Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 38 (2019) 808–815

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

29th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2019), June 24-28, 2019, Limerick, Ireland.

Automated Planning and Optimization of a Draping Processes Within the CATIA Environment Using a Python Software Tool

M. Körber*, C. Frommel

German Aerospace Center, Center for Lightweight Production Technologies, Augsburg 86159, Germany

Abstract

This thesis deals with the challenge of the virtual generation and optimization of complex production processes within the Computer Aided Design (CAD) environment CATIA V5. A flexible robot end-effector –(Modular Gripper) is used for automated draping process which serves as an example application. Previous experiments have shown that manually selecting all process parameters that are needed to adapt the grippers geometry is a very time consuming and inaccurate method. For this reason, a tool was developed that is able to determine or optimize the end-effector specific process parameters offline. Since all existing models are available in the CATIA environment, the goal was to perform this optimization process within the same environment. The software tool is based on a hierarchical architecture where all calculations and logic optimization processes are processed in the programming language Python. All commands in the CATIA environment are performed with the help of small, modularized CATScript programs that are able to interact with main Python algorithms. This work is intended to lead to a virtual optimization process that can be universally applied to complex manufacturing processes. Since it is based on a Python framework that is able to interact directly with the CATIA environment, it is flexible enough so that it can be adapted to further processes.

Keywords: autonomous manufacturing; automated preforming; virtual planning; optimization; digital twin; offline programming

* Corresponding author. Tel.: +49 821 319874 1051
E-mail address: marian.koerber@dlr.de

10.1016/j.promfg.2020.01.113

1. Introduction

Due to high lightweight construction potential the amount of carbon fibre reinforced plastics (CFRP) parts in modern airplanes and space structures is rising. The negative side of this development is the still cost-intensive production of these components. The high financial effort results on the one hand from high material costs, but also from the fact that the majority of these components are still manufactured manually. For this reason the Center for Lightweight Production Technologies (ZLP) of the German Aerospace Center (DLR) in Augsburg is developing automation solutions for manufacturing processes of large-area fiber composite components. The aim is to create reliable and accurately processes to lower the production costs and time.

One of the main challenges the team at ZLP is facing is the automation of the dry carbon fibre preform process, for example for rear pressure bulk head (RPBH). The challenging characteristic of this component is its double-curved surface. A preform consists of several layers of dry fibre cut-pieces, which are stacked on top of each other in the tooling, taking into account their considered fibre orientation. The tooling of the RPBH has an oval geometry with diameter between 3.8m to 4.2m. This means that the curvatures of the tooling surface are not constant through the geometry. An automated process therefore requires end-effectors that can adapt flexibly to the different curvatures in the tooling. Figure Fig. 1 shows the RPBH tooling during the preform process. More detailed information about the draping process can be found in [1] [2].

Fig. 1. Rear Pressure Bulk Head tooling in the robot cell

The automated preform process can be represented as a pick and place procedure and is implemented by means of an industrial robot equipped with the modular gripper. First, a flat fiber cut-piece is picked up from a table. Since the surface of the RPBH tooling is double curved, the cut-piece must also be transferred into target geometry. It is then placed in the tooling at the intended position. The main challenge during deformation is the draping of the cut-piece. This topic is part of several ongoing and already completed works at ZLP.

1.1. Draping process with the Modular Gripper

The Modular Gripper is an end-effector system developed by the company Schmalz together with the ZLP Augsburg during the LUFO project AZIMUT. The main property of the Modular Gripper is the ability to deform its suction-capable surface from a flat state to a double curved geometry. To fulfil this task the end-effector is designed on the base of a spine-rip-architecture with 18 actuators in total. The spine consists of three linear actuators that are connected to the end-effectors structure and two glass fibre rods leading through the entire structure. With the movement of the actuators, the glass fibre rods are being pulled and bend as a result. The suction surface is connected to these rods and thereby adopts the same curvature. Each of the remaining actuators are mounted on one of the rips. The rips consist of suction units connected in series. The actuator drives a shaft, which changes the angle of the pitch between the units uniformly. This way the end-effector is able to arch each of the 15 rips independently. Together the two single curvatures of spine and rips create a double curved surface whose geometry depends on 18 parameters. The suction surface consists out of 127 suction units that are individually controllable in its suction intensity. Figure Fig. 2 shows the deformed end-effector with a draped textile attached to the suction surface. [5][4]

Fig. 2. The Modular Gripper with textile after deformation

The challenge during the described preform process is to find the specific set of parameters that allow the end-effector to adapt its suction surface geometry to the tooling geometry at a defined position. As described above, the curvatures in the tooling vary greatly, which means that the parameters must be selected or calculated anew for each cut-piece position. In previous experiments, the position of the end-effector and the deformation parameters were manually adjusted directly at the experiment setup using the try-and-error method. Subsequently, the cut-piece was picked up from a flat table, deformed and deposited at the approximate target position. The problem at this process step was to link the gripping position and rotation of the cut-piece on the suction surface with its designated position inside the mould. Validation of the placement result has so far only been possible with laser projections after placement. For this reason, the recording position and rotation were manually adjusted until the position of the cut-piece matched the laser projection. [1]

The virtual models of the tooling, the three-dimensional cut-pieces inside the tooling and the Modular Gripper are available as CATIA files. The question to be worked out in this thesis is whether the calculation of the process parameter and the planning of the preform process can be realized inside the CATIA environment. The goal is the development of a software tool which is able to simulate the deformation of the end-effectors suction surface and its interaction with the virtual RPBH tooling. For this purpose, the end-effector should be able to be set virtually to a predefined position in the tooling. Then the parameterized actuators should interact step by step to their optimal position by constantly evaluating the distance between the tooling surface and the suction surfaces. [3]

2. Method principal

The idea behind the software tool mentioned in chapter 1 is to develop an automated process that determines the optimal end-effector configuration. The optimal process configuration is considered to be a set of parameter that creates a suction surface geometry that is as close as possible to the surface of the RPBH tooling. This also includes the determination of the position and rotation of the end-effector in Cartesian space relative to the robot cells coordinate system. For this purpose, a method has to be developed that allows step-by-step approximation of the optimal configuration with the help of the simulation environment and an iterative method. We have chosen CATIA because it is widely used within the aerospace industry as construction tool. In addition to the classic design functions, this software offers the option to automate some of its functionalities using the Visual Basic for Applications (VBA) or CATScript scripting language. Those small scripts are also referred to as macros. With the help of such a macro, the user is enabled to write commands that can, for example, change the position of a component within a product translatorically as well as rotatorically. In the same way, the properties of components can be adapted. These properties include construction elements, predefined parameters or the color. Within this work, this functionality offers the possibility to develop algorithms with which the desired optimization process can be realized. [6]

2.1. CAD model of the Modular Gripper

The software tool is based on existing CAD models in CATIA format. The models consist of two main components. Firstly, the tool mold of the RPBH component, which is firmly anchored in the world coordinate system of the robot cell. On the other hand the model of the Modular Gripper. This model is to be implemented as a free-moving and rotating object. It consists of 254 complex individual components. This density of detail leads to a high calculation effort even with smaller geometrical changes. Since the aspired procedure has to carry out a multitude of calculations with its try-and-error method, it is advantageous to reduce the detail density of the model. Therefore, all components except the adaptable wire-frame model were removed from the original model. The remaining virtual bodies in the simplified model are the suction surfaces, whose geometric complexity has also been greatly reduced. This way, the software is saving computing load, which has to be mainly invested in the visual rendering of the complex model. Figure Fig. 3 show on the left side the complex end-effector model, on the right side the simplified model with reduced level of detail.

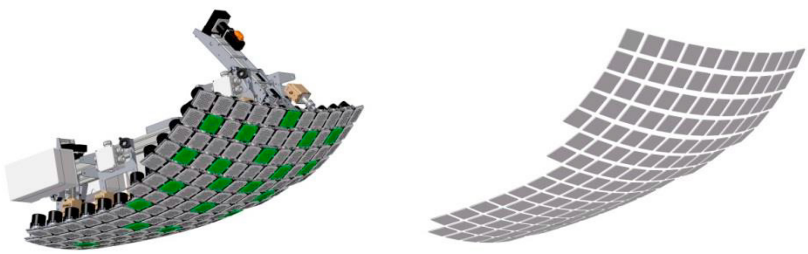

Fig. 3. Left: Full CAD model of the end-effector; Right: Model with reduced details

The main function of the end-effector model is to reproduce the geometry of the suction surface depending on the given motor values. The conversion of the rib deformations is relatively simple, since the angle changes between the suction units of a rib behave linearly to its motor value. The calculation of the spine curvature is more complicated. Here the curvature is specified by the elastic properties of the two elastic glass fibre rods. These are connected to the end-effector structure at four points. One support point is firmly connected to the structure, while the other three are connected to the mentioned linear actuators. By pulling on the rods, the position of the support point can be changed in three dimensions, resulting in a deformation of the rods. This mechanism is reproduced in the mechanized CAD model using a wire-frame model (Fig. Fig. 4).

In the model the actuators are implemented as red protruding lines, which length can be varied with the help of linked parameters. These actuator lines are connected to a spline element that symbolizes the glass fiber rods. By changing the actuator line length, the spline adapts to the new boundary conditions by adjusting its curvature. The wire-frame designs of the suction units are connected to the spline. This model concept generates a kinematic chain that can virtually reproduce any deformation of the real end-effector based on the real motor values.

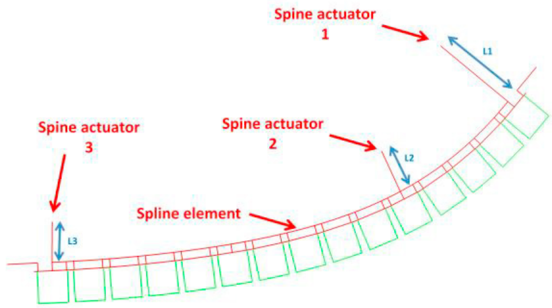

Fig. 4. Wire-frame model of the end-effectors spine

2.2. Software architecture

As described above, the CATIA user is able to make changes to the CATIA model using a VBA macro. However, the tool developed for the optimization process goes beyond the simple modification of models. It is supposed to offer a modular platform to control different virtual processes as well as to perform calculations and evaluations. Figure Fig. 5 shows the structure of the architecture and the flow of data.

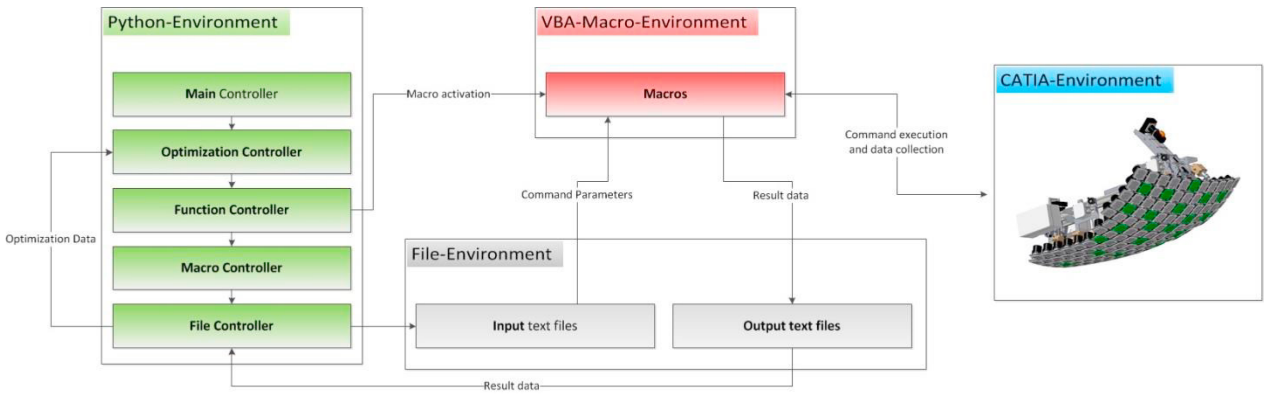

Fig. 5. Software architecture

For this purpose a software architecture was developed, which is divided into three environments. The first environment allowes the possibility to write the process control, the optimization logic and the data management in the programming language Python. Python offers a large number of libraries, debug functions and a large community to support challenging problems. It is a language with which large amounts of data can be managed and processed. The Python environment is based on five hierarchically structured modules. Each module has a predefined function and is only able to access the module at the next lower level. This way the concept ensures modularity that allows this software tool to be used as a framework for further applications.

The main controller manages all parameters required for the process. These include the initial motor values, abort criteria and optimization step sizes. These parameters are used to initialize the class in the optimization controller. The optimization controller is responsible for controlling the process logic of the optimization and reacting to incoming process data. It uses different function classes in the function controller. These function classes provide the tool to perform different actions with the CATIA models. These include translational and rotational movements of individual model components, deformation of the end-effector model, output of distances between two bodies etc. The individual functions are able to control the individual macros available with the help of the Macro Controller. This includes the execution of the macros, the run-time management as well as the handling of the input and output data of the macros. To accomplish this, the File controller offers libraries for writing and reading text files in which input and output data are transferred. In order to control CATIA V5 using external scripts, the VBA programming language must be used. Since this language has a much smaller feature set and debugging capabilities, it is attempted to keep the realized functionality in VBA macros as small as possible. Only the features necessary to control the models are to be programmed in VBA. All other functionalities were implemented in Python. The goal was to write these scripts on the basis of a template. Thereby required initializations and modules are pre-given. The individual source text is inserted at a predefined location. With the help of the template modules command parameters can be imported and process results stored in an output file. The File Environment offers the possibility to exchange information between the VBA based macros and the Python tool. There are two types of files. The input files that pass the command parameters to the macros. These include motor values for virtual deformation or the selection of suction units whose distance to the tooling surface is to be determined. The output files return data collected in CATIA, such as distances and current configurations, to Python. The files also serve to determine whether a script has been completed and the next optimization step can be initiated. [6]

3. Offline generation and optimization of the process parameter

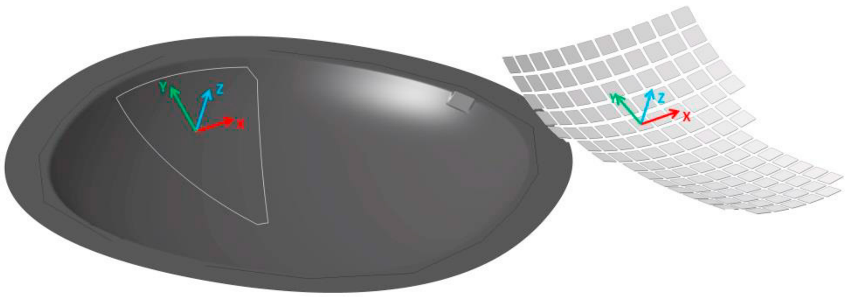

Fig. 6. Axis alignment at the end-effector TCP and the on the tooling surface

The process starts with the manual positioning of the target Tool Center Point (TCP) on the virtual surface of the RPBH mold. The counterpart of this point is located in the center of one suction unit on the end-effector suction surface. With the help of these two points a vector, that describes the translational displacement, can be calculated. With the help of a coordinate transformation the end-effector model is moved exactly to the given position.

To move the gripper into the initial orientation a coordinate cross is generated automatically in the TCP of the tooling, whose Z-axis is aligned orthogonally to the surface and the Y-axis is aligned in the direction of the RPBH molds center. The end-effectors alignment are based on this coordinate system. This way the initial orientation of the model is ensured. During the optimization process, this orientation will be continuously updated by means of a rotation in the X-axis and the parallelism to the tooling surface. Figure Fig. 6 shows the simplified end-effector model and the corresponding axis orientations. The optimization fitness values are mainly based on the distance measurements of each suction unit to the tooling surface. These calculations are triggered by macros and calculated within CATIA. However, CATIA always provides positive distances, regardless of whether the suction unit is positioned in front or the back of the tooling surface or if it penetrates the surface. This would lead to an uncontrollable optimization process if multiple suction units pass through the virtual tooling. For this reason, a distance measurement that detects a surface penetration must be introduced and is able to produce positive as well as negative distances.

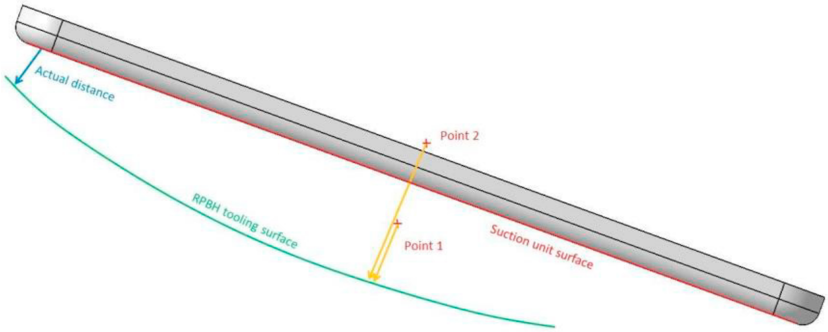

Fig. 7. Representation of the sign calculation of the distance between suction unit surface and tooling surface

To solve this problem, two new points were added to each suction unit. Both points were positioned 5 mm orthogonally from the center of the surface, but in different directions. Thus, each unit has one point on its front and one on its back. The macro has been modified to request a distance three times. The first measurement between the suction unit and the tooling surface is calculated, which is always positive or zero. The sign is determined using two further distance measurements. Depending on which of the two points the tooling is closer to, the distance is given the corresponding sign. In order to provide visual feedback, the suction units are colored according to the distance indicated. Units with a positive distance to the surface are colored green. The units that get in contact with the tooling and thus have a distance of 0 turn yellow. Those with a negative distance are colored red. In this way, the user gets a quick overview of the result and can identify possible areas with greater pressure forces. Figure Fig. 7 shows a configuration where the suction surface is located in front of the tooling surface. The fact that point 1 is closer to the surface than point 2 gives the actual distance a positive sign. Figure Fig. 8 shows an end-effector model whose suction units have been color-matched to the distances to the tooling surface.

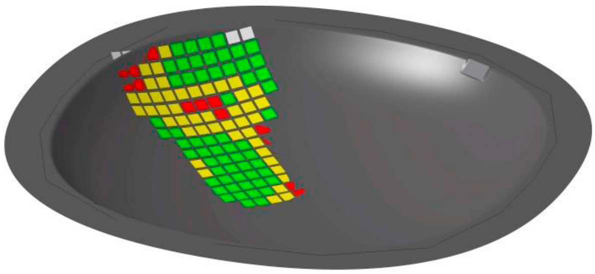

Fig. 8. End-effector during optimization process with with coloured suction surfaces

Before starting the optimization process, the geometry of the end-effector is deformed to maximum curvature, so that the deformation can be slowly approximate from the most extreme configuration to the optimal configuration. The process is divided into two phases. In the first phase, the spine section is adapted to the tooling geometry. The three linear actuators are addressed in a given sequence and reduce the curvature of the glass fiber rods step by step. The value of the linear axis displacement is determined by the distance between the connected suction unit and the tooling. This is repeated until a corresponding stop criterion is met.

In the second phase of the optimization process, the 15 ribs are formed one after the other. The process starts with the top rib. Here, too, the size of the deformation step is determined on the basis of the distances between the suction units and the tooling. This is carried out until a termination criteria, based on the distances between gripper and tooling surfaces , is fulfilled.

After completion of the optimization process, a virtual end-effector is available whose actuator values and position as well as rotation in the robot cell coordinate system can be retrieved. These parameters are transferred to the robot program of the real process.

4. Result

This optimization process was applied to 23 cut-pieces with different geometries and positions in the RPBH mold and thus 23 parameter sets were generated. Using the virtual simulation environment, the cut-piece position and rotation can be defined on the curved suction surface. In order to calculate the cut-piece position and rotation on the flat suction surface, a so-called flattening must be carried out with the aid of a draping simulation. Here the draping process is recalculated to obtain the cut-piece condition before it was draped. This data defines the gripping configurations on the basis of which the end-effector positions itself correctly on the cut-piece during the take up.

The tests carried out have shown that the placement results were very promising. The position of the deposited cut-pieces was compared to the laser projection of target positions. As described in chapter 1, the RPBH mold has no constant curvature and varies greatly. The tests showed that the simulation results led to a satisfying placement quality, especially in the area with low curvature variation. In the area of strong curvature variation, the suction units on one rib side were still a few centimeters away from the surface, while the other rib side was pressed against the tooling surface and elastic deformation occurred. That caused the cut-piece to fall down on the moulds surface. As a result, it could no longer be guaranteed that the boundary curve of the cut-piece would be at the intended position. However, the areas that had direct contact with the surface showed promising deposit quality.

The original optimization method based on evolutionary algorithms could not be applied because the calculation times in CATIA turned out to be too long. The calculation of a complete optimization process for one cut-piece takes 50 minutes with the available workstation (Intel i7, 3.3GHz, 16GB RAM). The most time-consuming calculations are the determination of the distances between the suction unit and the tooling surface, as well as the deformation of the end-effector.

5. Conclusion and future work

The results of the presented process optimization procedure have shown that a complex process in the CATIA environment can be planned and optimized with the help of a software tool based on VBA macros and Python. The existing models do not have to be transferred into a separate development environment. Using a system for data exchange between Python and CATIA, the commands and calculations are reliably transferred.

The outcomes in areas of strong curvature variations should be improved. For this purpose, a separate optimization step should be introduced in which the end-effector model rotates around the Y-axis and is validated whether the rotations carried out generate a better fitness value. This optimization can also be performed iteratively over several passes.

An opportunity for improvement represents the calculation time. This problem is based on internal calculations in CATIA that cannot be accessed directly. One solution would be to reduce the use of calculation-intensive processes.

References

[1] H. Voggenreiter and D. Nieberl, eds. AZIMUT Abschlussbericht. Augsburg: TIB, 2015

[2] T. Gerngross. “Projekt AZIMUT - Automatisierungslösungen für große Faserverbundbauteile in der Luftfahrt.” In: Ingenieurspiegel (2015), pp. 1870 –1879

[3] C. Frommel et al. Versuchsbericht PROTEC-NSR Preforming mit Modulargreifer und QS mit FWM, TScan und Wirkflächensensoren. Tech. rep. Augsburg: German Aerospace Center (DLR), 2017

[4] M. Körber and C. Frommel, eds. Sensor-Supported Gripper Surfaces for Optical Monitoring of Draping Processes. Stuttgart: SAMPE, 2017

[5] K. Gupta et al., eds. Vision Based Deformation and Wrinkle Detection for Semi Finished Fiber Products on Curved Surfaces. IEEE International Conference on Automation Science and Engineering (IEEE CASE 2018), 2018

[6] Dassault Systèmes, CATIA documentation (2013).