Journal of Energy and Power Engineering 8 (2014) 1590-1598

Single Platform Integration Environment for Turbine Rotor Design and Analysis

Francois Lagloire[1], Yannick Ouellet[1], Benoit Blondin[2], Francois Garnier[1] and Hany Moustapha[1]

1. École de technologie supérieure, Montréal H3C 3R9, Canada

2. Pratt and Whitney Canada, Montréal J4G 1A1, Canada

Received: January 14, 2014 / Accepted: March13, 2014 / Published: September 30, 2014.

Abstract: The present article covers briefly state of the art software interoperability technical solutions and the development of the first module of a new single platform D & A (design & analysis) tool for simulation and prediction of stress and burst behavior of turbine rotating disc a preliminary design stage. This platform singularity requires integration of multiple CAD (computer assisted design) & FEA (finite element analysis) tools processing in batch mode and driven from a SPIE (single platform integration environment). This first module is also to demonstrate, for an axial turbine disc hub axi-symmetric component, feasibility and usefulness of such a platform at preliminary design stage. Expected benefits of the D & A single platform are to improve output accuracy, reduce cycle time, improve process quality and improve resource productivity.

Key words: Aero-engines, gas turbines, preliminary design, turbines, optimisation.

Nomenclature

API	Application program interface
CAD	Computer aided design
CAE	Computer aided engineering
CF	Centrifugal
D & A	Design & analysis
DOS	Disc operating system
FEA	Finite element analysis
GUI	Graphical user interface
IE	Integration environment
QA	Quality assurance
SPIE	Single platform integration environment
STEP	Standard exchange of product model data
UI	User input
2D	Two dimensional

1. Introduction

The preliminary design phase of a turbine rotor has an important impact on the architecture of a new engine, as it sets the technical orientation right from start and provides an estimate of future product performance, weight and cost. In addition, preliminary design cycle time has become critical into capturing business opportunities [1-3]. Improving upfront accuracy and quality also alleviates downstream detailed design work and therefore reduces overall product development cycle time.

Preliminary design of a turbine rotor is also a complex task which implies D & A simulation work of multi-component assemblies (disc, blade, fixing, etc.) and multi-physics modeling (aerodynamic, structural, etc.). Being highly iterative, the rotor D & A process usually leads to sub-optimal technical solutions when iterations are limited by time constraints.

The present methodology and design process can be improved as it requires the use of multiple unlinked CAE, CAD and FEA software tools since there is no apparent commercial software product performing all the required tasks at desired utmost level.

Commercial software tools are also usually closed black box type systems and offer little or no visibility of underlying FEA calculations or solving algorithms.

Corresponding author: Hany Moustapha, professor and director of aerospace programs, ETS, research fields: aero-engines, propulsion design, and turbines. E-mail: Hany.Moustapha@etsmtl.ca.

Additionally, aerospace enterprises handle specific and complex design problems that require and rely on in-house specific extensive knowledge. No commercial CAE, CAD and FEA tool fully addresses all of these design problems [4-6].

Furthermore, total ownership over product engineering analysis in the aerospace industry calls for total proprietary engineering analysis capabilities and controls. Hence, actual tools used to design and analyse rotors at preliminary design stage are usually in-house coded programs that translate proprietary knowledge into specific in house analysis codes.

Even though these programs have significant D & A capabilities, they are bounded to operate within a single environment, i.e., they have no links to external CAD or FEA systems and thus do not offer all of which CAD or FEA systems allow.

Improving efficiency to accelerate the D & A process or to reduce the gap with an optimal technical solution is thus actually relevant. Since the mid 90's, projects pertaining to integration of multiple software tools to improve the D & A process within the aerospace industry have abounded [1, 4, 5]. Hence developing a SPIE that integrates multiple CAE, CAD and FEA capabilities executed by different commercial software products and/or proprietary program codes is hereby experimented.

One challenge of software integration (interoperability) is to use data models that exist in different topology environment in a way that neutral format application protocols such as STEP do not allow. Exchanging data form one software product to another must ensure native models can be accessed and model syntactic definitions and significations are formally maintained throughout exchanges.

An additional challenge of software integration relates to the external tools execution control. The SPIE must have direct authority over these external tools for execution command and coordination.

2. SPIE Capabilities

For a new disc D & A platform to be useful, it must provide at least equivalent or improved capabilities compared to existing tools such as asdisc component configuration, design, analysis and auxiliary functionalities.

Such items for component configuration include, amongst others, material selection, temperature setting, symmetric/asymmetric web profiles and shaft to web appending.

Design capabilities include editing of the complete set of geometric parameter axisymmetric model; view of axisymmetric model equally scaled and explicitly represented in relation to the engine axis to properly gauge the disc aspect ratio and representation of axial and radial center of gravity position.

Analysis capabilities include one dimensional stress and calibrated burst calculations and 2D stress analysis with:

- FE mesh density UI control;
- 2D non-linear geometry FE calculations for elastic and plastic material behavior;
- Maximum stress results display.

2D analysis cover burst speed analysis with:

- FE mesh density UI control;
- 2D non-linear geometry finite element calculations for plastic material behavior;
- Burst speed results display.

Auxiliary features are also important and include mainly data management capabilities such as legacy data file loading and current data file loading and storage. All current output files generated during a D & A run are accessible to the user from a dedicated output directory on an active session temporary base. All required inputs and selected outputs are permanently saved within an activity log that the user can search through and extract data he wishes to manipulate further. A summary report generation also reduces typical data and file manipulation burden and inefficiencies.

Since the 2D FEA execution take up some time, activity reporting is communicated to the user to inform of activity progress. These monitoring/reporting codes are also handy for programmers while working or upgrading the application.

3. SPIE Concepts

A SPIE concept does not simply imply having a single GUI, but rather an application running within a single software environment with automated and secured links to external design, analysis or solving tools running in batch mode. This is the key aspect of the platform singularity.

Multiple solutions can enable interoperability and single platform concepts and range from:

- A completely commercial-free in house coded IE, CAD and FEA system;
- A completely commercial base system comprising of SPIE linking to other commercial base CAD and FEA tools.

A commercial base SPIE offers, amongst others, the following benefits:

- complex geometry model editing,
- powerful geometry model interrogation,
- programming effort free implementation.

But these benefits come at a cost, as this concept requires a more elaborated set-up and complex infrastructure such as servers or dynamic link libraries.

They are also implicitly license dependent. Therefore, users are limited by license availability and cost of ownership and maintenance is usually significant.

Version control with CAD and FEA tools can be an issue as these evolve in unsynchronized fashion.

Interactions between SPIE and given CAD or FEA are limited by the provided interoperability capabilities. These can be different from given specific needs and thus limiting desired functionalities.

PHX Model Center supplied by Phoenix Integration Inc is one of the latter type solutions. Wrappers are used to encapsulate a wide choice of CAD and FEA software products or legacy codes [7] and analysis process can be designed from the SPIE GUI with great flexibility for specific needs. But such commercial SPIE products are fairly new and survivability and supportability is still uncertain.

On the other end of the solution scale, commercial free concepts eliminate almost all license dependencies and capability limitations but require total ownership and can be very labor intensive.

Complex geometric models can be very hard to handle and require extensive programming and proofing efforts.

Hybrid or intermediate solutions located between these two boundaries are also valid and can alleviate some of the limitations of the commercial base solution and uncertainties of commercial free solutions.

Specific needs evaluations are paramount to properly select the best architecture solution to fit specific requirements.

In the present study case, the SPIE reusability and ease of implementation are of paramount importance since the SPIE will be reused for other single component modules, scaled up to multi-components assemblies and extended to thermal and aero disciplines.

Other important platform selection criteria include speed of execution and complex geometry handling capabilities.

The present SPIE experiment and study work is hence carriedout to link with limited programming efforts: a proven commercial CAD tool; a commercial FEA tool that allows in-house specific analysis design; and an opened commercial SPIE that can interface with these selected CAD and FEA through commercial APIs.

Such SPIE concept architecture is illustrated in Fig. 1. The main elements defining this architecture are: the SPIEGUI through which the user interacts (user inputs-controls-outputs); the CAD tool operating in batch session; and the FEA tool also operating in batch session.

The APIs, operate as the bridge interfacing the IE and the above mentioned design and analysis external tools.

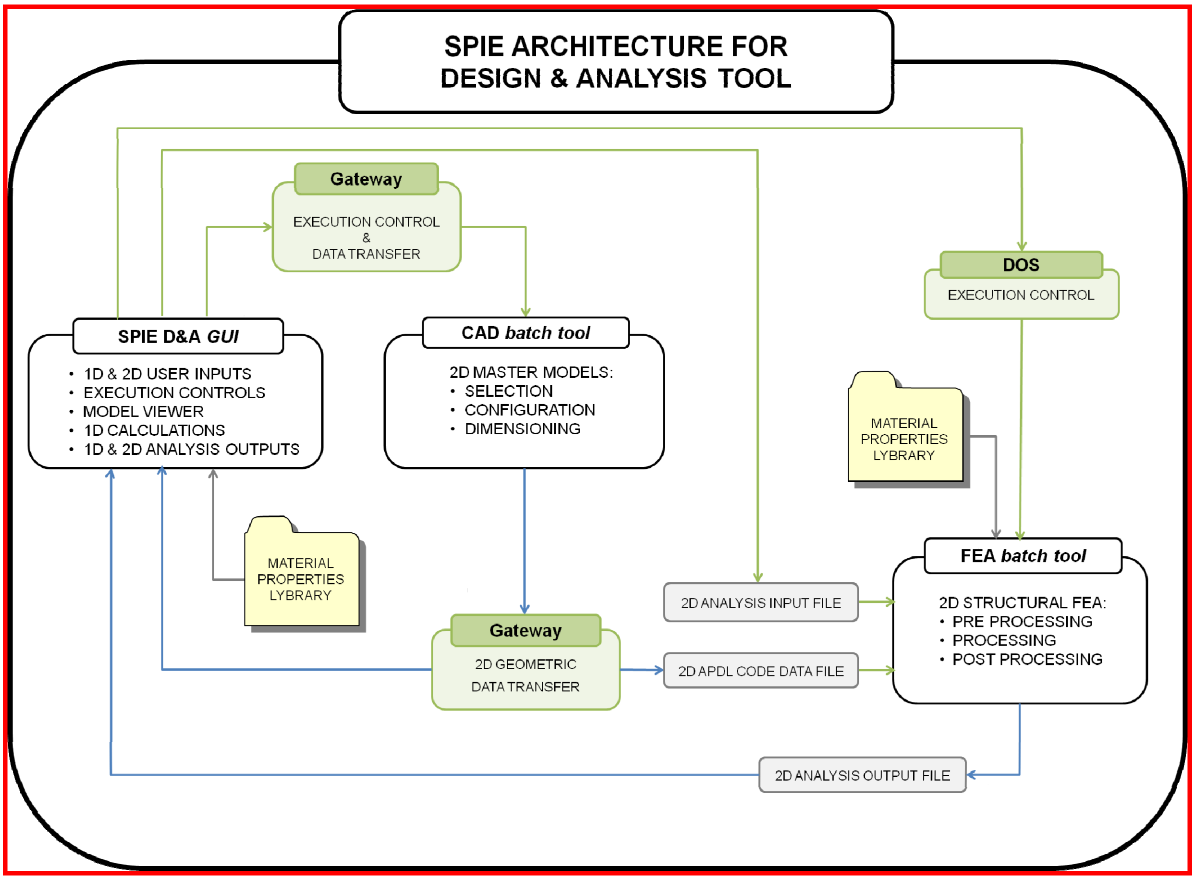

Fig. 1 Single platform integration environment "SPIE" architecture.

This generic architecture is expected to be reused and adapted with limited efforts for:

- other parametric models (multi models),
- other discipline analysis module.

The commercial IE has been selected based on numerous key capabilities such as :

- customized GUI creation,
- internal execution controls and coordination (internal scripts),
- external tool execution control and coordination,
- model geometric representation,
- data structure and management capability.

Established CAD products used throughout the aerospace industry, amongts others, include ProEngineer, Catia or NX. These CAD products offer similar model parameterization accuracy and flexibility and they can also be driven by external commands and operate in batch mode execution without any active GUI. Any one of these can now be linked through availablecommercial gateways or specific APIs.

The commercial FEA tools used to execute FEA solving can be operated in batch mode through DOS commands.

3.1 SPIE D & A GUI

The single platform GUI has been created for user to system bi-directional interaction and allows data input editing, execution control and data output display.

The layout has been developed to provide intuitive learning and use for both design and analysis functionalities. Data input and output clustering and coloring is used to acknowledge easily and rapidly data meaning, value and status. Controls are located to ensure meaning and limit mouse travel.

Upon any user input modification, every user input dependent analysis control object is set to yellow color to highlight the unsynchronized state between the user input and the analysis output. The control object color is reset to its original green color upon successful execution completion. This logic is applied throughout all of the disc module functionalities and across different GUI windows, and provides the user with a rapid sense of input to output relation status.

3.2 SPIE-CAD Bi-directional Link

The interface between SPIE and the CAD software is a translation-free connectivity and is a key feature of the single platform architecture. Scripts invoke APIs that allow bi-directional data exchange and execution control. These APIs, considerer as "black boxes", are C++ scripts that the SPIE is able to parse and compile from their native language format and use for a given execution.

Three code scripts stand out as key elements into achieving this SPIE to CAD bi-directional connectivity. The first key script allows loading and managing the CAD master models from a CAD batch session. The second key script allows the master model to be synchronized with the user design intent (GUI user input) through a model indexing synchronization logic. The third key script captures the native CAD model representation and reproduces this representation through the SPIE GUI whenever there is model regeneration in the CAD system. This step is bypassed if there is no regeneration previously executed. Synchronization of model representation within the GUI viewer is controlled by a model index code that gets to be incremented as per model regeneration in the CAD environment. The regular CAD product licence is typically required on a continuous base for this batch session to operate.

3.3 CAD Master Model

The disc CAD models, such as the rotor disc illustrated in Fig. 2, are defined by an established set of parameters. The disc CAD master model parameterization used for the single platform is inspired from this existing parameter set but some modifications have been made, without altering the final part result, in order to include four appendages that can be activated/deactivated by boolean type parameters; improve model robustness (unfeasible model detection); and allow zero value input setting (with offsets).

3.4 SPIE-FEA Link

Although the FEA are batch executed sessions, the SPIE to FEA relation is different from the SPIE to CAD relation and is not managed by API calls but by simple DOS execution commands.

Complementing the DOS execution command from the SPIE to the FEA are two key input files. The first is a text file generated by the SPIE tool containing the specific information and instructions required for analysis execution (preprocess/process/post-process). The second input file contains the geometric model definition written in the specific FEA programming language format that allows reconstruction of the given CAD model within the FEA environment during analysis preprocessing (geometry components and labels). This item is novel and key into achieving the automated link from the CAD modeler to the FEA analysis tool. It can only be generated during the active CAD batch session by a specific API.

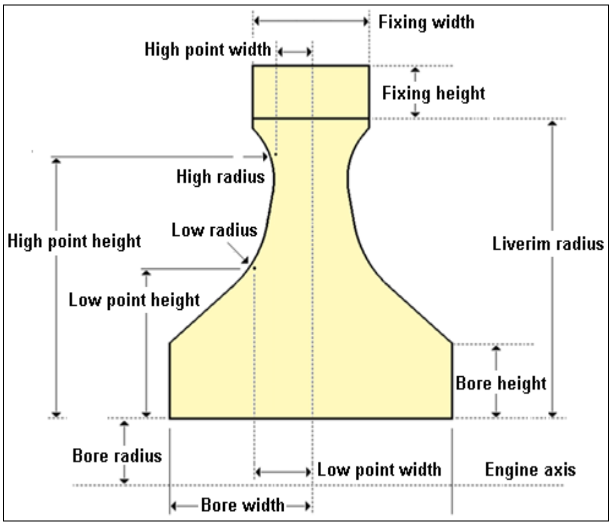

Fig. 2 CAD master model.

The SPIE tool maintains contact with the FEA as long as the execution command status is active after which the SPIE tool extracts and displays specific data outputted by the FEA. The program manages the FEA release version, its license availability and a single FEA license is required during FEA processing and is released once the analysis is completed. All these aspects related to the FEA product use are either new or an improvement compared to actual tools as they provide an integrated, automated and secured link and reduce user license retention.

3.5 FEA Analysis

All operations executed are predefined and automated and include pre-processing/processing/post-processing. The FEA process offers pre-set analysis sequences, shown in Fig. 3, which are automatically performed from the analysis input data file without any user intervention. The codes executing these steps are either based on existing proprietary programs that have been slightly modified or appended (stress or burst processing) or completely new as they introduce new capabilities.

3.5.1 FEA Pre-processing

The preprocessing prepares the CAE model for FE analysis and includes the complete disc and appendages reconstruction from the geometric data file. The combination of appendages with the hub section is completed using radii values from the analysis input data file. The final shape is meshed using axisymmetric elements for hub section and plate elements for the fixing section in regards to the size and area of the geometric model. Node count is increased near/at bore radius and disc to appendages radii and is globally increased or decreased as per UI mesh density factor. Node temperature allocation is based on linear interpolation of specified temperature map distributions as per node position, for:

- 1D uniform from analysis input file,
- 2D radial temperature distribution from selected temperature mapping file,

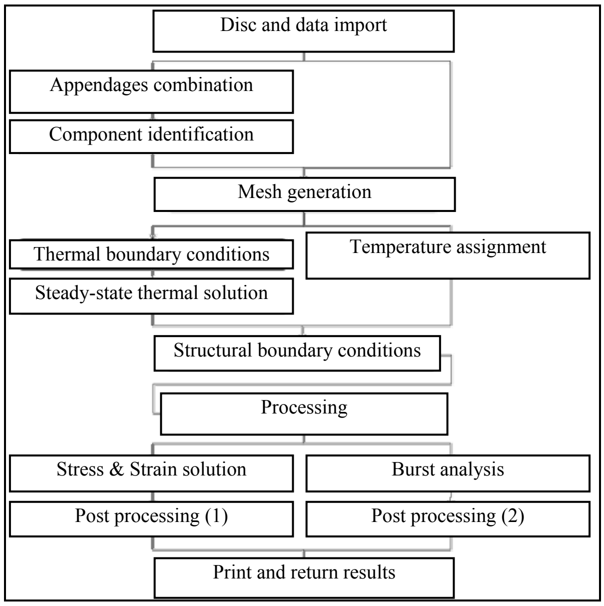

Fig. 3 Automated CAE workflow.

- 2D radial and axial temperature distribution from selected temperature mapping file.

Preprocessing steps also includes the addition of mass elements to simulate blade and fixing CF pulls, the application of structural boundary conditions (including rotational speed) and the material properties extraction from dedicated FEA material files library.

3.5.2 FEA Processing

Processing executes all the required finite element analysis using the FEA tool solving capabilities from the analysis input file data and per selected options such as:

- linear and non-linear geometric behavior,
- elastic and plastic material behavior.

3.5.3 FEA Post-Processing

The post processing section automatically extracts and generates output results and technical plots, sorting maximum stress/strain values and node location. Analysis convergence is also checked at maximum stress/stain nodes and returned to the user with the output results.

3.6 FEA Codes

The single platform application is coded using simple and common programming's best practices.

Function scripts are clustered as per program functionalities to ease testing and reuse, as per upstream and downstream data dependency to ease future module integration and as per potential changes related to future upgrades.

Application reliability and stability requires the program to be error/fault free. One way to accomplish this has been to combine logical conditional statement programming with error catching/error variables which numerical values are feed by:

- SPIE function/command built-in feedback return codes,
- API functions feedback built-in return codes,
- coded operation checks.

One such check is executed in the SPIE environment in order to ensure the geometric model reproduced in the FEA environment, from the FEA programing language code, is cohesive with the user intent by comparing the disc geometric properties extracted from the CAD master model to geometric properties calculated and outputted by the FEA.

Interference detection (user access within input or output files) has also been set to ensure no errors are generated by execution interferences.

When preceding D & A programs are not completed , the required predecessor functionalities prior to the given analysis run.

3.7 Data Management

Rotor components modeling can be described to be hierarchical with numerous parent/child relations. The rotor data structure has, therefore, been set to be cohesive with this component modeling with all of sub-components and its parameters in a tree like pattern [3].

This “rotor” data set branch can be extended later on as the rotor disc module gets to be appended with any additional module data set that share a similar parent-child relation with the rotor such as the blade or the cover plate component modules [8].

This data structure allows storing and linking almost any attributes to specific parameters or group of parameters. Parameter name, value, units, accuracy, index are example of used attributes.

UI data format displayed through the GUIs has been set to match the calculation data format to reduce as much as possible calculations errors incurred by numerical data truncation and ensure cohesive results between different analysis runs. This data attribute is set within the data structure as a data attribute.

Material properties required to run the 1D and 2D analyses are all extracted from the same material library created specifically for the calculations. These properties are stored within text files and accessed from the CAE tool for the 1D calculations and by the FEA tool for the 2D calculations.

4. Application Performance

As stated in the disc application D & A cycle duration is critical and paramount to improve resource productivity. Execution monitoring and scripts streamlining has enabled the application to perform to its utmost level, without adding unnecessary lags or delays as application features and capabilities expanded. Even if some small delays are incurred by GUI interactions, error catching and conditional flow codes execution, the performance is within expected targets. Automation has led to a 30% to 40% reduction in 2D stress and burst analysis execution cycle time.

4.1 Improved D & A Fidelity

1D stress and burst calculations are based, amongst others, on disc area. Relying on a CAD for providing a high-fidelity model data/description ensures 1D calculation utmost accuracy. This has led up to 5.0% accuracy gain over current simplified disc area calculations.

4.2 Reduced D & A Cycle Time

Managing the interfaces between the hosting environment and the external tools and therefore eliminating user manual operations ensures, all data transfer and execution commands are executed almost instantly. This automation eliminates the manual operations delays and lags and hence reduces the overall D & A cycle duration.

Having the 1D calculations and the 2D FEA linked within the same environment allows direct 1D calibration between 1D and the 2D FEA when required. This calibration process is now integrated and automated, i.e., without external tools or user action.

4.3 Improved D & A Quality

Once again, eliminating user manual operations ensures all data transfer and execution commands are secured and can only be impaired by system discrepancy and failures. Eliminating user manipulations eliminates manual data transfer errors from the D & A processes cycle.

Furthermore, having all functionalities within the same platform allows for easy input and output data synchronization and management. This can be troublesome to accomplish if input and output data are hosted in different environments. The one dimensional analysis automated calibration is one case where manual errors are eliminated.

4.4 Improved Productivity

All of above mentioned benefits are expected to have an impact on human resources productivity as rotating structure specialists spend more time on D & A added value tasks and less time on non-added value tasks (open/close software products, copy/paste data files). This enables processing more preliminary case studies or executing more iterations and refining further a given preliminary case study.

The automated on-demand use of FEA licenses, as the SPIE releases the FEA product once a FEA run is completed, makes for a better use of the limited number of licenses and eliminates unproductive license retention.

4.5 Tool Implementation

The disc module SPIE application is deployed as a simple stand-alone application and used in a compiled format, which limits any code tampering ensuring application integrity and reliability.

The SPIE D & A module tool stands as a valid proof of concept of multiple CAD and CAE interoperability. This base architecture is considered to be generic and can be reused for scaling up the disc module with additional rotor component modules (blade, cover plate) or other physical disciplines.

5. Limitations and Risks

The gateway link does not allow view of the shape whengeometric errors happen in the CAD tool. This is to be improved in a future version of the gateway.

As stated earlier, the single platform makes a better use of CAD and CAE tool licenses, but is implicitly constraint by new dependencies towards the API licenses.

Every software tool used in this single platform is provided by an external commercial source and evolves through time. Tool version fit with the API as product version increment must not be neglected through time.

6. Conclusions

The new single platform D & A tool concept has proven to be feasible and a real improvement compared to the present D & A tool, enabling a cohesive single integrated simulation environment capturing the strength of targeted commercial software for their specific capabilities.

The reduction in execution time and user manipulation, through an automated and secured system, demonstrated evident gains when turbine rotor D & A speed and quality are of paramount importance.

This first SPIE module was also put forth with acceptable in-house exploration and programming efforts. Hence, the concept architecture module sets the frame base for scaling up the simulation tool capabilities through increase of data volume (input/output), link other external tool (such as iSight for optimization capabilities), D & A of multiple components and integration of multiple disciplines.

References

[1] Panchenko, V., Patel, K., Moustapha, S. H., Dowhan, M. J., Mah, S., and Hall, D. 2003. "Preliminary Multi-Disciplinary Optimization in Turbomachinery Design." Presented at 2003 NATO-AVT Symposium, Paris, France.

[2] Atherton C. 2002. "An Approach to Multidisciplinary Design, Analysis & Optimization for Rapid Conceptual Design." Presented at the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, USA.

[3] Japikse, D., Katz, J., Wisler, D., Guinzburg, A., and Yamada. H. 2002. *Developments in Agile Engineering for Turbomachinery*. Concepts NREC, White River Junction, USA.

[4] Ouellet, Y., Roy, F., Moustapha, H., and Garnier, F. 2013. *Preliminary Design of Axial Turbine Disc for Aircraft Engines*. CASI (Canadian Aeronautics and Space Institute), Toronto,

[5] Sobieszczanski, J. S., and Haftka, R. T. 1997. "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments." *Structural and Multidisciplinary Optimization* 14 (1): 1-23.

[6] Townsend, J. C., Samareh, J. A., Weston, R. P., and Zorumski, W. E. 1998. *Integration of a CAD System into an MDO Framework*. Hampton, USA: NASA (National Aeronautics and Space Administration) Langley Research Center.

[7] Karl, A. G., Lori, P. O., James, W. F., Elwood, W. S., and Wu, L. 2011. *Integration of Multifidelity Multidisciplinary Computer Codes for Design and Analysis of Supersonic Aircraft*. Hampton, USA: NASA (National Aeronautics and Space Administration) Langley Research Center.

[8] Carsten, M. L. and Martin, H. 2011. *A Distributed Toolbox for Multidisciplinary Preliminary Aircraft Design*. Germany: German Aerospace Center.