ABCM Symposium Series in Mechatronics - Vol. 3 - pp.346-354

GRAPHIC ROBOT SIMULATION FOR THE DESIGN OF WORK CELLS IN THE AERONAUTIC INDUSTRY

Aguiar, Adriano José Cunha de, ajca@ita.br
Silva, Alex Sandro de Araújo, alexs@ita.br
Villani, Emília, evillani@ita.br
Comando-Geral de Tecnologia Aeroespacial – CTA
Instituto Tecnológico de Aeronáutica – ITA
CCM – Centro de Competência em Manufatura
Divisão de Engenharia Mecânica-Aeronáutica
Pça. Mal. Eduardo Gomes, 50 - Vila das Acácias
CEP: 12228-900 – São José dos Campos/SP, Brazil

Abstract. *This paper investigates the use of graphic robot simulation for designing work cells in the aeronautic industry. It considers the riveting process of aircraft structures such as fuselage sections. It proposes and analyzes several configurations verifying accessibility and collision avoidance. Simulation results demonstrate that the access to some regions of the fuselage is very difficult or even impossible in some cases.*

Keywords*: graphic simulation, industrial robots, riveting, aerostructures.*

1. INTRODUCTION

Today, a new global scenario is emerging for aircraft industries. New powers, such as China, are investing on the creation of national aircraft industries, which, in the near future, must represent a strong competition to the current industries. In order to face these new challenges, the Brazilian aircraft industry must guarantee its competitiveness by investing in the research of new technologies and production strategies.

Concurrently, the manufacturing organizations around the world, not only in the aeronautics sector but also in other sectors such as automotive, compose a hyper-competitive environment that forces a changing of paradigm. The traditional approach of investing in projects that focus on punctual improvements in the production line is quickly reaching diminishing returns in the ability to reduce costs and improve productivity (Cimdata, 2005). Manufacturing organizations must focus on the introduction of new paradigms, such as Digital Manufacturing and Product Lifecycle Management (PLM).

Product lifecycle management (PLM) is the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service and disposal (Cimdata, 2006). PLM is one of the four cornerstones of a corporation's information technology structure. All companies need to manage communications and information with their customers (CRM - Customer Relationship Management) and their suppliers (SCM - Supply Chain Management) and the resources within the enterprise (ERP - Enterprise Resource Planning). In addition, manufacturing engineering companies must also develop, describe, manage and communicate information about their products (PLM) (Evans, 2004).

The benefits of PLM include (Evans, 2004, Day, 2002):

- Reduced time to market;
- Improved product quality;
- Reduced prototyping costs;
- Savings through the re-use of original data;
- A framework for product optimization;
- Reduced waste;
- Savings through the complete integration of engineering workflows.

The second paradigm considered here is Digital Manufacturing. 'Digital Manufacturing' is a term that has been used by software vendors and machine marketers for years, but only recently it has become a new concept for manufacturing systems. The basic idea behind it is to move bits instead of moving atoms. Digital manufacturing is the ability to describe every aspect of the design-to-manufacture process digitally—using tools that include digital design, CAD, CAM, analysis software, simulation, and so on.

Merging the two paradigms, Digital Manufacturing can be seen as a set of tools and methods that support the implementation of PLM. This concept is illustrated in Figure 1.

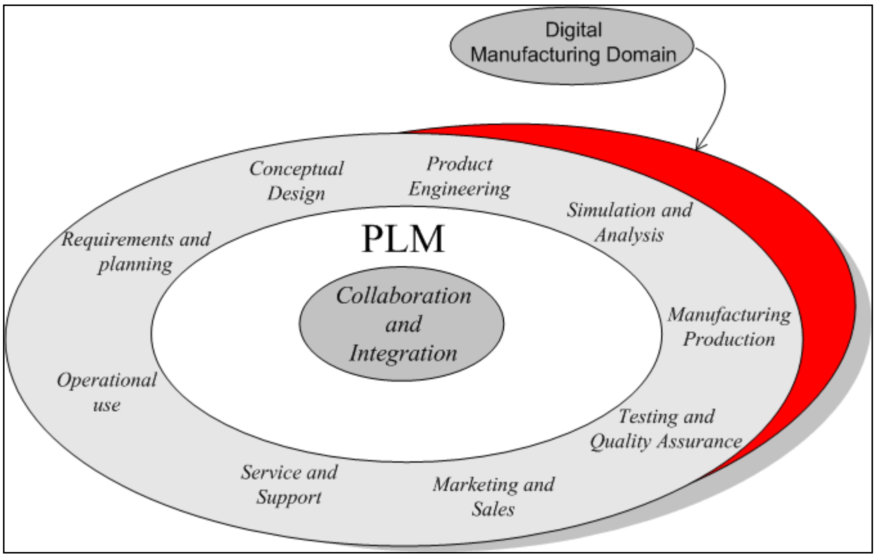

Figure 1. Digital Manufacturing in a PLM environment.

Among the many facets of Digital Manufacturing in a PLM environment, this paper approaches the simulation of robotics cells. Although the use of robots in the automotive industry is a well established reality, the same can not be said for the aeronautics industry.

The introduction of robots in manufacturing industries has many advantages. Basically, in relation to human labor, robots work to a constant level of quality. Waste, scrap and rework are minimized. They can work in areas that are hazardous or unpleasant to humans. Robots are advantageous where strength is required, and in many applications they are also faster than humans. Also, in relation to special-purpose dedicated equipment, robots are more easily reprogrammed to cope with new products or changes in the design of existing ones. Dedicated equipment usually requires expensive strip-down and rebuild in these situations and often has to be discarded as obsolete. Furthermore, when changing from manual to robotic methods, the product components will often have to be redesigned to provide simplicity of presentation, positive gripping points, unambiguous orientation and location, adoption of the stacking principle for assembly, and ease of location for screws, etc. This usually results in a simplified, better and cheaper design for the product. Quality will be improved in many areas as automatic inspection techniques are adopted. Design changes can be implemented more quickly and new products introduced efficiently. Lead times can be reduced. Work in progress can be reduced. In comparison to dedicated equipment, smaller batch sizes can be handled and downtime between product changeovers is reduced.

In the aeronautic industries, the current automation solutions for the assembly of aerostructure are based on large dedicated machines, such as auto-riveters, with the majority of the individual components being manually assembled prior to the riveting process. The high capital cost and long lead times of these machines can lead to capacity bottlenecks and the requirements for significant manual riveting. The use of excessive manual riveting also has health and safety implications (Webb, 2002). The requirements for manual assembly also add a significant amount of time to the process and large area of the factory space is required for individual workstations. The use of manual handling also represents a significant health and safety risk coupled with a large probability of component damage during assembly. These factors have led to a renewed interest in the development of more flexible automation approaches (Webb, 2002).

The application of flexible robotic cells systems has so far been confined to small product specific cells often performing single process such as drilling. A number of factors have limited the widespread adoption of more flexible robot based technologies (Webb, 2005), these include the following.

- The parts are compliant and tend to distort during handling and processing. Assembly systems must be able to compensate for this;
- The parts vary in size from millimetres to metres. As a consequence, material handling systems must be flexible and reconfigurable;
- Product volumes are relatively low making significant investment in product specific cells uneconomical. Typical volumes for most assemblies are of the order of 10s per year (Webb, 2005).

The inherent compliance of many of the components and assemblies used in aerostructure manufacture means that significant distortion occurs during handling and processing. This results in a high degree of geometric and positional uncertainty and means that conventional pick and place techniques cannot be used because it is impossible to define and fix the exact geometry and position of parts within the assembly (Webb, 2005). The problem can be mitigated by use of templates and complex structures but they increase the cost and reduce flexibility by making large parts of the system product dedicated.

The problem of designing new robotic cells can be treated by Digital Manufacturing. Although several Digital Manufacturing tools, such as CAD/CAM/CNC chain, CAE simulations, PDM (Product Data Management) and even Virtual Reality tools, have already been incorporated to the Brazilian manufacture of airplanes, it does not explore in its fullness the simulation of robotic cells for the production of aircrafts. Graphical simulation is used to design and evaluate a work cell layout before it is built. The robot motion can be programmed on the simulation and downloaded to the robot controller. Simulation software includes libraries of commercially available robots and postprocessors for off-line robot programming.

In this context, the purpose of this paper is to verify the benefits of digital manufacturing for the design of robotic cells in the aeronautic industry. This paper is organized as follows. Section 2 introduces digital manufacturing, giving particular emphasis to the design and graphical simulation of manufacturing cells. Section 3 presents a case study and discusses the advantages of the proposed approach. Finally, Section 4 presents some conclusions and discusses future works.

2. GRAPHIC SIMULATION OF MANUFACTURING CELL

Simulation can be defined as the technique of building a model of a real or proposed system so that the behavior of the system may be studied. In the case of robotic work cells, graphical simulation is usually employed. It aims at visualizing and verifying the performance of a robot in a manufacturing cell, determining features such as reachability and workspace. Furthermore, in the case of robotic cells, the graphical simulation software is also used for off-line programming (Silva, 2004), (Stobart, Dailly, 1985).

Among the features that a graphical simulator must provide, there are:

- module and robot builder;
- 3D graphical task simulation;
- “Universal” inverse kinematics;
- full dynamics models;
- trajectory and task planning;
- off-line programming.

Off-line programming makes possible the visualization of the robot movements before actually programming it. In the case of a robot with six or more joints is practically impossible to image the behavior of the end-effector when some of the joints are manipulated simultaneously (Silva, 2004). The possibility of viewing the process in a graphical simulator brings a number of advantages to the robot programming operation. It is possible to verify the robot trajectory and detect collisions with other objects of the working cell. This is particularly true for the case of working cells where two or more robots are moving simultaneously with interception among their working volumes.

Other benefits brought by the use of graphical simulators are (Silva, 2004):

- Reduction in production times. With the help of simulation, it is possible to determine the time of each operation, detect bottlenecks, and seek for best trajectory solutions.
- Verification of accessibility. Graphical simulators come with libraries composed of a large range of commercial robot models. Therefore, it is possible to test the access and reach of different robots and compare solutions without having the robots or building the work cells.
- Programming reuse and flexibility. It is possible to modify and reuse programmed operations. Regular and symmetric parts can be programmed using the mirror function.

The graphical simulation of robots was born with the need of off-line programming (Orady, Osman, 1997b). With the invention of robots and their wide spread acceptance in the manufacturing floor, there was a basic need for off-line programming, similar to CNC machine tools, of the robots in order to improve their utilization and productivity. In the 1960's and through the 1970's, off-line programming languages such as AL, AUTOPASS, RAPT, VAL, etc. were used for programming the robots. In the late 1970s, virtual reality software packages appeared for the purpose of constructing manufacturing cells, planning the robot path, detection of robot collision with other equipment in the cell, and obtaining off-line programs for the robots. These software packages were based on wire frame graphics which are difficult to visualize and have no automatic collision detection capabilities. In the 1980's, with the increase of computer power and particularly the appearance of powerful workstations with powerful graphical capabilities, new virtual reality software packages appeared in the market that were based on solid model graphics and efficient intelligent algorithms. Several vendors have developed virtual reality software. SILMA developed CIMSTATION, Tecnomatix developed ROBCAD, Deneb developed IGRIP (Orady, Osman, 1997b).

Among the main features that must be considered when selecting a simulation environment, there are (Orady, Osman, 1997a):

- Libraries available for selecting robots and composing work cell models;
- Animation features provided for visualizing the process;
- Features available for defining new robots and work cell components;

- Compatibility with other CAD programs, used for creating models of the work cell components.
- Processing speed;
- Absence of errors in the generation of robot control program;
- Modules for specific applications such as painting, welding, etc.

Figure 2 illustrates the process of modeling and simulating a robotic work cell (Chan, Kwan, 2003). The main steps of this process are related to the basic components of a graphical simulator. The process begins with the modeling of the robot (1) and the component of the manufacturing cell (2). Simultaneously, the process must be decomposed in tasks and described in the appropriate language (3). These are the inputs of the pre-processor (4), which gives as output the process visualization and data about the robot movement. When necessary, this output is used for the redefinition of the work cell layout (5). Once the simulation presents satisfactory results, a post-processor generates the robot program in the appropriate language (6). This program is uploaded in the real system, and is tested and calibrated (7). If necessary, new modifications are introduced and the process is repeated in an interactive way.

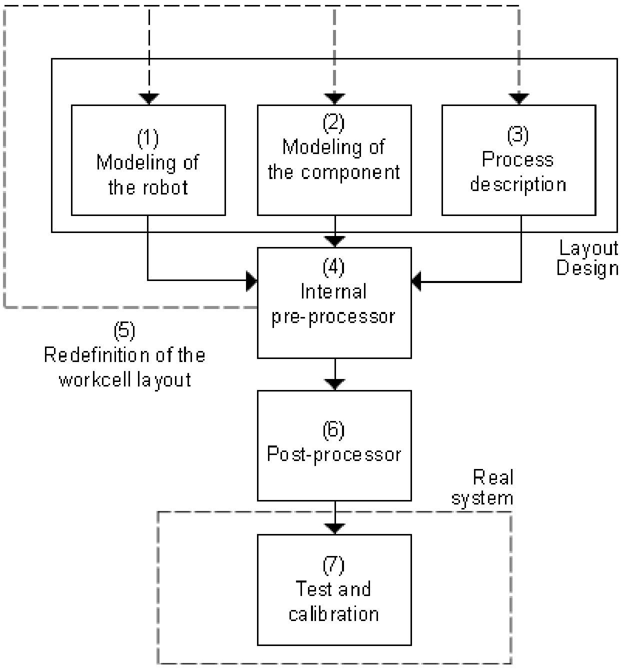

Figure 2. Graphical simulation of robots.

Currently, the number of commercial software available on the market come back to the graphical simulation and off-line programming of robots is ample. These products had started to be developed by companies that already possessed other traditional CAD products, extending to other new companies. Besides that, companies that project and commercialize robots launch their software on the market, that generally are optimized for their robots and applications in which these companies are specialized (Silva, 1996). Robcad, RobotStudio, Easy-Rob, KUKA.Sim Pro and Grasp are some examples of commercial software used and available on the market (UGS, 2006) (ABB, 2005) (Easy-Rob, 2007) (KUKA, 2007) (BYG, 2007).

The Robcad enables the design, simulation, optimization, analysis and off-line programming of multidevice robotic and automated manufacturing processes in the context of product and production resource information. It provides a concurrent engineering platform to optimize processes and calculate cycle times. With Robcad, you can design life-like, full-action mock-ups of complete manufacturing cells and systems on 3D graphics computer workstations (UGS, 2006).

RobotStudio is built on the ABB VirtualController, an exact copy of the real software that runs robots in production. It thus allows very realistic simulations to be performed, using real robot programs and configuration files identical to those used on the shop floor (ABB, 2005).

Easy-Rob is planning and simulation software for manufacturing plants with robot work cells. All processes and movement sequences for example, handling, assembly, coating and sealing with one or with multiple robots can be programmed and are instantly visualized within a 3d scenario (Easy-Rob, 2007).

Grasp10 enables the creation of accurate 3D models and real-time interactive simulations for cell layout design, planning and optimisation. As a tool for off-line programming, the instructions can be automatically translated into the required native robot language. An optional module for discrete event simulation extends Grasp10's applications areas to factory simulation, warehousing, logistics and materials handling (BYG, 2007).

KUKA.Sim Pro was developed for the offline programming of KUKA robots. The product is connected in real time to KUKA.OfficeLite, the virtual KUKA controller, thus allowing cycle time analyses and the generation of robot programs (KUKA, 2007).

Table 1 presents a comparison of features among some available products for graphical simulation and programming off-line of robots.

Table 1. Comparison among some available products for graphical simulation and programming off-line of robots (UGS, 2006) (ABB, 2005) (Easy-Rob, 2007) (KUKA, 2007) (BYG, 2007).

	Robcad	**RobotStudio**	**Easy-Rob**	**KUKA.Sim**	**Grasp10**
Supplier	UGS Unigraphics Solutions	ABB Automation Technology Products	EASY-ROB™ Simulation System	KUKA Robotics	BYG Systems Ltd.
Current version	7.5.1	5	4.305	2.0	10
Supported systems	Windows 2000 and XP PCs, SGI Unix workstations, HP Unix workstations and Sun Unix workstations.	---	Microsoft Windows with the OpenGL graphic library.	Microsoft Windows 2000 and Windows XP.	Standard PC with Windows 2000 or XP.
Features	– Integrate with most industry MCAD systems, including native data from Catia, NX™ software Pro/Engineer, I-deas® NX Series software, CADDS5, direct CAD interfaces or neutral formats such as JT, IGES, DXF, VDAFS, SET, STL and STEP; – Robots, machines, tools, equipment libraries; – Modeling of components; – Modeling of complex kinematics of robots and other mechanisms; – Robotic and assembly path definition with reachability check, collisions detection and optimized cycle time; – Motion simulation and synchronization of several robots and mechanisms; – Off-line programming; – Optimized programs downloaded to robots on the shop floor; – Uploading existing production programs for optimization.	– Support data in major CAD formats including IGES, STEP, VRML, VDAFS, ACIS and CATIA; – Contain autoPath. It is possible to automatically generate the robot positions needed to follow the curve; – Automatically analyses reachability; – Automatically detect and warn about programs that include motions in close vicinity to singularities, so that measures can be taken to avoid such conditions; – Collision detection; – Robot program can be downloaded to the real system without translation; – It is possible to run several virtual robots at the same time; – Contain libraries: models of ABB robots.	– All processes and movement sequences for example, handling, assembly, coating and sealing with one or with multiple robots (Single-/Multi Robot Version) can be programmed; – Check reachability, collisions and travel ranges; – Analyze and estimate cycle-time; – Offline programming; – Custom product development (API - Application Program Interface); – Integration of custom mathematical methods and solutions (API); – Tool for training and education.	– Components are parametric in design; – Allow to load CAD data from other systems or build models using CAD tools; – Allow to build grippers, welding guns and other kinematical structures; – Contain libraries: models of KUKA robots, additional models of grippers, pallets, conveyors, fences, forklifts and several simulations; – Models available in the Internet; – Detect collisions and near misses; Simulate grippers, servo welding guns and other kinematical structures; – Offline modifications possible without interrupting production; – Cycle time analysis and optimization, program structuring.	– Windows style interface and menu system with icon toolbars; – High level interactive 3D graphics; – In built interactive 3D solid modeler; – Robot reach validation and configuration checks; – Cycle time analysis; – Dynamic collision and near miss detection; – General kinematic modeller with a library of industrial robots; – Support for multiple robots and synchronised external axes; – Macro driven real-time interactive simulation playback, for results presentation and demonstrations; – Output of simulation models in AVI, VRML, Runtime and Grasp10 Preview formats; – Off-line programming with support for most native robot languages. Grasp10 uses accurate robot and work-cell calibration techniques with no external measuring equipment required.

3. CASE STUDY

The need of creating more flexible cells forces the incorporation of robots in the production line of aeronautic industries. In this context, the use of graphical simulator is essential for the conception and design of new cells. It is a powerful tool for verification and validation, helping on the reachability analyses, equipment selection and determination of process times (Dooner, 1983).

This section presents as a case-study the reachability analysis of an industrial robot for orbital riveting of aircraft fuselage. The fuselage is composed of large and heavy sections that cannot be rotated in order to be riveted. Currently this operation is performed manually and faces a number of problems that justify its automation (TECMES, 2001).

Among them is the long time demanded, the need of lining the fuselage sections, etc. The orbital riveting is currently used in the aeronautic industry and has proved to be a good solution in quality and speed when comparing to other technologies (ICE, 2005).

This case study has originally been presented at (Aguiar et al, 2007). In this paper we start from the results obtained in (Aguiar et al, 2007) and propose new configurations for the cell, in order to achieve the necessary flexibility.

3.1 Problem Description

The problem considered in this paper is the junction of two fuselage tubes. A number of rivets must be performed along the circumference of the fuselage junction. The software used for graphical simulation is the ROBCAD, version 7.5.1, from UGS – Unigraphics Solutions. The choice of the simulation software is based on the availability of licenses in the CCM laboratory, at ITA. The fuselage has a circular section with an external diameter of 2280 mm, according to Figure 3.

3.2. Robot Selection

The robot selected for the operation has six revolution joints, a reach of 3500mm, a reasonable work volume (as illustrated in Figure 4) and a payload of 150 kg. These features make it appropriate to perform the orbital riveting in the aircraft fuselage.

The manufacturing cell is provided with a track of 7700 mm. The robot is accomplished to the track and gains another degree of freedom, which gives it a large access range in the fuselage surface. As a result the system (robot+track) has six revolution joints and one prismatic joint: four joints are used for positioning and three for orientation. According to the study presented in (Aguiar et al, 2007), the best position for the track may vary from 1500 to 2000 mm of the fuselage.

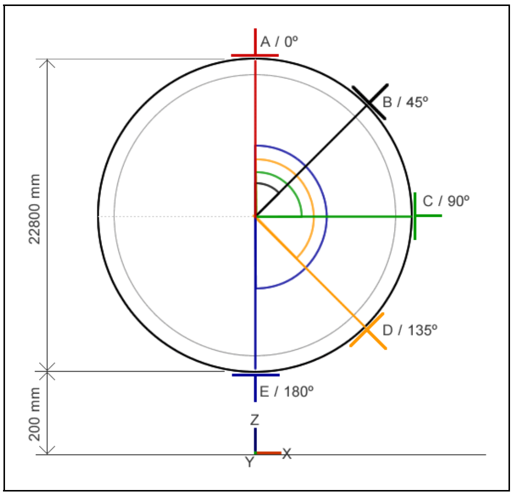

Figure 3. Fuselage section and points of access.

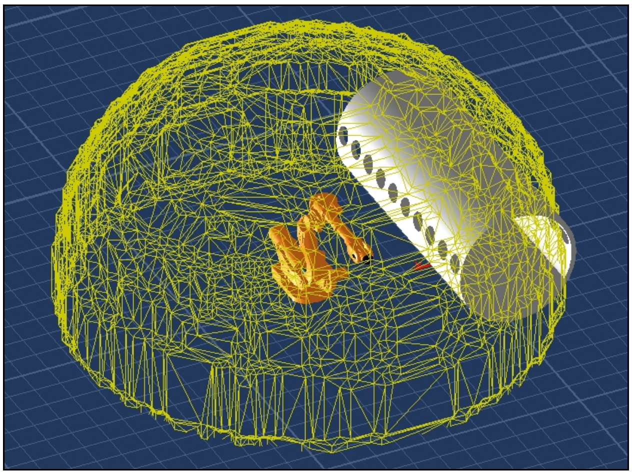

Figure 4. Robot work volume.

3.3. Specification of Solutions

The study presented in (SBAI, 2007) concluded that the system specified in that paper does not have the flexibility necessary to reach all the access points of the fuselage – particularly, point E is not reached. In order to seek for a satisfactory configuration for the work cell, the solution proposed here is to vary the height Z from the floor to the fuselage (Figure 5). The distance X has been fixed at 1500 and 2000 mm.

The purpose of the simulation study is to orientate the TCP frame of the robot in such a way that its Z axis intercepts the fuselage surface and its X axis is parallel to the Y axis of the world reference system, according to Figure 6. Moreover, the TCP frame must maintain a distance of about 300 mm from the fuselage surface. This positioning assures that the riveting operation will be performed in a correct way (the riveting process must be performed perpendicular to the fuselage surface).

The major difficulty of the robot positioning refers to the lower part of the fuselage. For this purpose the proposed approach is to vary the fuselage height Z. The following values are considered: 200, 400, 600, 800, 1000 and 1600 mm. These tests are performed for both X=1500 and X=2000 mm.

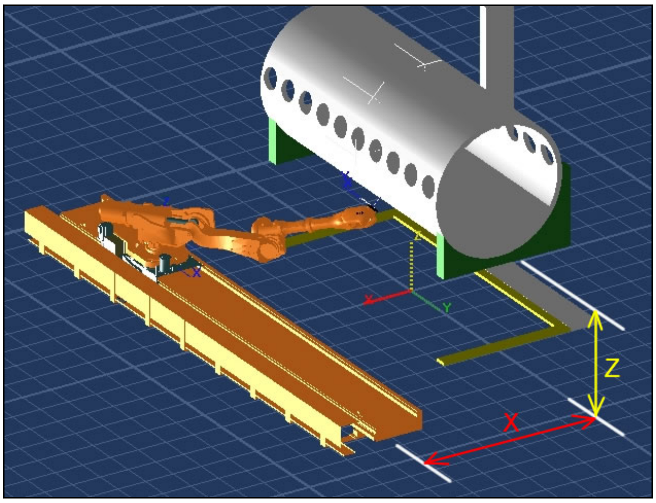

Figure 5. Fuselage positioning.

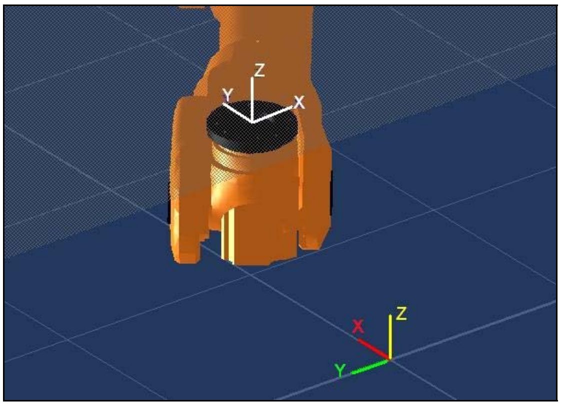

Figure 6. TCP frame, world frame and fuselage surface.

3.4. Simulation Results

The simulation results are presented in Tab. 2, for X=1500 mm, and Tab. 3 for X=2000 mm. The results pointed out that an increasing value of Z does not assure the access to the lower zone of the fuselage and may deteriorate the access to the upper points.

For X=1500 mm and Z varying from 200 to 600 mm, the access possibilities are not changed. From 800 mm, some points become unreachable for the robot. For X=2000 mm, for Z great than 800 mm, the access to point E is possible, but point A becomes unreachable.

Table 2. Results of accessibility tests for X = 1500 mm.

Z (mm)	Access A	Acess B	Access C	Access D	Access E
200	YES Point inside axis limits. No risk of collision.	YES Point inside axis limits. No risk of collision.	YES Point inside axis limits.	YES Risk of collision between the robot and fuselage.	NO Point outside the robot axis limits
400	YES Point inside axis limits. No risk of collision.	YES Point inside axis limits.	YES Point inside axis limits.	YES Risk of collision between the robot and fuselage.	NO Point outside the robot axis limits
600	YES Point inside axis limits. No risk of collision.	YES Point inside axis limits.	YES Point inside axis limits.	YES Risk of collision between the robot and fuselage.	NO Point outside the robot axis limits.
800	NO Point outside the robot axis limits	YES Point inside axis limits.	YES Point inside axis limits.	YES Risk of collision between the robot and fuselage.	NO Point outside the robot workspace.
1000	NO Point outside the robot axis limits	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	NO Point outside the robot axis limits
1600	NO Point outside the robot axis limits	YES Point inside axis limits.	NO No robot trajectory available.	YES Point inside axis limits.	NO Point outside the robot axis limits

Table 3. Results of accessibility tests for X = 2000 mm.

Z (mm)	Access A	Access B	Access C	Access D	Access E
200	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	NO Point outside the robot axis limits.
400	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	NO Point outside the robot axis limits.
600	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	NO Point outside the robot axis limits.
800	NO No robot trajectory available.	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.	YES Point inside axis limits.
1000	NO No robot trajectory available.	YES Point inside axis limits.	NO No robot trajectory available.	YES Point inside axis limits.	YES Point inside axis limits.
1600	NO No robot trajectory available.	YES Point inside axis limits.	NO No robot trajectory available.	YES Point inside axis limits.	YES Point inside axis limits.

3.5 Analysis of the Results

The results obtained in this study shows that the system in the proposed configuration is not able to access all the specified points of the aircraft fuselage surface. The main problems faced are the robot axis limits, the risk of collision between the robot, the track and the fuselage, and no trajectory available (no inverse for the robot cinematic).

However, this study guides the specification of another configuration that must satisfy all the requirements. In this new configuration, the system will be provided with a flexible Z that may vary from 600 to 800 mm.

4. CONCLUSION

This work highlights the benefits incorporated to the aircraft manufacturing system design by robot graphical simulators. It approaches the problem of automating the orbital riveting of aircraft fuselage sections. Complex access points are easily verified and different configurations are tested without the need of having the physical devices.

Among the future works are the verification of access points for other fuselage diameters (corresponding to other aircrafts) and other industrial robots, and the use of two or more coordinated robots in the riveting operation.

5. ACKNOWLEDGEMENTS

This work received financial support of the governmental agencies FAPESP, CNPq and CAPES. Particularly, the authors would like to thank the UGS for providing the ROBCAD software.

6. REFERENCES

ABB, 2005, “RobotStudio 5 Data Sheet”. 15 Apr. 2007, <http://library.abb.com>

Aguiar, A.J.C., Silva, A.S.A., Villani, E., 2007, “Simulação Gráfica de Robôs Aplicada à Indústria Aeronáutica”. VIII SBAI – Simpósio Brasileiro de Automação Inteligente, Florianópolis, Santa Catarina, Brasil (under review).

BYG, 2007), “BYG & Robotics”. 4 May 2007, <http://www.bygsimulation.com>.

Chan, S.F.; Kwan, R., 2003, “Post-processing methodologies for off-line robot programming within computer integrated manufacture”, Journal of Materials Processing Technology, 139(1-3): 8-14.

Day, M., 2002, "What is PLM?. CAD Digest", 5 Apr. 2007, <http://www.caddigest.com>.

Dooner, M., 1983, “Computer simulation to aid robot selection”, IEE Control Engineering - Robotic Technology, Series 23, Chapter 10, Stevenage, England, pp. 103-111.

Easy-Rob, 2007, “Product”. 4 May 2007, <http://www.easy-rob.com/en>.

Evans, M., 2004, "The PLM debate". 5 Apr. 2007, <http://www.cambashi.com/research/plm_debate/plm_scm.htm>.

ICE - International Cooperative Education, 2005, “Civil Assembly Fuselage”. 3 Apr. 2007, <http://www.icemenlo.com/EADS/CD/PDF>.

KUKA, 2007, “KUKA Software”. 4 May 2007, <http://www.kuka.com/usa/en/products/software>.

Orady, E.A.; Osman, T.A., 1997a, “Capability Study of Robotics and Manufacturing Cell Simulation Software”. Computers and Industrial Engineering, 33(1-2): 83-86.

Orady, E.A.; Osman, T.A., 1997b, “Virtual Reality Software for Robotics and Manufacturing Cell Simulation”. Computers and Industrial Engineering, 33(1-2): 87-90.

Silva, M.S., “Simulação e Programação Off-line de Robôs de Montagem”. 1996. 218 f. Dissertação (Mestrado) - Faculdade de Engenharia da Universidade do Porto, Porto.

Silva, M.S., 2004, “Aplicação da Simulação à Programação Off-Line de Robôs Industriais”, Ingenium, No. 81, pp. 72-77.

Stobart, R.K.; Dailly, C., 1985, “The use of simulation in the off-line programming of robots”. IEE Control Engineering - Robots and automated manufacture, Series 28, Stevenage, United Kingdom, pp. 11-28.

TECMES, 2001, “Sistema projeta avião em três dimensões”. 1 Apr. 2007, <http://www.tecmes.com.br/home/News_Embraer_24.html>.

UGS, 2006, “Robcad - Digital manufacturing environment for robotic work cells verification and off-line programming”. 15 Apr. 2007, <http://www.ugs.com/products/tecnomatix/docs>.

Webb, P., 2002, "The application of mechatronic systems within aero-structure assembly - Drivers and Barriers", Proc. Mechatronics, Twente, pp. 1126-33.

Webb, P.; Eastwood, S.; Jayaweera, N.; Chen, Y., 2005, “Automated aerostructure assembly”. The Industrial Robot, 32(5): 383-387.

7. RESPONSIBILITY NOTICE

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