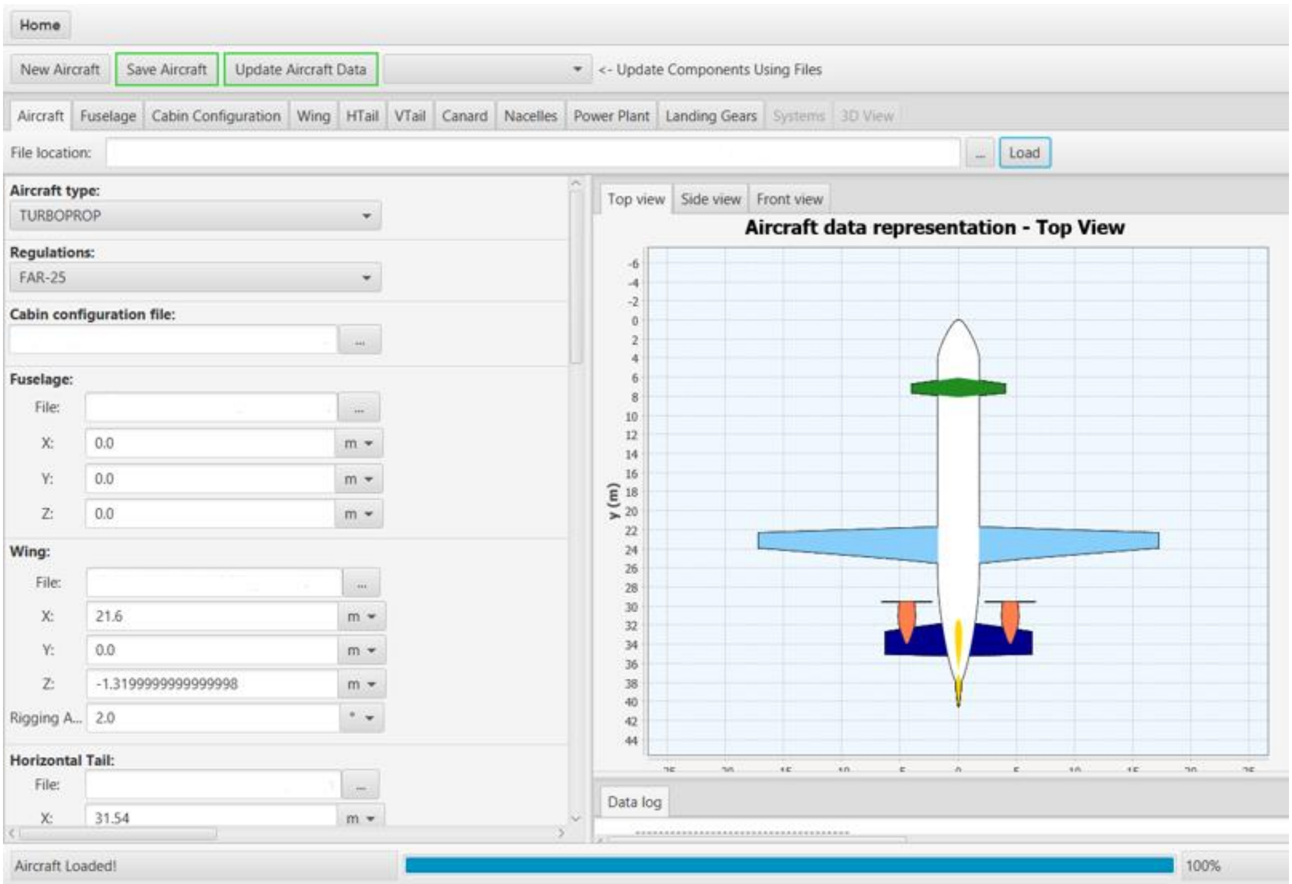

Fig. 2. JPADCommander – Input Manager example

To enhance the user-friendliness of the library, JPAD has been provided with a Graphical User Interface (GUI), named *JPADCommander*, completely designed using the JavaFX [36] library together with a JavaFX-based development tool named SceneBuilder [37]. The main goal of the GUI is to guide the user throughout all the JPAD functionalities starting from the aircraft model generation up to the visualization and the management of the analyses results. An example of the *JPADCommander* aircraft creation process is shown in Fig. 2. Furthermore, the CAD model generated by the JPAD library can be can be represented in the GUI input manager thanks to the *MeshView* library of JavaFX which allows to convert it in the native JavaFX format.

JPAD allows to obtain different kind of output: analyses report in Excel format, charts in .png and .svg format and charts points in .csv format. Using Excel file, the comparison between two or more aircraft (or simply between slightly different configurations of the same aircraft) is easier and more efficient.

Using specific native Java classes (such as *java.lang.Runtime* and *java.lang.Process*), this programming language allows JPAD to interconnect directly to external tools that can be launched in batch mode. At this time, the JPAD library is provided with launchers for the following external software: AVL(a program for aerodynamic and flight dynamics analysis) [38], Digital Datcom (a static stability and dynamic-derivative characteristics calculator) [39], STAR-CCM+ (a CFD analysis software) [40] and JSBSim[2] (a multi-platform, general purpose object-oriented Flight Dynamics Model (FDM) written in C++) [41]. The interface with JSBSim is currently under development and expects to interconnect the JPAD library with the JSBSim software using the CPACS data format. The use of this kind aircraft modelling format comes from a collaboration of the DAF research group with the German aerospace research institute DLR, within the European H2020 project named AGILE[3] [42].

3 Sensitivity Analysis and Optimization Module

The JPAD library is designed as an interconnection of different modules each one dedicated to a specific task. In the previous section an overview of the Core module has been provided.

However, since the main goal of the JPAD library is to carry out multi-disciplinary analyses and optimizations (MDAO), the focus will be on two modules dedicated to sensitivity analysis and multi-objective optimization. These use all the Core features of the JPAD library and allow users to easily analyze a large number of different aircraft models searching for one or more optimum configurations.

A first attempt to solve MDAO problems expects to entrust all the analyses to an expert well versed in all disciplines to reduce communications and organization problems. This approach, named Monolithic Design (MD), has been widely used to carry out conceptual design phases in the past and is suitable only for simple problems or when approximate results are acceptable.

Nowadays a single expert is unable to monitor a complex process, like the design of a complete aircraft, and new multidisciplinary design techniques are required. To manage all disciplines, a way could be to define a process in which the aircraft is designed thanks to the collaboration of a group of different experts (one per discipline). This is the Collaborative Design (CD) approach.

The third generation of MDAO approaches, core of the European AGILE project, is a direct evolution of the previous one and is called Collaborative Remote Design (CRD). This involves a group of experts geographically located in different parts of the world that can communicate and exchange their own tools or results through a remote server connection. In this way is possible to take advantage of the knowledge of several aerospace research centers or companies in each certain discipline. A case study concerning the CRD approach is provided in [43].

[2] https://github.com/JSBSim-Team/jsbsim

[3] https://www.agile-project.eu/

The two JPAD modules together with all the Core features, define a closed MDAO environment which concerns the MD approach. However, the possibility given by JPAD to be potentially interfaced with external tools and to use standalone modules makes this library suitable also for modern MDAO approaches (CD, CRD). As a result, the optimization module has been widely used to solve MDAO problems belonging both to MD [44] both to CRD [45].

The JPAD sensitivity analysis module allows users to have access to all the possible input variable needed to define an aircraft model thus they can specify which one have to be changed and within which interval. As shown in Fig. 3, this module creates different aircraft equal to the number of combination of all the design parameter array elements, (full factorial combination).

Each of these is then analyzed using a combination of *JPADCore* modules represented in Fig. 1. The possibility to invoke individually each analysis module, or even a single output parameter calculation method, plays a key role in reduction of the computational time required for the whole calculation process. In addition, thanks to the possibility to easily manage multiple parallel threads, the user can further reduce the amount of computational time running more than one analysis simultaneously.

To carry out a complete analysis cycle, the JPAD library uses a combination of its analysis modules as shown in Fig. 4.

The analysis starts with a first estimation of the amount of fuel needed for the specified mission. Then a balance analysis is carried out to determine the center of gravity excursion. For each center of gravity, the aerodynamic and stability module estimates the trimmed drag polar in all the following flight condition: take-off, climb, cruise and landing. Finally, the performance module uses these data to make a detailed simulation of the initial mission profile estimating a new amount of fuel needed to cover the mission. Thus, an iterative process is carried out until the first estimated fuel mass is equal to the one calculated by the mission profile analysis.

Once the preliminary iterative loop has converged, the JPAD library reads from file all the analysis that the user wants to perform and invokes only the required analysis modules.

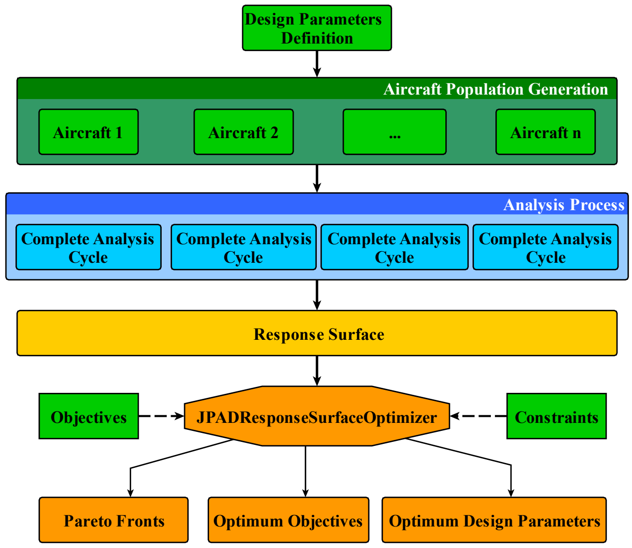

Fig. 3. JPAD – Sensitivity analysis and optimization module flowchart.

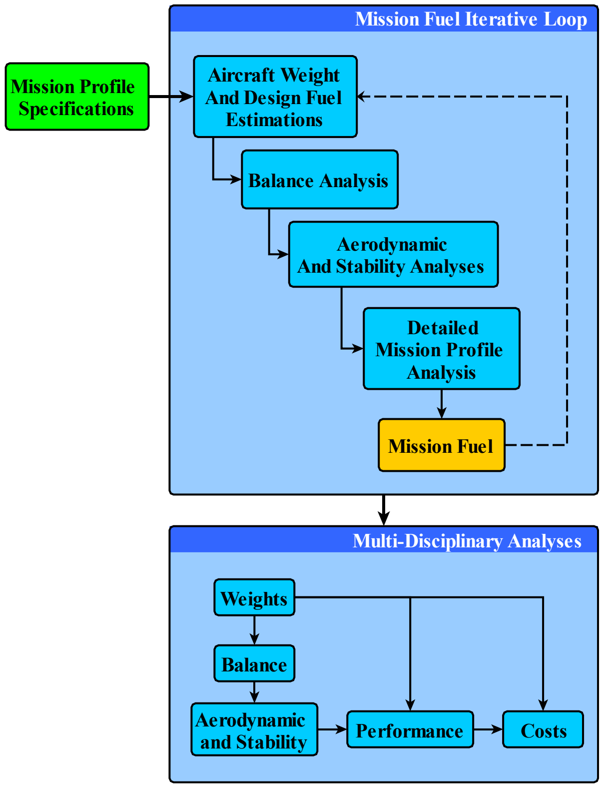

Fig. 4. JPAD – Complete analysis cycle flowchart

At the end of each analysis cycle JPAD stores in an external dataset all the output variable that the user has decided to monitor defining, this way, a cloud of solution points. As shown in Fig. 3, all these data are passed to a standalone optimization module.

The core of the JPAD optimization module is based on well-known metaheuristics algorithms, among which the most commonly used are Genetic Algorithms (GA) and the Particle Swarm Optimization (PSO) algorithms. The use of metaheuristics algorithms allows to easily manage complex optimization problems with a reduced amount of calculations if compared with classical deterministic algorithms (i.e. gradient based like Newton-Raphson). As explained in [46], gradient-based algorithms show some issues with discontinuous objective functions due to the use of derivatives to find the optimum solution. On the other hand, metaheuristics algorithms do not rely on derivatives but only on objective function values, thus they can easily manage complex and even discontinuous response surfaces.

JPAD is provided with all the current state-of-the-art metaheuristic optimization algorithms thanks to the use of a dedicated external library named MOEA Framework, a free and open source Java library for Multi-Objective Evolutionary Algorithms (MOEAs)[4]. Although the optimization module can use every algorithm provided by this library, two of them have been chosen due to results quality and computational efforts: ε-NSGAII (Non-dominated Sorting Genetic Algorithm) and OMOPSO (Optimized Multi-Objective Particle Swarm Optimization) algorithms are used.

The ε-NSGA-II algorithm is an extension of NSGA-II that uses an ε-dominance [48] archive and randomized restart to enhance search and find a diverse set of Pareto optimal solutions [47]. Full details of this algorithm are given in [49].

OMOPSO is a multi-objective particle swarm optimization algorithm that includes an ε-dominance [48] archive to discover a diverse set of Pareto optimal solutions [47]. The algorithm was originally introduced in [50].

Using both these algorithms, the JPAD optimization module can easily solve complex MDAO problems reading all the following required instructions from a dedicated configuration file.

1. the number of design variables, objectives and constraints;
2. whether or not an objective has to be minimized or maximized;
3. upper and lower boundaries
4. constraints values and the type of violating condition (i.e. outside an interval, bigger than a prescribed value, etc.);
5. which algorithm must be used.

Together with this information, the complete set of points of the response surface must be passed to the module as a *.csv* file. Before the optimization process, all response surface points are interpolated using n-dimensional cubic spline functions. At the end of the process, charts of all possible combinations of Pareto fronts as well as a series of *.csv* files (one per algorithm) containing the complete set optimum values of design variables and objectives are produced.

4 Case Study: ATR72

In this section a case study concerning the multi-disciplinary optimization of a regional turboprop aircraft model, similar to the well-known ATR72, is presented. A case study concerning a complete analysis cycle of this aircraft has been already shown in [11] validating, this way, JPAD calculation for such a regional platform.

In the spirit of reproducible research philosophy, authors provide the full set of configuration files necessary to reproduce this case study on the JPAD GitHub website[5].

This case study was conducted on a quadcore Intel Core i7-7700 with 32Gb of RAM. All calculation times have been estimated assuming this hardware configuration.

The constrained multi-objective optimization problem is stated as follow (see Table 1): the case study objective functions are the Operating Empty Weight (OEW), the Block Fuel (BF) for the design mission of 800nm together with the related Direct Operating Costs (DOC) and Global Warming Potential (GWP), defined accordingly Ruijgrok and Van Paassen [51]. Optimization constraints are related to ATR72 ground performance and longitudinal stability

[4] http://moeaframework.org/

[5] https://github.com/Aircraft-Design-UniNa/jpad

requirements in the form of Static Stability Margin (SSM).

Starting from geometrical data acquired from online public data and the 3-view of the ATR72, the JPAD library has produced a parametric model of the aircraft, which CAD model is shown in Fig. 5.

The MDAO problem is stated as summarized in Table 1. The variables for this application are the main wing planform and position parameters:

- Wing position (body reference frame) in meters - X_{LEw};
- Wing aspect ratio - AR_w;
- Wing thickness ratio - $(t/c)_w$;
- Kink station position with respect to the wing semispan - η_k.

Table 1. Multi-objective optimization problem definition.

	$\boldsymbol{Min}$:
Objective functions:	$\boldsymbol{f_1 = DOC}$ $\boldsymbol{f_2 = BF}$ $\boldsymbol{f_3 = GWP}$ $\boldsymbol{f_4 = OEW}$
	$w.r.t$:
Constraints:	$4.0 \geq SSM \geq 6.0 \ (\%MAC)$ $TOFL \leq 1315\ m$ $LNFL \leq 1169\ m$
	$by\ varying$:
Variables:	$X_{LE_w} \in [10.2; 10.8]\ with\ 13\ values$ $AR_w \in [9.5; 15.0]\ with\ 23\ values$ $\left(\frac{t}{c}\right)_w \in [0.15; 0.21]\ with\ 7\ values$ $\eta_k \in [0.25; 0.335]\ with\ 6\ values$

A full factorial Design Of Experiment (DOE) has been carried out to define a Response Surface (RS) suitable for numerical optimization. More than 12500 aircraft configurations have been analyzed, stored and made available for the optimization problem with the JPAD library as described in the previous section with a total calculation time of about 15 hours. Authors are currently working on software performance optimization aiming to reduce this time.

For the optimization process both ε-NSGA-II and OMOPSO algorithms have been used leading to the Pareto fronts shown in Fig. 6, Fig. 7 and Fig. 8.

Fig. 5. Comparison between the ATR72 (above) and the JPAD parametric CAD model (below).

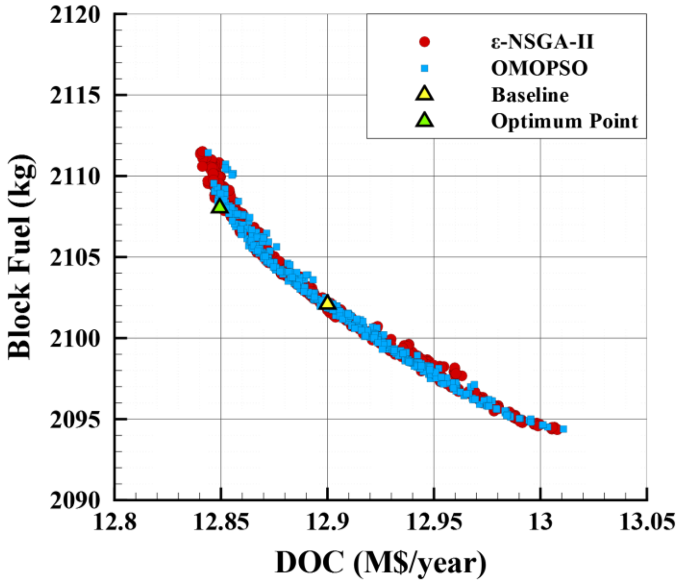

Fig. 6. Block Fuel vs DOC Pareto front.

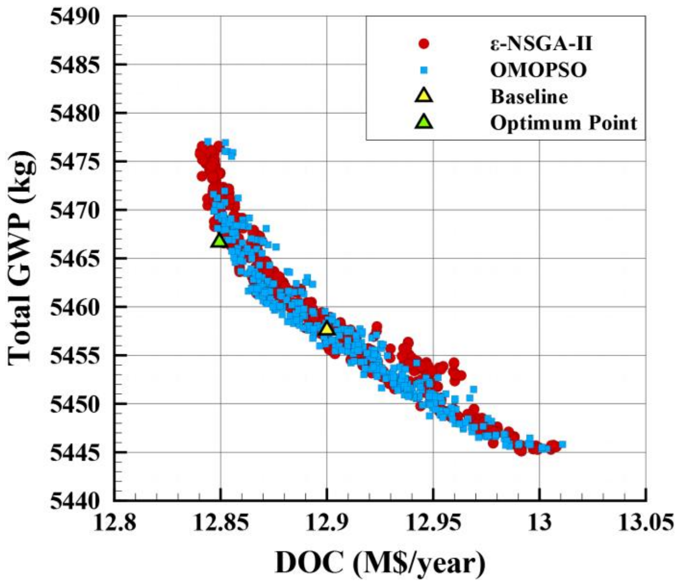

Fig. 7. Total GWP vs DOC Pareto front.

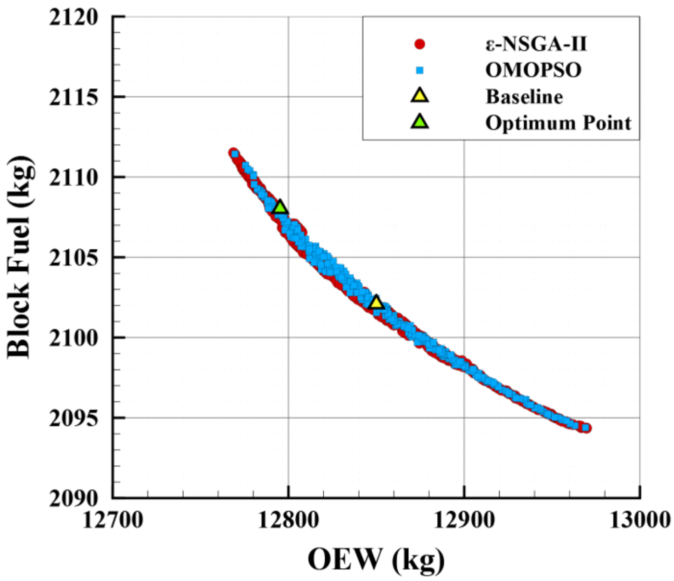

Fig. 8. Block Fuel vs OEW Pareto front.

As shown in Table 2, the OMOPSO algorithm provided less optimum solutions in less time with respect to ε-NSGA-II with a similar global Pareto efficiency parameter.

Table 2. Optimization algorithms comparison.

	Time (s)	No. Solutions
ε-NSGA-II	44	589
OMOPSO	32	358

The driving factor of the multi-objective optimization has been the DOC.

Although the baseline aircraft was already on the Pareto front (see Fig. 6, Fig. 7 and Fig. 8), a DOC and OEW reduction (50000$/year and 55kg respectively) could be achieved in spite of a slightly increase (less than 10kg) of BF and GWP values. The OEW reduction leads to a lower cruise lifting coefficient which provides a lower aerodynamic efficiency. This is the reason why a reduction of the OEW provides an increase in BF and, consequentially, in GWP. Furthermore, a lower OEW is also the main reason behind the DOC reduction.

However, as shown in Table 3, these variations are quite limited due to a great effect given by the assigned constraints (see Table 1) which dramatically reduces the optimization research domain.

Finally, Fig. 9 and Table 3 provide a comparison between baseline and optimized aircraft models.

Table 3. Baseline and Optimized aircraft comparison

	Baseline ATR72	Optimized ATR72	Difference
X_{Lew} (m)	10.63	10.51	-1.13%
AR	12.00	11.00	-8.33%
$(t/c)_w$ (%)	18.00	17.7	-1.66%
η_k (%)	29.79	33.35	+11.95%
Wing Area (m^2)	61.00	61.00	0.00%
H-Tail Area (m^2)	12.75	13.24	+3.84%
V-Tail Area (m^2)	12.49	11.85	-5.12%
OEW (kg)	12850	12795	-0.43%
MTOW (kg)	22500	22455	-0.20%
Block Fuel (kg)	2102	2108	+0.29%
Total GWP (kg)	5458	5466	+0.15%
Total DOC (M$/year)	12.90	12.85	-0.39%
S.S.M. (%)	5.12	4.55	-11.13%
Take-Off Field Length (m)	1311	1312	+0.076%
Landing Field Length (m)	1168	1163	-0.43%

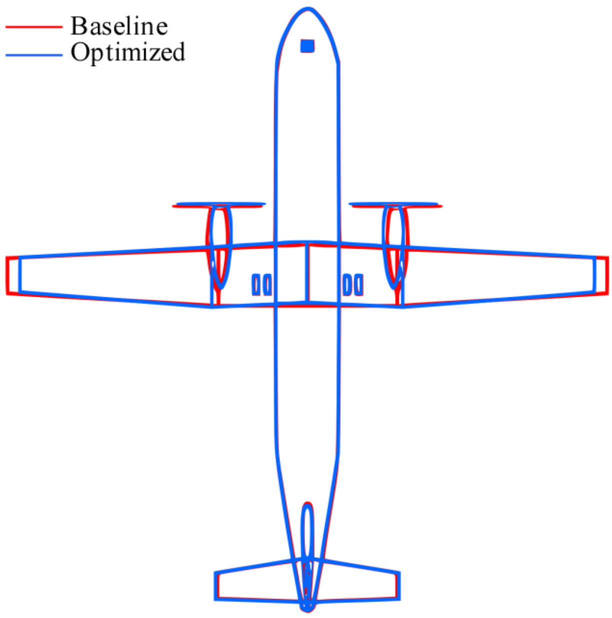

Fig. 9. Baseline and Optimized aircraft models

References

[1] Stanford Aerospace Design Lab. *SUAVE*. Stanford Aerospace Design Lab. Available at: http://suave.stanford.edu/. Access date: June 2018.

[2] CFS Engineering. *CEASIOM*. Available at: https://www.ceasiom.com/wp/. Access date: June 14th 2018.

[3] DAR Corporation. *AAA - Advanced Aircraft Analysis*. Available at: www.darcorp.com/Software/AAA/. Access date: June 14th 2018.

[4] Raymer, D. *RDS-wing aircraft design software*. Available at: www.aircraftdesign.com/rds.shtml Access date: June 14th 2018.

[5] Lissys Ltd. *Piano*. Available at: www.piano.aero/. Access date: June 14th 2018.

[6] TXT Group. *Pacelab APD*. Available at: https://www.txtgroup.com/markets/solutions/pacelab-apd/. Access date: June 14th 2018.

[7] University of Manchester. *FLIGHT*. Available at: http://www.flight.mace.manchester.ac.uk/index.html Access date: June 14th 2018.

[8] Nicolosi F and Paduano G. Development of a Software for Aircraft Preliminary Design and Analysis. *Proceedings of the 3rd CEAS Air and Space Conference*, Venice, Italy, 2011.

[9] Zhang M, Rizzi A W, Nicolosi F and De Marco A. Collaborative Aircraft Design Methodology Using ADAS Linked to CEASIOM. *Proceedings of the 32nd AIAA Applied Aerodynamic Conference*, Washington DC, 2014.

[10] Nicolosi F, De Marco A, Attanasio L and Della Vecchia P. Development of a Java-based framework for aircraft preliminary design and optimization. *AIAA Journal of Aerospace Information Systems*, Vol. 13, 2016. https://doi.org/10.2514/1.I010404.

[11] De Marco A, Cusati V, Trifari V, Ruocco M, Nicolosi F and Della Vecchia P. A Java Toolchain of Programs for Aircraft Design. *Proceedings of the 6th CEAS Air and Space Conference*, Bucharest, Romania, 2017.

[12] Ciliberti D, Della Vecchia P, Nicolosi F and De Marco A. Aircraft directional stability and vertical tail design: A review of semi-empirical methods. *Progress in Aerospace Sciences*, Vol.95, pp.140-172, 2017. http://dx.doi.org/10.1016/j.paerosci.2017.11.001

[13] Nicolosi F, Della Vecchia P and Ciliberti D. Aerodynamic interference issues in aircraft directional control. *ASCE's Journal of Aerospace Engineering*, Vol.28, N.1, 2015 https://doi.org/10.1061/(ASCE)AS.1943-5525.0000379

[14] Nicolosi F, Vecchia P D, Ciliberti D and Cusati V. Development of new preliminary design methodologies for regional turboprop aircraft by means of CFD analyses. *29th Congress of the International Council of the Aeronautical Sciences (ICAS)*, St. Petersburg, Russia, 2014.

[15] Nicolosi F, Della Vecchia P and Corcione S. Aerodynamic analysis and design of a new twin-engine commuter aircraft. *28th Congress of the International Council of the Aeronautical Sciences (ICAS)*, Brisbane, Australia, pp. 321-332, 2012.

[16] Nicolosi F, Della Vecchia P and Corcione S. Design and Aerodynamic Analysis of a Twin-engine Commuter Aircraft. *Aerospace Science and Technology (Elsevier) AESCTE*, Vol. 40, pp.1-16, 2015. http://dx.doi.org/10.1016/j.ast.2014.10.008.

[17] Nicolosi F, Corcione S, Della Vecchia P. Commuter Aircraft Aerodynamic Characteristics through Wind Tunnel Tests. *Aircraft Engineering and Aerospace Technology (EMERALD)*, Vol. 88, N. 4, pp. 523-534, 2016. https://doi.org/10.1108/AEAT-01-2015-0008.

[18] Nicolosi F, De Marco A, Della Vecchia P and Sabetta V. Roll performance assessment of a light aircraft: flight simulations and flight tests. *Aerospace Science and Technology (Elsevier)*, Vol. 76, pp. 471-483, 2018. ISSN 1270-9638 https://doi.org/10.1016/j.ast.2018.01.041.

[19] Nicolosi F, Della Vecchia P, Ciliberti D and Cusati V. Fuselage Aerodynamic Prediction Methods. *Aerospace Science and Technology (Elsevier) AESCTE*, Vol. 55, pp. 332-343, 2016. ISSN 1270-9638. http://dx.doi.org/10.1016/j.ast.2016.06.012

[20] Coiro D and Nicolosi F. Design of Low-Speed Aircraft by Numerical and Experimental Techniques Developed at DPA, *Aircraft Design Journal*, Vol. 4, N. 1, pp. 1-18, 2001. ISSN 1369-8869

[21] Pascale L and Nicolosi F. Design and aerodynamic analysis of a light twin-engine propeller aircraft, *26th Congress of International Council of the Aeronautical Sciences (ICAS)*, Anchorage, Alaska, USA, Vol. 1, pp. 3890-3904, 2008.

[22] Nicolosi F, Ciliberti D, Della Vecchia P and Corcione S. Wind tunnel testing of a generic regional turboprop aircraft modular model and development of improved design guidelines. *AIAA Aviation and Aeronautics Forum and Exposition (AIAA Aviation)*, Atlanta, Georgia, 2018.

[23] Della Vecchia P and Nicolosi F. Aerodynamic guidelines in the design and optimization of new regional turboprop aircraft. *Aerospace Science and Technology (Elsevier) AESCTE*, Vol. 38, pp. 88-104, 2014. ISSN 1270-9638. https://doi.org/10.1016/j.ast.2014.07.018

[24] Coiro D, Nicolosi F and Grasso F. Design and Testing of Multi-Element Airfoil for Short-Takeoff-and-Landing Ultralight Aircraft. *Journal of Aircraft*, Vol. 46, N. 5, pp. 1795-1807, 2009. ISSN 0021-8669. https://doi.org/10.2514/1.43429

[25] Della Vecchia P, Corcione S, Pecora R, Nicolosi F, Dimino I and Concilio A. Design and integration sensitivity of a morphing trailing edge on a reference airfoil: The effect on high-altitude long-endurance aircraft performance. *Journal of Intelligent Material Systems and Structures*, Vol. 28, N. 20, pp. 2933-2946, 2017. https://doi.org/10.1177/1045389X17704521

[26] OPEN CASCADE SAS. *OPEN CASCADE*. Available at: https://www.opencascade.com/. Access date: June 14th 2018.
[27] Sforza P. *Commercial airplane design principles*. Oxford, Butterworth-Heinemann, 2014.
[28] Carichner G and Nicolai L. *Fundamentals of aircraft and aircraft design*. AIAA, Reston, 2013.
[29] Raymer D. *Aircraft Design: A Conceptual Approach*. AIAA Education Series, Washington DC, 1992.
[30] Sadraey M. *Aircraft design*. Chichester, Wiley, 2013.
[31] Howe D. *Aircraft conceptual design synthesis*. Norwich, NY, Knovel, 2006.
[32] Kundu AK. *Aircraft design*. Cambridge, Cambridge University Press, 2014.
[33] Torenbeek E. *Advanced Aircraft Design*. Wiley, Delft, The Netherlands, 2013.
[34] Torenbeek E. *Synthesis of subsonic airplane design*. Delft University Press, 1976.
[35] Trifari V, Ruocco M, Cusati V, Nicolosi F, De Marco A. Java framework for parametric aircraft design – ground performance. *Aircraft Engineering and Aerospace Technology*, Vol. 89 Issue: 4, pp.599-608, 2017. https://doi.org/10.1108/AEAT-11-2016-0209
[36] Oracle. *JavaFX: Getting Started with JavaFX*. Available at: http://docs.oracle.com/javase/8/javafx/get-started-tutorial/jfx-overview.htm#JFXST784. Access date: June 14th 2018.
[37] Gluon. *Scene Builder*. Available at: http://gluonhq.com/products/scene-builder/. Access date: June 14th 2018.
[38] Drela M. *AVL (Athena Vortex Lattice)*. MIT Aero & Astro Harold Youngren, Aerocraft, Inc. MIT. Available at http://web.mit.edu/drela/Public/web/avl/ Access date: June 14th 2018.
[39] Carmichael R. *USAF Digital Datcom*. 2013. Available at: www.pdas.com/datcom.html. Access date: June 14th 2018.
[40] Siemens AG. *STAR-CCM+*. Available at: https://mdx.plm.automation.siemens.com/star-ccm-plus. Access date: June 14th 2018.
[41] Berndt J and De Marco A. Progress on and Usage of the Open Source Flight Dynamics Model Software Library, JSBSim. *Proceedings of the AIAA Modeling and Simulation Technologies Conference*, Chicago, Illinois, USA, 2009.
[42] Ciampa PD and Nagel B. The AGILE Paradigm: the next generation of collaborative MDO. *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Denver, Colorado, 2017. https://doi.org/10.2514/6.2017-4137
[43] Prakasha1 PS, Ciampa PD, Della Vecchia P, Ciliberti D, Voskuij M, Charbonnier D, Jungo A, Zhang M, Fioriti M, Anisimov K and Mirzoyan A. Model Based Collaborative Design & Optimization of Blended Wing Body Aircraft Configuration: AGILE EU Project. *AIAA Aviation and Aeronautics Forum and Exposition (AIAA Aviation)*, Atlanta, Georgia, 2018.
[44] Nicolosi F, Corcione S, Trifari V, Cusati C, Ruocco M and Della Vecchia P. Performance evaluation and DOC Estimation of an Innovative Turboprop Configuration. *AIAA Aviation and Aeronautics Forum and Exposition (AIAA Aviation)*, Atlanta, Georgia, 2018.
[45] Della Vecchia P, Stingo L, Nicolosi F, De Marco A, Cerino G, Ciampa PD, Fioriti M, Zhang M, Mirzoyan A, Aigner B and Charbonnier D. Advanced turboprop multidisciplinary design and optimization within AGILE project. *AIAA Aviation and Aeronautics Forum and Exposition (AIAA Aviation)*, Atlanta, Georgia, 2018.
[46] Yang X-S. *Nature-inspired metaheuristic algorithms*. Luniver Press, Frome, U.K., 2010.
[47] *MOEA Framework, A Free and Open Source Java Framework for Multiobjective Optimization - User Manual Version 2.1*. Hadka D. 2011.
[48] Coello Coello CA, Lamont G and Van Veldhuisen D. *Evolutionary algorithms for solving multi-objective problems*. 2nd edition. Springer, New York, 2007, pp.17-18.
[49] Kollat JB and Reed PM. Comparison of Multi-Objective Evolutionary Algorithms for Long-Term Monitoring Design. *Advances in Water Resources*, Vol. 29, N.6, pp.792-807, 2006.
[50] Sierra MR and Coello Coello CA. Improving PSO-based multi-objective optimization using crowding, mutation and ε-dominance. *Evolutionary Multi-Criterion Optimization*, Berlin,Germany, pp.505-519, 2005.
[51] Ruijgrok GJJ and Van Paassen DM. *Elements of Aircraft Pollution*. Delft University Press, Delft, Netherlands, 2005.

Copyright Statement