

## AN AUTOMATED CAD/CAE INTEGRATION SYSTEM FOR THE PARAMETRIC DESIGN OF AIRCRAFT WING STRUCTURES

ABDELKADER BENAOUALI, STANISŁAW KACHEL

*Military University of Technology (MUT), Warsaw, Poland*

*e-mail: abdelkader.benaouali@wat.edu.pl*

In order to take advantage of the sophisticated features offered by CAD and CAE packages for modeling and analysis during the design process, it is essential to build a bridge assuring a coherent link between these tools. Furthermore, this integration procedure must be automated so as to get rid of the repetitive costing effort. In this paper, a new automated procedure for the CAD/CAE integration, implemented for the parametric design and structural analysis of aircraft wing structures is presented. This procedure is based on the automation capacity available in modern computer aided tools via build-in basic programming languages as well as the capacity of the model data exchange. The geometric and numerical models can be controlled to generate a large variety of possible design cases through parameters introduced beforehand.

*Keywords:* CAD/CAE integration, design automation, aircraft wing, parametric design

### 1. Introduction

Due to the existing competition, manufacturing companies strive to decrease the product life cycle in order to produce more in a relatively reduced time (Asiedu and Gu, 1998). Consequently, a need has been raised for evolution in design tools, namely, Computer Aided Design (CAD) and Computer Aided Engineering (CAE) software (Sapuan *et al.*, 2006). One of the requirements is to shorten the time spent in the design process, especially when identical tasks exist while modeling different products or when the design process consists of repeated activities that can be programmed in a simple loop.

The progress in CAD and CAE technologies served as a major part in the improvement of the design process quality. The product development cycle changed from *design*  $\Rightarrow$  *build*  $\Rightarrow$  *test*  $\Rightarrow$  *fix* to *design*  $\Rightarrow$  *analyze*  $\Rightarrow$  *test*  $\Rightarrow$  *build* (Rowe, 2006). In the traditional approach, physical prototypes had to be built in order to be evaluated, and the design was based on the designers' experience and judgment. Nowadays, numerical analysis is used in the upstream phase simultaneously with the geometric design. The validation of a new design is performed using virtual prototyping, which involves modeling, engineering analysis and multidisciplinary optimization. CAD and CAE tools have become a quintessential part of the design process and have been widely applied in structure design (Dietrich *et al.*, 1999; Edke and Chang, 2006).

Generally, the limited capabilities of modeling features in CAE tools and those of structural analysis, if they are available, in CAD tools lead designers to use two or more independent software packages for modeling and analysis (Gujarathi and Ma, 2011). Moreover, looking for more efficient and lighter structures by a structural optimization process creates a need for sharing the geometric model and analysis information in an iterative environment.

Furthermore, to solve optimization problems, the model has to be updated after each iteration by changing the design parameters. If this task is to be performed manually, the designer has to perform a complicated and time consuming job, particularly in the case where the model is

governed by a large number of parameters (Roy *et al.*, 2008). Therefore, it is requisite not only to integrate CAD and CAE tools but also to automate the integration procedure in order to avoid designer's intervention at each iteration.

The integration between CAD and CAE tools can be easily performed due to the data exchange translators available in most popular CAD and CAE tools. In order to make the integration procedure automatic, the software used must offer the ability to automate tasks and access to features via a built-in programming language as well as the possibility to run in the batch mode.

In the recent years, a number of works has used the integration between CAD and CAE tools as a design solution. Su and Qin (2003) used IGES data files to integrate Pro/ENGINEER and ANSYS for a worm gearing design. Edke and Chang (2006) performed shape optimization of heavy load carrying components by integrating Solidworks, Pro/ENGINEER and ANSYS via IGES. Other works (Park and Dang, 2010; Wang *et al.*, 2014; Peng *et al.*, 2014) focused on developing an automated framework for CAD/CAE integration.

In this paper, a new procedure for the design process of aircraft wing structures is presented. The originality of the presented work lies in the full automation of the design process, the complexity in the case considered of the wing structure as well as the applicability to different types of wings via parametric modeling. The design process involves both geometric modeling and structural analysis by integrating two commercial CAD and CAE tools. The control of the geometric and numerical models can be done using design parameters, introduced manually in order to explore and compare different design cases. The approach is supported by a graphical user interface GUI in order to ease parameter data input and result access.

## 2. Need for CAD/CAE integration

As mentioned before, the design is no longer a pure geometric modeling task due to the considerable progression of the numerical simulation becoming the key area in product design. Nowadays, the design process involves both Computer Aided Design (CAD) and Computer Aided engineering (CAE). In addition, the evolution in the design process requires geometric modeling and mechanical analysis to be driven in parallel. The model has to be submitted into analysis in order to be verified and, therefore, modified on the basis of the results obtained. Although most of the commercial CAE tools support modeling, available CAD functions do not fulfill all desired requirements. The aim to reach a better design in an iterative environment calls for a strong connection between CAD and CAE tools, which bring us to what is called "simulation driven design" (Fig. 1). This interconnection accelerates the loop design-analysis-redesign and thus reduces costs and improves product quality.

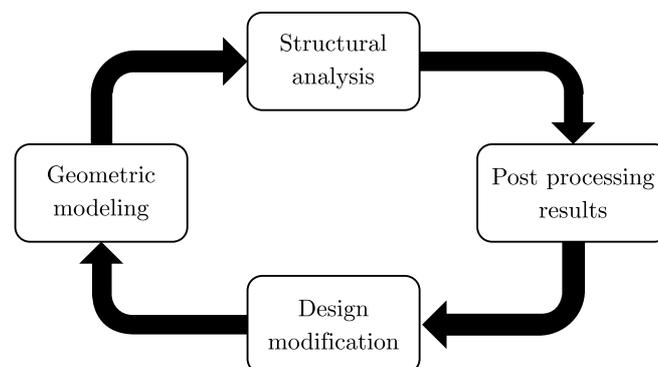


Fig. 1. Simulation driven design (CAD/CAE integration process)

The “Post processing results” step is an evaluation of the model based on the design objective taking into account the analysis results. Consequently, the modifications to be done are spotted in the “design modification” step then submitted to action when the next loop starts.

In order to ease the design modification, engineers tend to use the parametric modeling, that is to include parameters to drive the geometry in CAD models through parametric and variational approaches (Fig. 2). Thus, new designs can be generated simply by changing parameter values, which facilitates the definition of the design space.

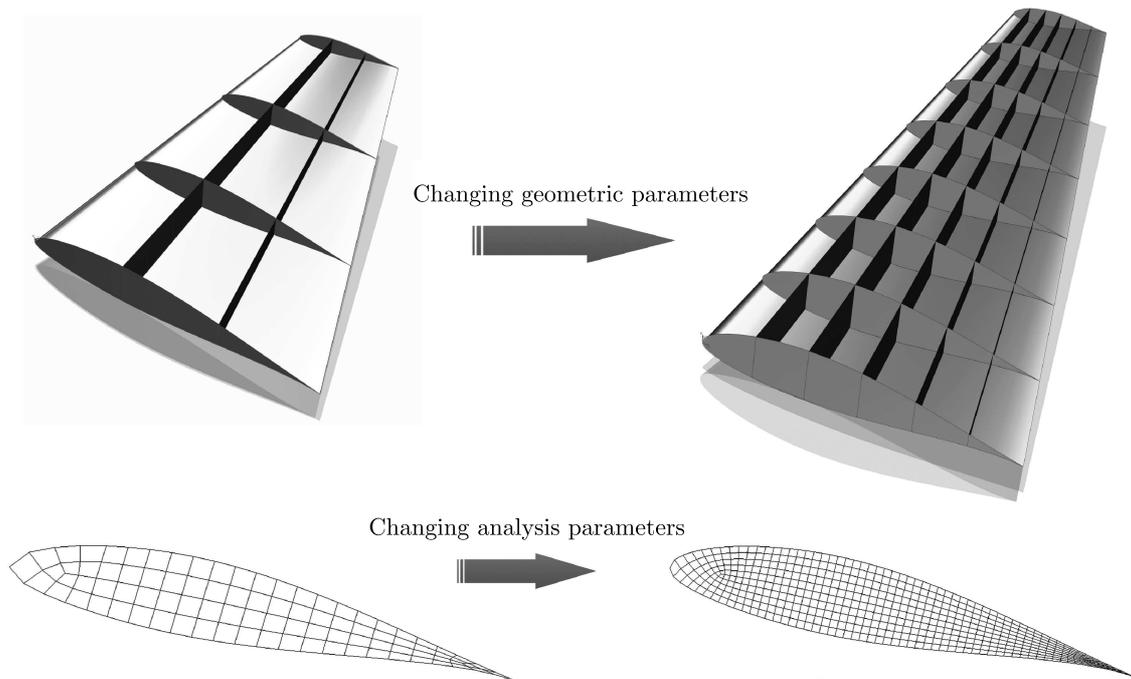


Fig. 2. Parametric modeling methodology on an aircraft wing

For the sake of ensuring the interconnection between CAD and CAE tools, data exchange translators have been developed and improved to transform data from one CAD system to another CAD file format. The CAD data translation can be done by the built-in export function supported in most of the commercial CAD. The common Data translation formats are: STEP, IGES, STL, parasolid. The geometric data can be easily provided to the CAE software via the built-in import function. Thus, the integration between CAD/CAE systems has been guaranteed. It remains to automate this export/import task so that the designer evades monotonous effort.

### 3. Design process automation

In order to modify the model, the designer must redo all tasks, namely, model the new geometry based on the new parameter values, export the model from CAD software and import it into CAE software, prepare the FEM model, submit into analysis and finally explore results. Therefore, the best way to reduce human unnecessary effort and time consumption is to precede by automation.

In other words, automation is the ability provided by the software to access modeling or analysis functionalities via a built-in programming language. The model in request can be generated after executing the program in-which several functions are called sequentially to exert different tasks. Accordingly, repeated tasks can be simply programmed in a loop and the model can be simply updated by modifying the variable values in the program. By adopting this methodology, companies can significantly reduce design time, perform more design iterations in

a relatively reduced time with a lower cost and, therefore, increase the overall quality of their design processes.

This automatic mechanism can be performed by CAD software based on the Application Programming Interfaces (APIs) and programming modules, such as the Pro/Toolkit of Pro/Engineer, CAA of CATIA, OpenAPI and OPEN GRIP of Unigraphics. In addition, most of the CAE packages are equipped with a parametric design language (PDL), e.g. Adaptive Programming Design Language (APDL) of Ansys, Patran Command Language (PCL) of Patran *et al.*. These PDLs are usually structured scripting languages by which the designers can import the geometric model, assign materials, prepare the FE mesh, apply loads and boundary conditions and submit the model to analysis. Moreover, the required analysis results can be extracted from the output file which can be post processed by programming as well. This is known as the automated finite element modeling technique AFEM (Nawijn *et al.*, 2006) and was used in many works to generate reliable FE models of complicated structures (Rocca and van Tooren, 2009; Sohaib, 2011).

The CAD and CAE software used in the integration system must allow the execution in the batch mode, that is without a graphic user interface (GUI). Consequently, it is possible to execute the CAD or CAE program from a windows command prompt which runs the corresponding software in an invisible mode. This will significantly decrease the required time for modeling and analysis compared to the graphic mode.

In addition to the two CAD and CAE tools, there is a need for an implementing software to complete the integration system. This software is responsible for controlling the integration process, it ensures the parameter value automatic modification in the programs, the execution of the programs in the software batch mode to invoke the modeling/analysis processes and the extraction of analysis results from the output of the CAE component.

The implementing software writes parameter values introduced by the designer to the modeling program. The CAD component, run in the batch mode, executes the program to generate the geometric model automatically and exports it to a common data format file. Again, the implementing software writes parameter values to the analysis program and runs the CAE component which imports the CAD model, identifies the analysis tasks, carries out the simulation process and provides the analysis results. Finally, the implementing software accesses result files and extracts the values required.

#### 4. Implementation of the integration system

The selection of the software forming integration system depends on the designer choice and the available tools. In the present work, SIEMENS NX, MSC.PATRAN/NASTRAN and MATLAB have been chosen as CAD, CAE and implementing software, respectively. The flowchart of the integration system is depicted in Fig. 3.

##### 4.1. CAD component

SIEMENS NX (formerly Unigraphics NX) is a leading parametric modeling CAD software. It offers various tools of automation: Knowledge Fusion, GRIP programming, Macros, NX Open, Journals, User Defined Objects etc.

Over the available automation tools, GRIP is used in our integration procedure. GRIP (Graphics Interactive Programming) is a convenient, flexible and powerful interactive graphic programming language. GRIP is used to create FORTRAN-like programs to operate the Unigraphics system. GRIP programming method can be used to automate most of the operations under the UG for developers. Commands are available to create geometric functions and modify existing geometry.

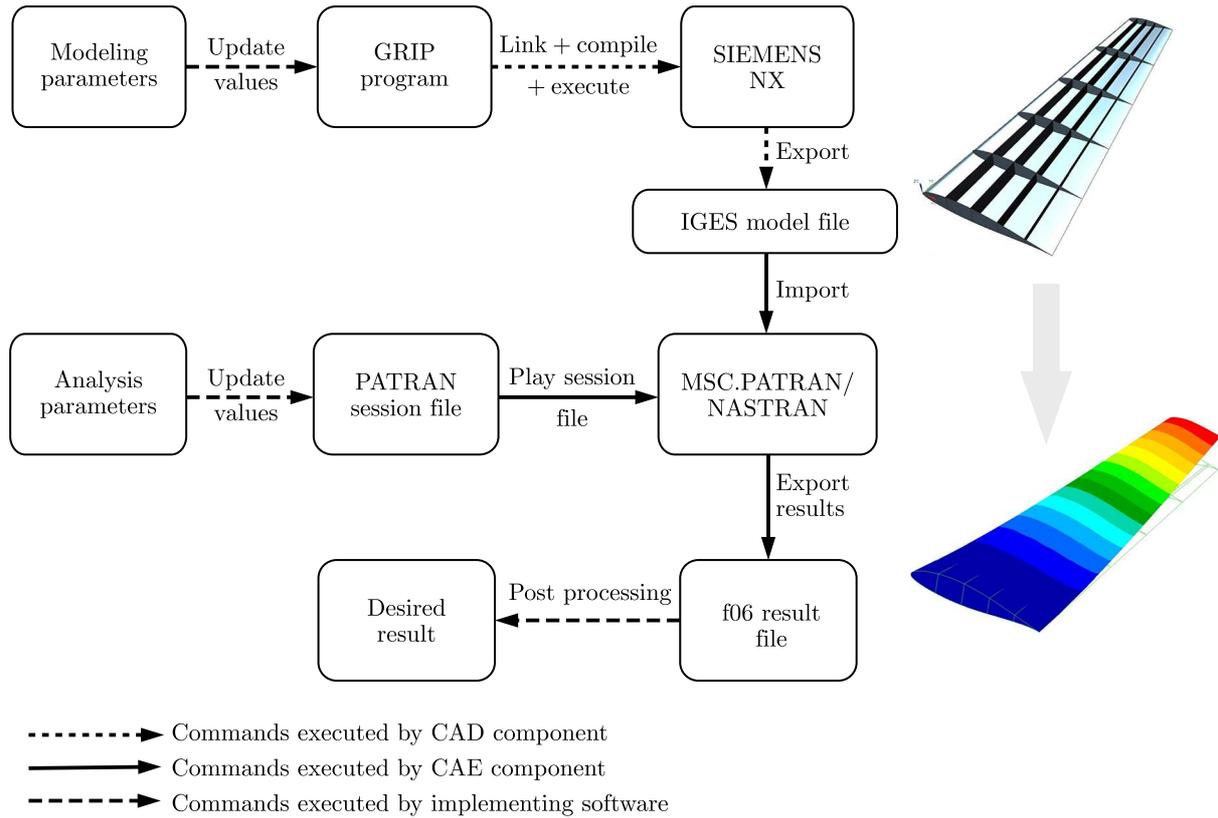


Fig. 3. Flowchart of the CAD/CAE integration system

The GRIP program is written to a text file with a (.grs) format. It has to be compiled then linked before it can be executed by the SIEMENS NX software. Thanks to the batch mode running ability, compilation, linkage and execution can be done by a WINDOWS command that operates on a batch file. This command can be executed from the implementing software MATLAB. Being the case, the designer avoid the action of opening GRADE (GRIP Advanced Development Environment) to compile and link the .grs file then SIEMENS NX software to execute the .grx file.

The GRIP program starts by creating a new active part which is necessary to execute the modeling process and ends by exporting the model to a geometry exchange format file, in our case IGES format. Unfortunately, there is no GRIP function to export the model in a format other than .prt. The solution is to use another tool for automation which is journaling, by writing a VB journal to export the model in IGES format and call it from the GRIP program.

#### 4.2. CAE component

MSC/PATRAN is a pre/post-processing software for Finite Element Analysis (FEA), providing solid modeling, meshing, analysis setup and post-processing for multiple solvers including MSC NASTRAN, ABAQUS, LS-DYNA, ANSYS, etc. It uses a simple step-by-step approach that helps to create, analyze and interpret a mathematically realistic model of the structure.

The programming language which can be used in conjunction with MSC/PATRAN is called Patran command language (PCL). PCL is a high level block structured language. It has many features similar to traditional programming languages. The PCL program is in the form of a text file called the session file (.ses) formed by a succession of PCL functions.

The session file starts with importing the geometric model, then creating materials and properties, then it prepares the FE mesh, applies loads and boundary conditions and finally

submits the analysis job to MSC.NASTRAN (The FEA solver chosen). When the job is done, the result file (.f06 Format) is generated.

## 5. Case study of an aircraft wing structure

One of the most critical components and complicated structures of an aircraft is the wing. It is a framework made up of spars and ribs and covered with a skin. The particular design of a wing depends on many factors, such as the size, weight, speed, rate of climb, and use of the aircraft. The parametric modeling of an aircraft wing is based on parameters which concern the shape, dimension and configuration of the wing. From the large number of wing parameters, the following can be listed (Kachel, 2013): airfoil specification, root chord length, wing span, taper ratio (ratio between the tip chord and the root chord), sweep angle, dihedral angle, twist angle, incidence or setting angle, number of spars (and their configuration), number of ribs (and their configuration).

### 5.1. Parametric geometry model generation

The aircraft wing shape is mainly described by the geometric profile called “airfoil”. The airfoil profile can be defined from the coordinates of the upper and lower curves. These coordinates are provided in text files that can be read by the GRIP program as the first step (Fig. 4). If the coordinates are given in percentages, they must be multiplied by the root chord length as follows

$$x = \frac{x\%}{100}c \quad y = \frac{y\%}{100}c \quad (5.1)$$

where  $c$  is the value of the chord length.

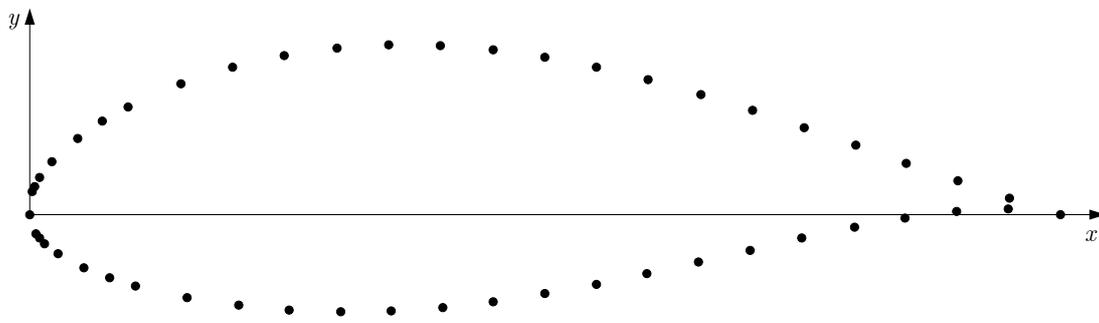


Fig. 4. Points defining of the aerodynamic profile

The developed aircraft wing modelling program uses the B-spline functions which make it possible to apply the parametric approach to the modelling process of virtual structures.

The B-spline curve has the zero value in all parametrisation subintervals, except for  $m + 1$ . Such curves can be defined recurrently in the following manner

$$N_{i,0}(x) = \begin{cases} 1 & x_i \leq x \leq x_{i+1} \\ 0 & \text{for other } x \end{cases} \quad (5.2)$$

the  $m$  degree B-spline in the interval is defined as

$$N_{i,m}(x) = \frac{x - x_i}{x_{i+m} - x_i} N_{i,m-1}(x) + \frac{x_{i+m+1} - x}{x_{i+m+1} - x_{i+1}} N_{i+1,m-1}(x) \quad (5.3)$$

When taking into consideration equations (5.2) and (5.3), we can find a clear form of the B-spline curve of any degree, which is directly used in algorithms designed for determining curves in CAD systems. When applying the presented approach, we can easily describe the shape of the aerodynamic profile used to construct the aircraft wing.

The tip airfoil profile is created by applying multiple transformation on the root profile:

- Translation by the wing span
- Scale by the taper ratio
- Multiple rotations by sweep, twist and dihedral angles.

These transformations are programmed in the developed GRIP program as follows:

```

...
CSYS0=&WCS
MAT1=MATRIX/TRANSL,0,span,0
upper_curve(n)=TRANSF/MAT1,upper_curve(1)
CS(n)=TRANSF/MAT1,CSYS0
&WCS=CS(n)
MAT2=MATRIX/SCALE,taper_ratio
upper_curve(n)=TRANSF/MAT2,upper_curve(n),MOVE
MAT3=MATRIX/ZXROT,twist_angle
upper_curve(n)=TRANSF/MAT3,upper_curve(n),MOVE
...

```

Next, the wing skin, spar and rib surfaces are generated from the tip and root profiles. The creation of the spars and ribs is programmed in loops, where the increment represents the rib/spar position. For example, the loop for creating rib surfaces from curves can be written as follows:

```

...
DO/rb1:,i,2,n-1
rib(i)=BSURF/CURVE,upper_curve(i),lower_curve(i)
rb1:
...

```

The following flowchart (Fig. 5) illustrates the structure of the GRIP program and shows the steps to follow in order to achieve the desired wing model.

In order to construct a correct FEM model without free edges, it is necessary to associate the mesh nodes on the surface edges. Thus, the skin, rib and spar surfaces must be split into small parts: we define the part of a rib between two neighbor spars, or the leading edge and the first spar, or the last spar and the trailing edge as a rib part. Also, the part of a spar between two neighbor ribs as the spar part and the skin part between two neighbor ribs and two neighbor spars as a skin part (Fig. 6)

The next step is to export the model as IGES file. As mentioned before, this is done by calling a VB journal that executes this task. To perform different tasks by programming, surface part labels are used as function inputs. Although, when the wing model is imported in MSC.PATRAN as a single IGES file, the surface parts are labeled as a whole and, therefore, labels can not be distinguished in order to operate on a specific group of parts. The solution is to export/import the model as three separate IGES files: “skin.igs”, “ribs.igs” and “spars.igs”.

## 5.2. Automated FE model generation

For the CAE component, the first task programmed in the PATRAN session file is to import the model IGES files one by one to identify the surface labels. For example, after importing the first file “skin.igs”, the number of surfaces imported can be determined, and thus it is known

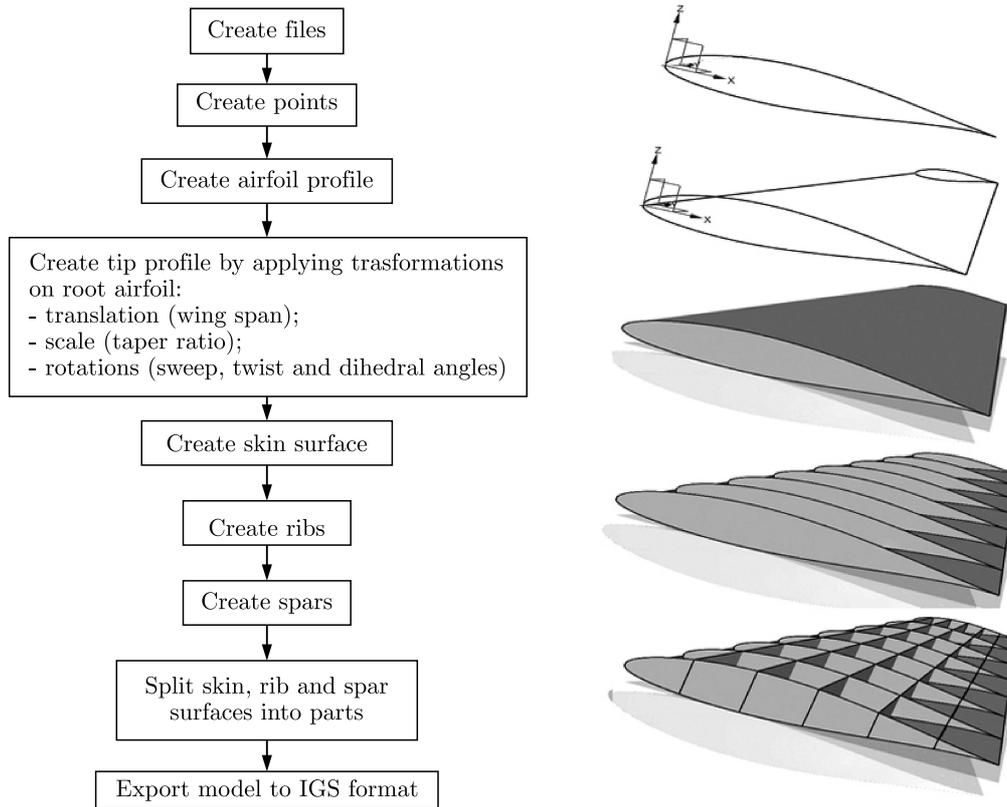


Fig. 5. GRIP program flowchart

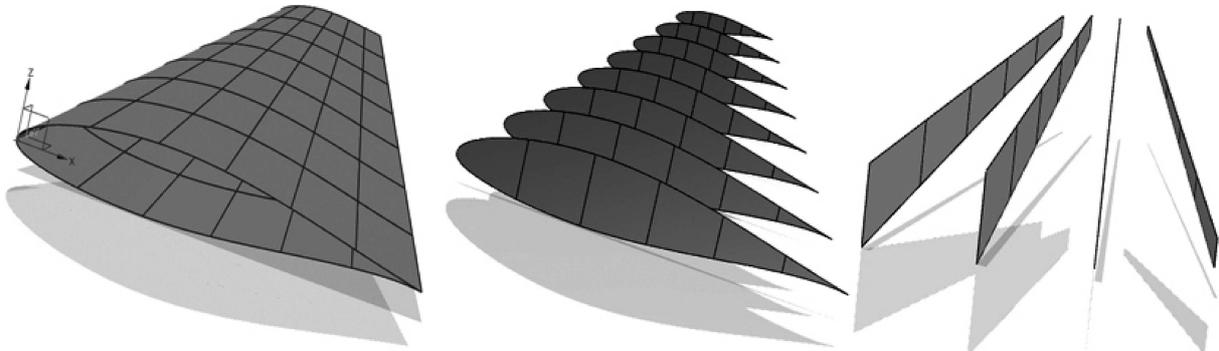


Fig. 6. Skin, spar and rib surfaces split into small parts

that the skin part labels are from 1 to the determined number, and so on. To illustrate more, a portion from the session file, showing PCL code used to perform this, is presented below:

```

...
INTEGER entities_imported(9)      $$ number of patran entities imported by
type
INTEGER N_skin_parts    $$ total number of skin parts
INTEGER N_spar_parts    $$ total number of spar parts
INTEGER N_rib_parts     $$ total number of rib parts
ugi_import_iges_v5("C:\skin.igs", 0,"default_group",...,entities_imported)
N_skin_parts=entities_imported(3)  $$ "3" for entity type: surface
ugi_import_iges_v5("C:\spars.igs", 0,"default_group",...,entities_imported)
N_spar_parts=entities_imported(3)

```

```
ugi_import_iges_v5("C:\ribs.igs", 0,"default_group",...,entities_imported)
N_rib_parts=entities_imported(3)
```

In addition to the total number of spar parts and ribs parts, it is necessary to determine the number of ribs and spars in the model, noted  $R$  and  $S$ , respectively. This can be performed using these equations

$$\begin{aligned} R &= \frac{N_{skin\_parts}}{2} - N_{spar\_parts} + 1 \\ S &= N_{rib\_parts} - \frac{N_{skin\_parts}}{2} - 1 \end{aligned} \quad (5.4)$$

Next, materials and properties are created based on analysis parameters (materials and thicknesses), and the surfaces are meshed after specifying a mesh seed to control element sizes and to ensure node association (equivalence verification). The PCL program developed is written in such a way that the aircraft wing geometric model can be modified and the mesh will be updated accordingly (Fig. 7). Finally, loads and constraints are applied and the analysis job is submitted to MSC.NASTRAN software.

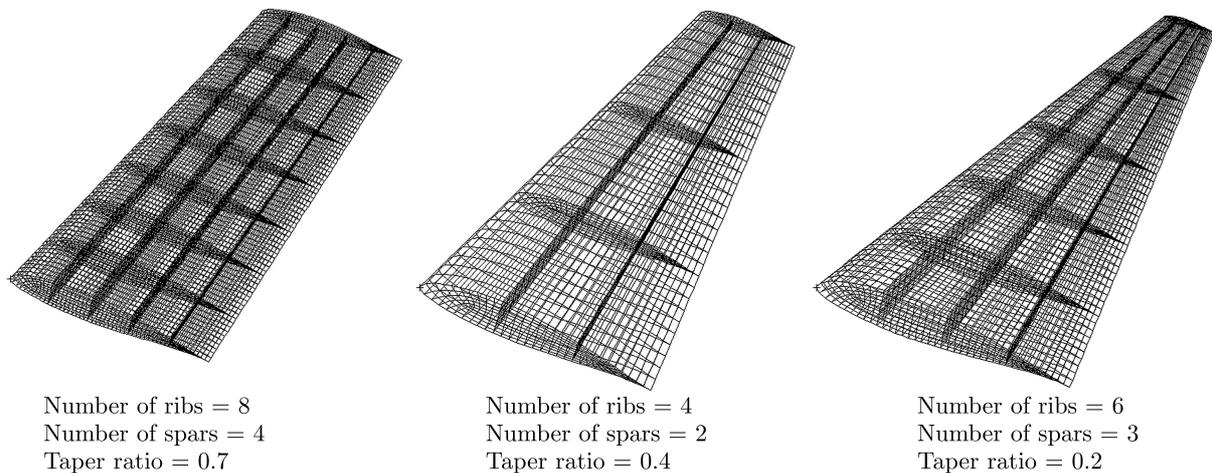


Fig. 7. Mesh update using the automated finite element modeling

After MSC.NASTRAN software finishes the analysis, output files are created. The MATLAB script reads the result files and looks for the specific result values, which can be displacements, stresses, natural frequencies, etc.

As analysis parameters, many choices are available, such as material specifications, thicknesses, load intensity (for static analysis), etc. For our simple case study, the aircraft wing is supposed to be clamped at its root. Both solution types “Linear static” and “Normal modes” can be supported by the presented integration procedure. For the static analysis, the wing is supposed to be subjected to a uniform pressure on its skin surface. Results desired can be either displacements or stresses of the wing structure due to the applied pressure for the static analysis, and natural frequencies for the normal mode analysis (Fig. 8).

### 5.3. Results and discussion

In order to simplify the use of the integration system, a MATLAB graphical user interface is implemented (Fig. 9). This GUI eases parameter value input and result access. The user can simply introduce the desired parameter values for modeling as well as for analysis, choose the analysis type and run the integration procedure. The script programs (GRIP program and PCL program) are then created according to parameter input values, and the software packages

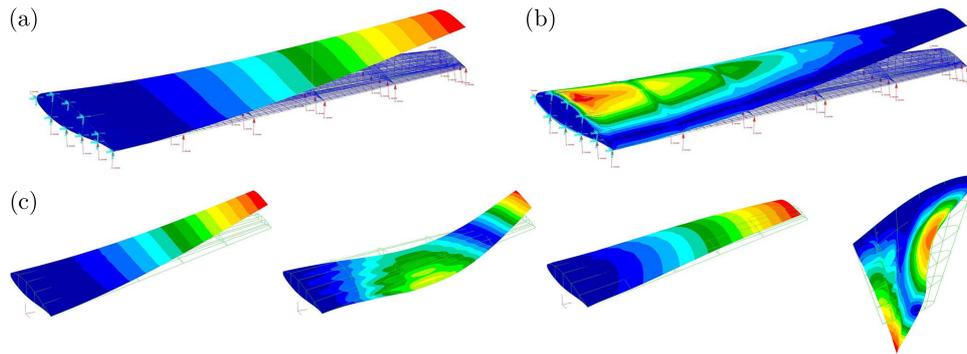


Fig. 8. Plots of the static and dynamic analysis results: (a) displacement result plot, (b) stress result plot, (c) normal modes

(SIEMENS NX and PATRAN/NASTRAN) are executed in the invisible mode to carry on the desired task. When the analysis is completed, the results as well as preview graphics are shown in the same interface. Accordingly, there is no interaction of the user with the design software tools, and the overall time of the modeling and calculation is only about 45 seconds. Thus, exploring the behavior of a new design case (different geometric/analysis parameter value) is no longer a time/effort consuming task due the automation capacity of the approach.

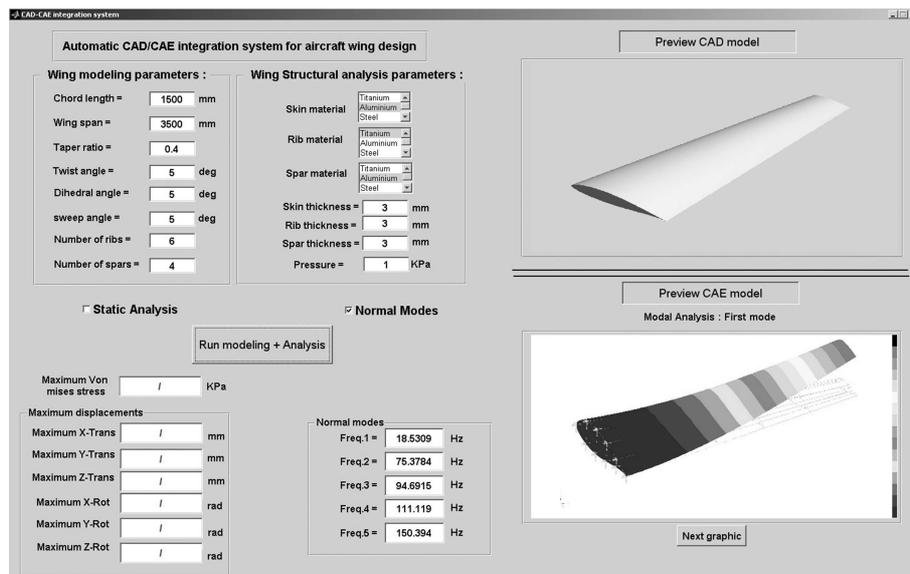


Fig. 9. A graphical user interface for the integration procedure

One other original quality of the developed framework is that it offers the ability to investigate the behavior of the structure with respect to changes of a chosen parameter. This is done by running the procedure in a simple loop, where the parameter value changes at each iteration. Therefore, the structure response can be related to its parameters, through approximate mathematical expressions using interpolation techniques, which allows the use of conventional optimization algorithms. The automated integration procedure is, consequently, indispensable for conducting an optimization task. The parameters, for modeling as well as for analysis, can play the role of design variables, and the objective function can be set as one of the result cases (minimize the maximum stress, maximize the lowest frequency value, etc).

In the present work, the procedure is used to emphasize the dependency of the wing structure maximal stress and first natural frequency on some of its parameters. One of the requirements for the structural design is to limit the maximal stress over the structure under critical load

cases. In order to predict the resistance of the wing structure under some specific load intensity, the maximum von Mises stress over the structure is calculated for different values of the wing parameters. This allows one to verify if the von Mises stress reaches the critical value, known as the yield strength, for the design cases described by the parameter values. Similarly, to prevent the structure from free vibrations and resonance damage, natural frequency dependency on the wing parameters must be determined. Figure 10 shows the effect of variation of the wing chord and the spar thickness on the maximum von Mises stress over the structure as well as the change in the first frequency value in function of the wing span and the taper ratio.

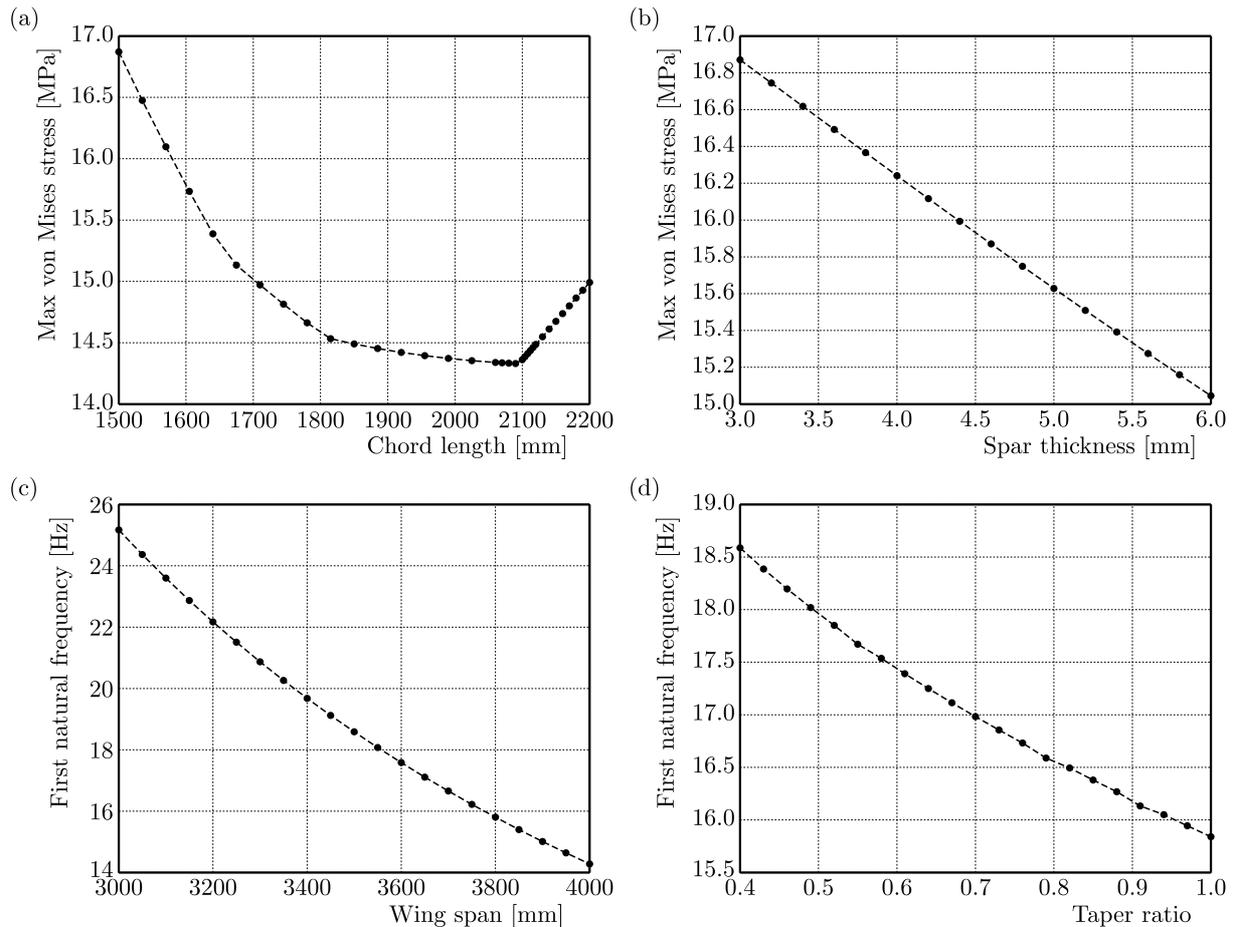


Fig. 10. Dependency of the wing behavior on its parameters: (a) von Mises stress in function of the chord length, (b) von Mises stress in function of the spar thickness, (c) first frequency in function of the wing span, (d) first frequency in function of the taper ratio

The presented figures show that some relations (e.g. Figs. 10b-10d) between the structure behavior and design parameter can be modeled by simple mathematical functions (linear, polynomial,...) and hence reduce the huge computational cost that the optimization may require. Some other relations (e.g. Fig. 10a) which present a discontinuity point out the usability of conventional optimization algorithms only after the fitting operation. Therefore, the presented approach proves its indispensability as a preparatory step for the optimization problem. Moreover, including geometric parameters (wing taper ratio, sweep back angle,...) as optimization design variables is not possible using the optimization module in CAE software. Thanks to the present methodology, the optimization can examine a large design space where both geometric and analysis parameters of the wing structure can be mapped for optimal values.

## 6. Conclusions

The CAD/CAE integration technique has become more and more popular in design process early stages as a tool for time and cost reduction. The idea behind this work is to develop a seamless automated integration procedure intended to ease design modification, explore a large design space in a relatively reduced time and to prepare for optimization activities. The developed procedure is based on popular commercial parametric CAD-CAE tools, SEMENS NX with its programming language GRIP and MSC.PATRAN/NASTRAN with its automation module PCL, and MATLAB as the implementing software. An aircraft wing structure has been the case to which the procedure was applied. The complexity of the considered case lies mainly in the encountered difficulties of programming the analysis tasks so as to be compatible with the changes of the geometry layout (change of the number of spars/ribs). In addition, acting on the imported entities (mainly surfaces) requires knowledge of their specific labels which are assigned automatically by MSC.PATRAN software, which complicates the programming task.

It has been shown that the presented approach is requisite for probing the design space and examine the structure behavior over it. Having the structure response in function of its parameters constructs a primary base to conduct an optimization problem without the need for excessive computational time. Thus the presented work serves as a solid ground for the implementation of a generic aircraft wing optimization framework.

## References

1. ASIYEDU Y., GU P., 1998, Product life cycle cost analysis: state of the art review, *International Journal of Production Research*, **36**, 4, 883-908
2. DIETRICH M., KĘDZIOR K., SKALSKI K., 1999, Design and manufacturing of the human bone endoprotheses using computer-aided systems, *Journal of Theoretical and Applied Mechanics*, **37**, 3, 481-503
3. EDKE M.S., CHANG K.H., 2006, Shape optimization of heavy load carrying components for structural performance and manufacturing cost, *Structural and Multidisciplinary Optimization*, **31**, 5, 344-354
4. GUJARATHI G.P., MA Y.-S., 2011, Parametric CAD/CAE integration using a common data model, *Journal of Manufacturing Systems*, **30**, 3, 118-132
5. KACHEL S., 2013, Optimization of wing parameters to achieve minimum weight at defined aerodynamic loads, *Journal of Theoretical and Applied Mechanics*, **51**, 1, 159-170
6. NAWIJN M., VAN TOOREN M.J.L., BERENDS J.P.T.J., ARENDSSEN P., 2006, Automated finite element analysis in a knowledge based engineering environment, *44th AIAA Aerospace Science Meeting and Exhibit*, 9-12
7. PARK H.-S., DANG X.-P., 2010, Structural optimization based on CAD-CAE integration and metamodeling techniques, *Computer-Aided Design*, **42**, 10, 889-902
8. PENG L., LIU L., LONG T., YANG W., 2014, An efficient truss structure optimization framework based on CAD/CAE integration and sequential radial basis function metamodel, *Structural and Multidisciplinary Optimization*, **50**, 2, 329-346
9. ROCCA G.L., VAN TOOREN M.J.L., 2009, Knowledge-based engineering approach to support aircraft multidisciplinary design and optimization, *Journal of Aircraft*, **46**, 6, 1875-1885
10. ROWE J., 2006, *Virtual Reality*, Cadalyst
11. ROY R., HINDUJA S., TETI R., 2008, Recent advances in engineering design optimisation: Challenges and future trends, *CIRP Annals-Manufacturing Technology*, **57**, 2, 697-715

12. SAPUAN S.M., OSMAN M.R., NUKMAN Y., 2006, State of the art of the concurrent engineering technique in the automotive industry, *Journal of Engineering Design*, **17**, 2, 143-157
13. SOHAIB M., 2011, *Parameterized Automated Generic Model for Aircraft Wing Structural Design and Mesh Generation for Finite Element Analysis*, Master Thesis, Linköping University, Linköping, Sweden
14. SU D., QIN D., 2003, Integration of numerical analysis, virtual simulation and finite element analysis for the optimum design of worm gearing, *Journal of Materials Processing Technology*, **138**, 1, 429-435
15. WANG D., HU F., MA Z., WU Z., ZHANG W., 2014, A CAD/CAE integrated framework for structural design optimization using sequential approximation optimization, *Advances in Engineering Software*, **76**, 56-68

*Manuscript received January 24, 2016; accepted for print September 15, 2016*