KNOWLEDGE-BASED ENGINEERING SYSTEM FOR DETERMINING THE SHAPE OF FUEL DISTRIBUTION IN AN AIRCRAFT WING

Joao Paulo Rebucci Lirani, Product Development Eng. , jp.lirani@embraer.com.br
Cristina Ferreira de Paula, DSc, cristina.paula@embraer.com.br
Fernando Lukas Miglorância, Product Development Eng. , fernando.miglorancia@embraer.com.br
Empresa Brasileira de Aeronáutica SA - Embraer

Prof. Luís Gonzaga Trabasso, Ph.D. , gonzaga@ita.br
Instituto Tecnológico de Aeronáutica - ITA

Abstract. *During the aircraft development process, one of the most important steps for fuel system design is to determine how the fuel is distributed in the wing fuel tank according to different aircraft attitudes and phases of operation. The purpose of this paper is to present the development of an application software for determining the shape of fuel distribution in an aircraft wing fuel tank in steady state regime considering the aircraft's acceleration and attitude. For the development of this application, Catia™ V5 Knowledgeware tools were associated with Visual Basic programming. Since there are convergence and performance problems, a study for determining the best numerical optmization method was also conducted. Such study represents the most significant theoretical contribuition of this paper. On the other hand, the Catia™ V5 Knowledgeware tools allow geometric parameterization, the creation of design rules and reactions, and knowledge encapsulation in templates. In the case analysed herein, the inputs are: wing geometry (solid), aircraft axis system, roll and pitch angles, accelerations, fuel density and target weight. Once the shape of the fuel is calculated, it is possible to locate wet and dry systems and determine air/fuel, fuel/tank surfaces for different phases of flight. To demonstrate the performance of the proposed application, some examples are analysed and presented. From these outputs, it can be inferred that major gains for industry are reduction of the development cycle and template re-use in different cases and aircrafts, as well as protection and standardization of corporate knowledge.*

Keywords: *parametric geometry, CATIA V5, fuel system, knowledge-based engineering, templates, optimization methods*

1. INTRODUCTION

During aircraft preliminary design, an important phase of its development is the design of the fuel system. Fuel system determines aircraft flight performance and is related to many aircraft safety parameters. It is very important to determine the amount of fuel in the wing tanks during different phases of the operation of the aircraft, so it is possible to locate wet and dry systems and determine air/fuel, fuel/tank surfaces, accordingly.

The fuel in an aircraft is stored in tanks usually in the wings (inside the wing-box), as shown in Fig.1. In steady-state flight (once fuel is in equilibrium), the shape of fuel distribution in the tank may be represented by a plane, reffered as the fuel plane. The fuel plane is parallel to the ground in unaccelerated movement, and varies with the different accelerations. As the aircraft change its attitude during different stages of flight, so does the fuel shape inside the wing tank. For the present work, change of aircraft attitude is considered in terms of roll angle (rotation around longitudinal axis - x) and pitch angle (rotation around transversal axis - y).

As the wing tank is a complex shape with internal structure, it is very difficult to predict analiticaly the volume /weight of fuel in each the different operations of the aircraft. The purpose of this paper is to present the development of an application for determining the shape of fuel distribution using a Knowledge-Based Engineering approach.

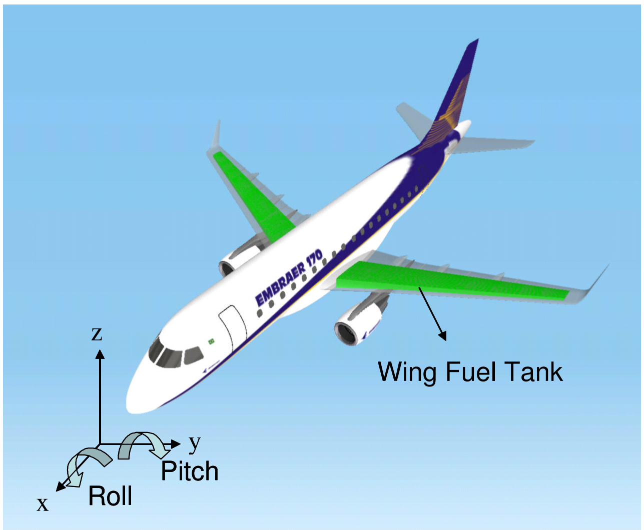


Figure 1. Wing Fuel Tank and aircraft axis

Briefly, Knowledge-Based Engineering is a technology for capturing best practices in a project and reusing it in a new one. The success of a KBE application depends heavily on whether it has accurate knowledge of a problem and how well the knowledge is presented and applied to each problem, as presented by Moka Project (2003). According to Choi *et al* (2005) CATIA™ V5 is a state-of-the-art CAD system implemented throughout the aerospace industry for component and systems design. Besides, it has a KBE environment as a standard workbench using Visual Basic as a programming interface and provides tools to construct template features for re-using knowledge. CATIA™ V5 tools to facilitate such activities are collectively termed 'Knowledgeware', and it applies to design rules and processes which can be formally captured by hard-coding into a CAD model for re-use in future projects. These models may be delivered to general use to rapidly create parts and assemblies; this is the basis of process 'templating'. By using the CATIA™ V5 Product Knowledge Template functionality, it is possible to create a model of such a feature with in-built code controlling what form it takes when inserted into a model, depending on the geometry (or references) used to position it. Template models should capture best practice and assure consistency of design. The use of templates reduces the development time and increases the profits with the product quality. Once created, the template model contains the building blocks for a variety of design iterations. The template and linked tooling models are adjusted and the resulting collection of models, although basic, meets all specified requirements. Furthermore, once in place it is only necessary to validate and maintain one approach.

Embraer has today several Knowledge-Based Engineering applications, and in particular, it may be cited a multidisciplinary system for Wing Structure Design (WSDS). This Knowledge-Based Engineering application has as purpose to assist engineers with the design of a complete wing box. WSDS applicability ranges from definition of general wing layout to detailed wing geometry, obtained through a series of structural analyses, both analytical and computational through finite element analysis. Also, reports and results, e.g. CAD files, can be generated throughout the design procedure, more detail Faria *et al*. (2002), Moura *et al*. (2003) and Paula *et al*. (2004).

The paper is structured in 3 main sections: Resources and Methods, Discussion of Results and Final Considerations.

2. RESOURCES AND METHODS

2.1. Framework for development

The chosen framework for developing the KBE application described herein was CATIA™ V5 as the basic geometric core allied with VisualBasic programming (VB) and V5 Knowledgeware tools. VisualBasic is the programming language used for implementing the user interface (UI), numerical methods, manipulating parts geometry and binding them all together. CATIA V5 API (Application Programming Interface) provides the Catia V5 and VB communication, as in Clark and Schneider (2003). The integration of these tools and general system algorithm is depicted in Fig. 2:

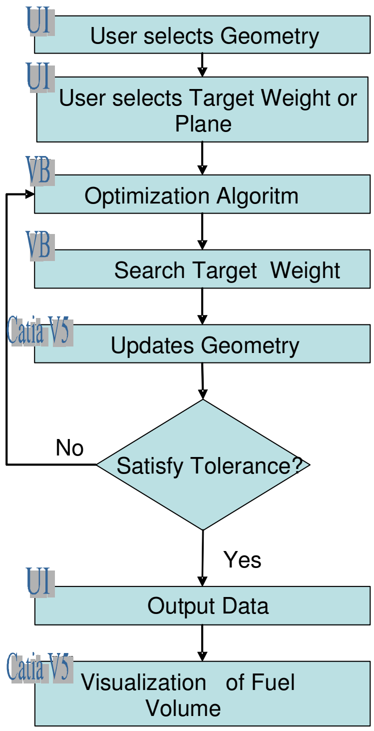


Figure 2. Algorithm of the application

The User Interface (UI) is used for collecting all input data, which in this case consist of:

- Wing tank geometry (Volume)
- Roll angle (degrees)
- Pitch angle (degrees)
- Longitudinal acceleration (m/s²)
- Transversal acceleration (m/s²)
- Target Fuel weight (kg)
- Fuel density (kg/m³)

Once the wing tank volume is input, an User-Defined Feature (UDF) will be instantiated via VB, representing the fuel geometry. Geometrical operations are all performed inside the UDF template (more details of UDF in item 2.2), thus, the code communicates with it passing the following parameters: roll angle, pitch angle, accelerations and plane offset (mm). Plane offset is the distance between line of fuel and the lowest point of a wing. As it cannot be calculated through analytical methods, an iterative process is used to determine it.

The algorithm must pass a plane offset representing fuel height for Catia V5 UDF, and with this new offset, the geometry is updated which implies in a new volume and a new fuel weight. This new weight is passed to VB algorithm and compared to the Target Weight. . This interactive calculation procedure stops when the difference between the CATIA and target weight is less than a given tolerance. Several numerical methods have been analyzed for faster convergence of the described process.

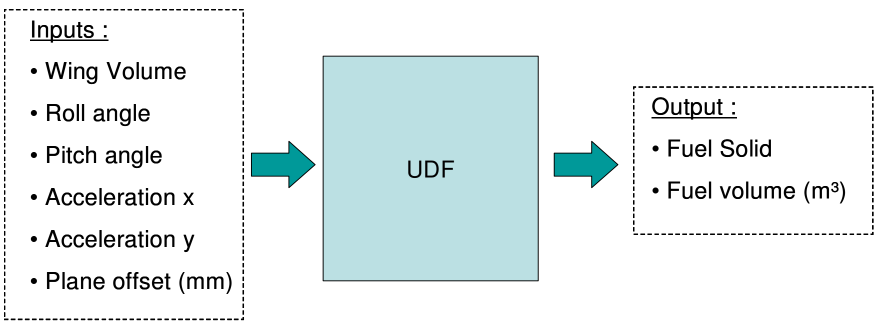


Figure 3. Description of the User-Defined Feature (UDF)

2.2. User-Defined Feature (UDF)

Initial attempts to manipulate geometry to obtain fuel geometry involved geometrical operations programmed in VB. An alternative for this procedure is to use a Knowledgeware tool available in Catia V5: an User-defined Feature (UDF). An UDF is a set of geometrical operations that are encapsulated in an easy-to-use template.

Once the UDF is built, its use is similar to a template, where inputs are given by the user, all geometrical operations are performed in realtime (with better performance than VB) and the result geometry is embeded in a icon in Product Tree like any other built-in feature (such as pad, hole, chamfer, etc) - thus the name user-defined *feature*.

For calculating the fuel weight, UDF inputs consisted of roll and pitch angles, accelerations and the plane offset. For a given user request (a certain airplane attitude and a target fuel weight), all input parameters remain the same and only plane offset varies in an iterative process. The UDF is constructed in a very interactive way, where coding is substituted by geometrical operations executed in a CAD system, as shown in Fig. 4. The geometrical operations for the present case consist of the following steps:

a) Wing geometry is input
b) A particular equation corrects acceleration in terms of roll and pitch angles
c) Rotates plane due to roll influence(b) , based on corrected roll parameter
d) Rotates plane due to pitch influence (b), based on corrected pitch parameter
e) Determines gravity direction, normal to plane (d)
f) Determines extremum point (lowest) in gravity direction
g) Determines the highest point in gravity direction (e). This maximum offset (maxoffset) will be used in VB
h) Creates plane normal to gravity direction, passing thru point
i) Creates a new plane, parallel of plane (h) , which offset is controlled by parameter
j) Splits wing solid with fuel plane (i)
k) Calculates the volume and weight of fuel solid and store in parameters

As described above, the influence of acceleration is decomposed in increments of roll and pitch angles:

$$\Delta Roll = \arctan\left(\frac{Ax}{9.81}\right) \quad (1)$$

$$\Delta Pitch = \arctan\left(\frac{Ay}{9.81}\right) \quad (2)$$

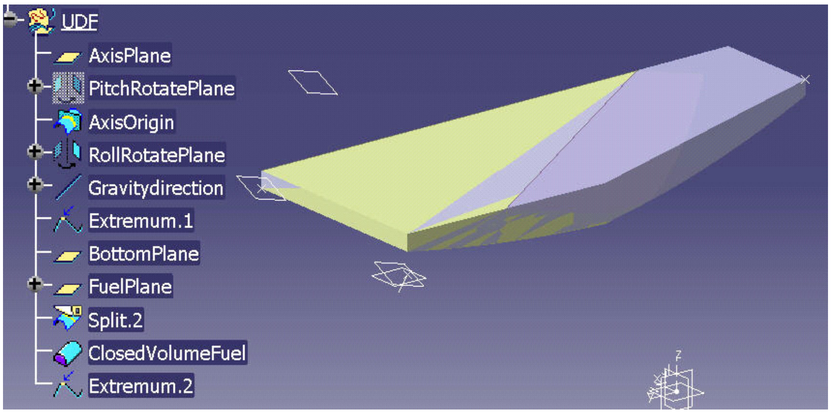


Figure 4. UDF construction

2.3. Numerical Methods

Basicaly, the mathematical problem may be resumed to finding the zero of the function

f(x) **= Weight**(x) **– TargetWeight** (3)

where:

x is the height of fuel, represented by the offset of a plane (mm), and varies from zero to a maxoffset value, given by the UDF (item 2.2, parameter g)

Weight(x) returns the weight of fuel for a given height. This is obtained with interaction with Catia, as shown in Fig.2

TargetWeight is the weight of fuel set by the user

A number of different numerical methods have been analyzed to solve this problem. They are: Gradient method, Golden Section, Newton-Rhapson and implementation of initial value interpolations, summarized as follows.

2.3.1. Product Engineering Optimizer (PEO)

CATIA™ V5 has a specific module for optimization (Product Engineering Optmizer-PEO), which was used initially for solving Weight vs Height iterations. The chosen optimization algorithm was a gradient method. The performance in runtime was satisfactory, around 4s per run, but since the application has to interact with VisualBasic, the V5 API had to be used. Unfortunately, some parameters that control the optimization convergence tolerance were not available in the API. Thus, the application used the default "high precision" value, which kept the optimization running for a tolerance of 0,0001 kg and consequently, it took around 45s for convergence. User requirements for the system are to present user interaction of *realtime* feel, so a maximum time of 2,5s per interaction has been established and tolerance of 0,1kg for the weight determined by the user (Target Weight). Run times are measured in a personal computer Intel Pentium IV with 1 Gb of RAM as reference. This lead to the analysis of other numerical methods which could meet the user requirements implemented in VB.

2.3.2. Golden Section

The first method tested in this work in VB was the Golden Section Method. Roughly it may be said that the Golden Section Method for numerically searching the root of a equation is very similar to the Bissection Method. Mathematicaly, though, it is proven that the Golden Section Method is more efficient than the Bissection Method, as referred in Venkataraman (2002). In short, the method can be summarized by the Fig. 5:

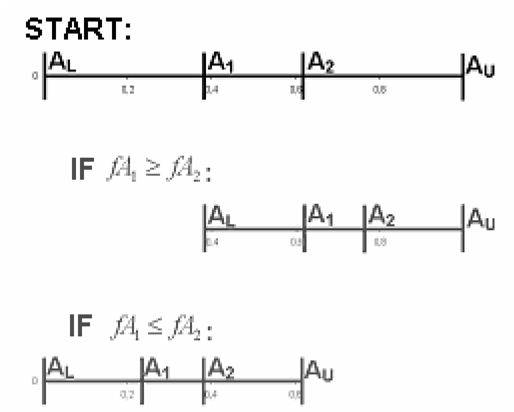


Figure 5. Golden Section Method

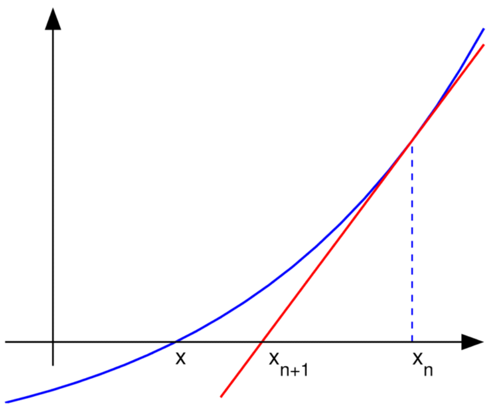
A very important fact to be underlined is that after each interaction the size of the search interval is reduced to 0.61803 of the former size. This leads to a stop criteria that, knowing the size of the initial search interval and fixing the size of the final search interval to a value less than ´TOL´, determines the maximum number of iterations (nevertheless, it is better to check at each iteration if any of the values f(Al), f(A1), f(A2) or f(AU) is close enough to zero so that the search can be finished) .

As a summary of the results obtained with this method, after some tests with several aircraft attitudes, it was found that the method converged in aproximatedly 25 iteractions and a total time of 7.5 seconds. A very good characteristic of the method was its robustness, since its convergence is mathematically guaranteed.

Even though the results found were very good, there was still the need of a response time less than 2.5 s, so other numerical methods had to be analyzed.

2.3.3. Newton-Rhapson

According to this method, the search begins with a initial guess x_0 and proceeds as shown in Fig. 6:



$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Figure 6. Newton-Rhapson Method

As for the computational implementation of these methods, two aspects must be stated:

- the exact (mathematical) value of the slope of the line tangent to the function was aproximated by a secant one;
- whenever the values of x_n exceeded the search limits (0<x<maxoffset), they were ´brought back´ (that is, if x_n> maxoffset, then x_n= maxoffset, and if x_n< 0 then x_n= 0).

As a summary of the results obatained for several airplane attitudes, it was found that the method converged very fast (around 2-3 s) for some cases, but also diverged several times. To understand why it happened, several studies were carried out to determine the typical shape of the curve "Weight vs. Height" (more details in next item). Due to this ´S-shaped´format of the curve, the method several times was trapped into a loop as shown in Fig. 7:

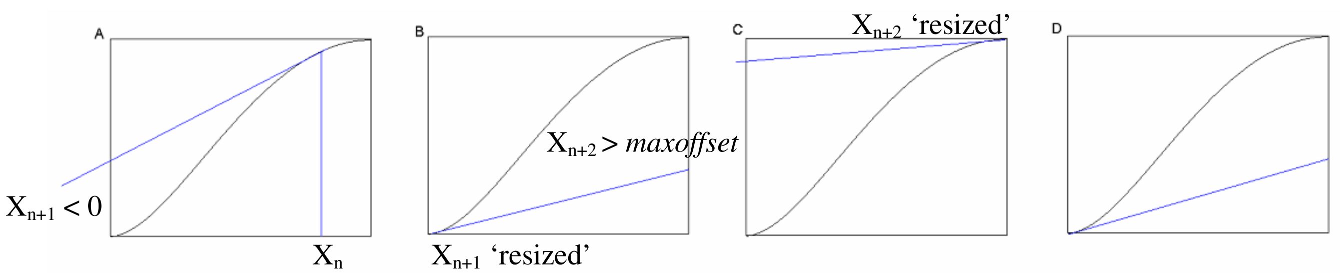


Figure 7. Case of divergence of Newton-Rhapson for the problem

This loop situation can be described as follows: in a given iteration n, the value of x_n falls out of the interval [0 maxoffset], as shown in Fig. 9-A, where x_{n+1} <0; x_{n+1} . Then it is resized and made equal to zero, but, as the tangent to the curve at this point is almost horizontal, x_{n+2} would be greater than the maxoffset (B), being, by its turn, resized to maxoffset. Again, the tangent is almost horizontal, making $x_{n+3} < 0$, and the the loop goes on.

In short, Newton-Rhapson was found to be an inadequate to solve the problem, since it was very fast, but did not guarantee convergance.

2.3.4. Weight vs Height curves for Fuel distribution

To investigate the divergence of Newton-Rhapson method for solving this problem, a detailed study of Weight Height curve (that is, the weight of fuel inside the wing for a given offset value) was conducted. More than 1000 runs for different aircraft roll and pitch angles, and different wing boxes were studied. It must be stated that without the aid of a integrated system linking numerical methods (implemented in VB) and CAD system, perfomed in batch mode, the generation of this amount of data would not be possible in the timeframe of this paper.

Analizing the results obtained from CATIA™ V5, it was observed that the curves had a characteristic ´S-shaped´ format, as shown in Fig. 8. This is confirmed in a similar study done by Budd and Slack (2007) that found a similar ´S-shape´, obtained through a more mathematical oriented approach, using H –adaptivity with piecewise cubic polynomials.

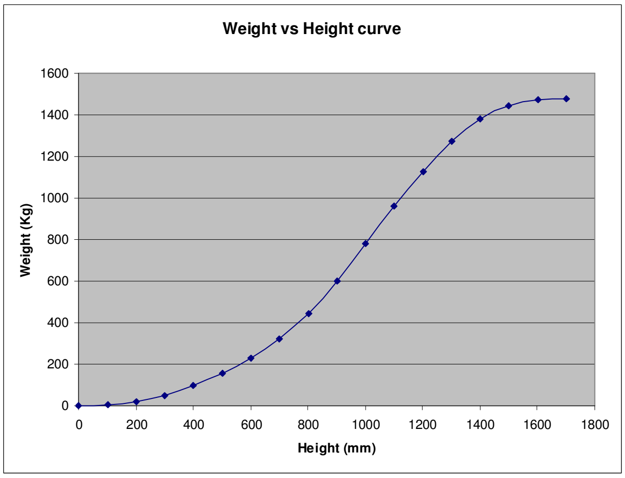


Figure 8. Height – Weight curve for a typical wing fuel tank

To obtain the characteristic shape of the problem in this case study was decisive for determining the best numerical method for solving the problem with best performance and robustness.

2.3.5. Golden Section with linear interpolation for Initial Value attribution

As the Newton-Rhapson method was not suitable for the problem, an analysis was made by using the Golden Section Method and introducing an initial guess based on a linear interpolation, as shown in Fig. 9:

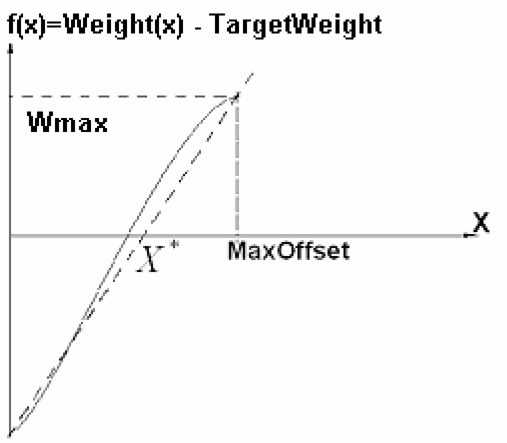


Figure 9. Golden Section with Linear initial value attribution

It was now achieved good robustness and speed in the convergence, once the method converged usually in 15 iterations, in an average time of 5 seconds.

2.3.6. Golden Section with Quadratic Interpolation for Initial Value attribution

A further improvement in the initial guess was made by interpolating f(x) with a quadratic function and then obtaining the root x* of this quadratic function to be used as a initial guess for the Golden Section Method.

The method to calculate the coeficients of this quadratic function is explained below.

2.3.6.1 Fitting the quadratic curve that best interpolates the function

It is proposed to interpolate the curve f(x)=Weight(x) in the interval [0, xmax] through the equation

$$\mathbf{a}x^2 + \mathbf{b}x + \mathbf{c} \quad (4)$$

(obs: xmax= maxoffset).

The first step to fit the curve is to adimensionalize the problem:
- x , originally representing offset is substituted by x= offset/(maxoffset/4)
- f represents WeightInPoint/ReferenceWeight, where ReferenceWeight = Weight(maxoffset/4)) = weight in ¼ of interval [0, maxoffset]

Briefly, the fitting function must pass through the following points (f2=Weight(x=2maxoffset/4)/ Weight (maxoffset/4) and f3=Weight (x=3maxoffset/4)/ Weight (maxoffset/4):

Table 1. Point Position vs Weight

Point Position	x	Weight	f
Maxoffset/4	1	Weight (maxoffset/4)	1
2 Maxoffset/4	2	Weight (x=2maxoffset/4	f2
3 Maxoffset/4	3	Weight (x=3maxoffset/4	f3

In other words, this adimensionalize the problem:

Table 2. Points for fitting function

Point	x	f
P1	1	1
P2	2	f2
P3	3	f3

In terms of equations, using eq(4) for the points described in Tab. 1 gives:

$$\begin{array}{lll} \text{Point P1} \rightarrow a.1^2 + b.1 + c = 1 & \rightarrow & a + b + c = 1 \\ \text{Point P1} \rightarrow a.2^2 + b.2 + c = f_2 & \rightarrow & 4a + 2b + c = f_2 \\ \text{Point P1} \rightarrow a.3^2 + b.3 + c = f_3 & \rightarrow & 9a + 3b + c = f_3 \end{array} \quad (5)$$

or in matrix form,

$$\begin{bmatrix} 1 & 1 & 1 \\ 4 & 2 & 1 \\ 9 & 3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{Bmatrix} 1 \\ f_2 \\ f_3 \end{Bmatrix} \quad (6)$$

Solving this system by using MatLab™, yields:

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = inverse \begin{bmatrix} 1 & 1 & 1 \\ 4 & 2 & 1 \\ 9 & 3 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 0.5 & -1 & 0.5 \\ -2.5 & 4 & -1.5 \\ 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ f_2 \\ f_3 \end{bmatrix} \quad (7)$$

or more explicitly:

$$\begin{aligned} a &= 0.5 - f_2 + 0.5 f_3 \\ b &= -2.5 + 4 f_2 - 1.5 f_3 \\ c &= 3 - 3 f_2 + f_3 \end{aligned} \quad (8)$$

It must be noted that, to find x*, the main objective is not to solve the equation

$ax^2 + bx + c = 0$, but $ax^2 + bx + c = target$, where target= TargetWeight/Weight(maxoffset/4).

Although the common sense could suggest that with a more accurate initial guess the convergence would be faster, this was not verified. What actually happened was that the convergence became slower; it was due to the fact that one of the extremities of the search interval in the Golden Section Method now became fixed in x*, while the other extremity was slowly brought near to the root of f(x). The value of x* obtained trough this quadratic interpolating, thought, was really accurate, which led to the next method studied

2.3.7. Newton-Rhapson with Quadratic Interpolation for Initial Value attribution

The method consists of using the Newton-Rhapson method taking the value of x* calculated by the previous method as initial guess. The method was very robust and fast, converging in a time typically less than 1.5 seconds, which led to the adoption of this method as the most adequate for solving the case study problem.

3. DISCUSSION OF THE RESULTS

The first step for implementing the system was to build a framework consisting of a integration of CAD, knowledge templates, user interface and numerical methods. A typical fuel distribution for a generic wing tank is shown in Fig. 10, along with the instantiated template (UDF) as presented to the final user. After developing such framework, the main focus was to determine the best numerical method to find a response with required precision and performance.

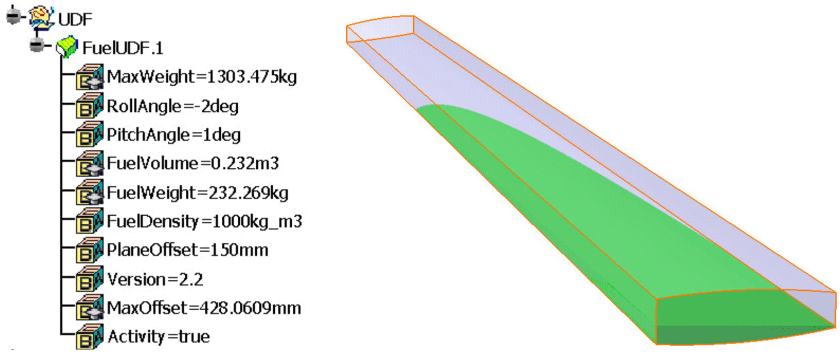


Figure 10. Instantiated UDF and typical fuel distribution visualization

As seen earlier, the initial system design involved the use of a built-in optimization module from CATIA (PEO). Due to a limitation of unavailable option for integration with VisualBasic, the results were convergence time of 45s, which did not fulfill system requirements. As presented in item 2.3, a series of numerical methods implemented in VB were used to solve the problem. The first method studied was Golden Section. It converged in around 25 iterations in 7,5s. – a summary of convergence curves for the studied methods is presented in Fig. 11. The next choice was Newton-Rhapson, which presented a much better convergence in 2-3 s but did not guarantee convergence for all cases due to the particular shape of the curve, as shown in Fig. 11a.

To improve performance of Golden Section method, an initial guess estimated by linear interpolation was adopted. This further improved convergence to 15 iterations and 4,5s average time. A second attempt to improve initial guess by increasing the order of approximation curve to a quadratic did not achieve the desired results.

After a detailed study of the Height – Weight function, a characteristic S-shape curve was found, which led to the choice of a tailored solution for this specific problem.

The first 3 iterations are used to fit a quadratic curve that is used to determine the initial guess, as shown in Fig. 11d. This initial value is found to be very close to the actual final value, so that few further interactions are needed and this also guaranteed convergence for Newton-Rhapson in all cases. This numerical method converged in around 6 iterations and 1,5 s, representing the best numerical method as it fulfilled the user requirements.

It is once again observed that in order to find the most suited numerical method, a detailed study of the problem (Weight Height curve) and its particular characteristics was determinant to obtain a customized and satisfying solution.

The use of a customized solution improved substantially the results, since first convergence times were 45 s and Newton-Rhapson with quadratic approximation reduced convergence to 1,5s, achieving the user requirements of 0,1 kg precision and real-time feel (time response of less than 1,5s).

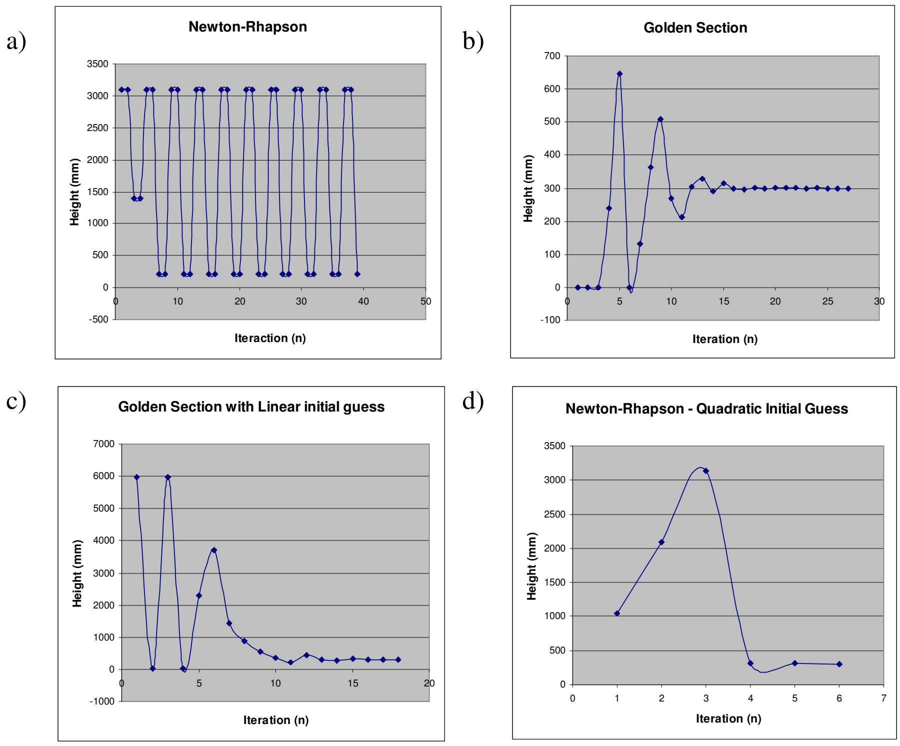


Figure 11. Convergence curves for the studied methods

4. FINAL CONSIDERATIONS

According to the obtained results, the fastest and more robust method was Newton-Rhapson with Quadratic Interpolation for the Initial Value Attribuition. It was verified by the users that the time spent for an intire analysis was an hour without the use of the template, 10 minutes with the template using the UDF and only 2 seconds with the template in addition to the chosen method.

It is remarkable that the use of the Knowledge-Based Engineering approach allows high quality in the analysis of the results and protection of the knowledge. Its usage, alongside the numerical method that has been developed, greatly reduces the development time and allows a broader vision of the problem by the designer.

5. REFERENCES

Choi, J. W.; Kelly, D.; Raju, J. and Reidsema, C., 2005, “Knowledge-Based Engineering System to Estimate Manufacturing Cost for Composite Structures”, Journal of Aircraft, Vol. 42, No. 6.

Clark, N., Schneider, T., 2003, “Getting Started with KBE: Working With the CATIA / Visual Basic Interface”, Vought Aircraft Industries, <http://www.voughtaircraft.com/ntcoe/data/presentation_schneider_111403.pdf>

Faria, A.R., Paula, C.F. and Silva, P.A., 2002, “Preliminary Wing Design Optimization: a Knowledge-based Engineering Aplication”, SAE 2002.

Moura, A.C., Paula, C.F., Liserre, L. and Silva, P.A., 2003, “Melhoria da qualidade do produto e redução de seu ciclo de desenvolvimento: Aplicação EBC na geração de modelos CAD-3D”, SAE 2003.

Paula, C. F., Junior, E. S., Liserre, L., Machado, G.C., Silva, P.A., 2004, “Redução do Ciclo da Análise Estrutural de Asas de Jatos Comerciais empregando-se a Engenharia Baseada no Conhecimento”, Proceedings of the XXXI Jornadas Sud-Americanas de Ingeniería Estructural.

Moka Project “MOKA – Framework for Structuring and Representing Engineering Knowledge”, 2003, <http://www.kbe.coventry.ac.uk/moka/>,

Slack,C., Budd, C., 2007, “Height-volume characteristics of a fuel tank”, ESGI 2006, <http://www.smithinst.ac.uk/Projects/ESGI56/ESGI56-AirbusFuelTanks/index_html>

Venkataraman, P., 2002, “Applied Optimization with MATLAB Programming”, Ed. John Wiley & Sons, New York, 398 p.

6. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.