Computer-AidedDesign

Taylor & Francis
Taylor & Francis Group

Tool path generation on doubly-curved free-form surfaces

J. Halbritter [ID], Ramy Harik [ID], A. Zuloaga [ID] and M. vanTooren [ID]

University of South Carolina, USA

ABSTRACT

In the industries of automobiles and aerospace, the use of carbon fiber has grown exponentially over the years due to its high specific mechanical properties. However, the existing processes for the placement of these fibers within these industries remain either restrictive to the doubly-curved free-formed surfaces required or economically unfeasible in respect to the larger scaled components desired. This paper presents a tool path generation process which will produce non-collisional pathing for the automated placement of uniform fibers on scalable doubly-curved free-formed surfaces. To begin this process, a high detail model of the component part is given and analyzed for optimizing the placement of both the part and the KUKA robot. We account for variable fiber placement angles based on design specifications of the part. The process will begin with a given three-dimensional rendering of the component model which will be transformed into a readable Robotic Scripting Language (RSL) file that the machine will be able to understand. This research is intended for the eventual development of manufacturing support software for automated fiber placement (AFP).

1. Introduction

The term fiber placement refers to composite fabrication processes that involve the laying down of reinforcing fibers along predefined trajectories onto a component [7]. A number of different automated processes have elements that could be considered fiber placement, such as filament winding, pultrusion, and preforming [5]. However, these are not considered in this study due to their limited range in adaptability to double-curved free-formed component surfaces. Manual lay-up processes, which are done by hand, allow for the widest range in double-curved free-formed component surface application by comparison. However, this type of manual lay-up process comes with the higher cost of being labor intensive, slow, and thus economically unfeasible for larger scaled component applications. For this study, larger scaled component from the industries of automotive and aerospace are the focus, where large doubly-curved free-formed component parts are expected to undergo a complete fiber placement process individually based on ground objects. Ground objects are then later removed in the process. This is where the potential automation of a manual lay-up process, referred to as an automated lay-up process here forth, should be able to adapt double-curved free-formed component

surface applications without the cost of intensive labor, while improving both accuracy, precision and efficiency [7]. This tool path generation process will then be able to enable the economic feasibility of automated lay-ups for larger double-curved free-formed component surfaces.

This tool path generated placement of automated lay-ups requires design support methodology that are currently not offered by any commercially available design software and are vital to the automated lay-up process development. This paper will focus on the use of highly adaptable AFP processes which will leverage an in-house tool path generation process specifically intended for doubly-curved free-form component surfaces. AFP machine development continues to improve in both accuracy and precision [1] and currently boasts over 7 degrees of freedom, 6 rotational and 1 translational for this study. In adapting this tool path generation process to an AFP manufacturing option, several key factors must be accounted for in both its planning and designing. This is done to avoid challenges found in tool pathing, for example, additive manufacturing challenges associated with running tow fibers along a curved free-form surfaces which may lead to gappages and overlaps [1]. In regards to planning, computational and simulation needs will

need to be assessed before designing the experimental plan. It is through the experimental plan, not covered in this paper, that an AFP's application feasibility for this purpose can later be determined. This paper will first explore the findings and analysis of various workcell setups. A workcell being defined as the arrangement of resources in a manufacturing environment to improve the quality, speed and cost of the fiber laying process which is designed to improve the process flow and eliminate waste throughout the automated lay-up process. Workcell designs in this study will focus on both the initial placement and movement boundaries of both the robot and the component part. Several demonstration workcells were built with the perspective of a future robotics platform that would provide extended degrees of freedom to generate placement paths and avoid workcell collisions. To do this, we present a Java slicer developed specifically for this end that outputs RSL code, needed for the functionality of the demonstration workcell. In regards to designing a tool path generation process for AFP, it becomes critical to define both the abilities and limitations of a lay-up process. Through the use of an AFP system, designers are provided with increasing capabilities for non-conventional composite manufacturing. Non-conventional, in this study, will be primarily concerned with fiber steering which may fall outside the standard 0/45/90 degrees within each ply. However, manufacturing limitations on fiber tow steering design will not be covered in this paper. For more information on fiber steering limitations refer to Lopes [6]. The following sections summarize the parameters used in the various pre-set styles. Figure 1 is a chart representing the overall manufacturability study being conducted for our client, from which this paper will only cover the first three phases for this study.

2. Implementation

The following is the methodology used to make the carbon fiber placement machine follow a path about a doubly-curved free-form surface of a given component which is uniquely generated based on the defined user input. The process will begin with a given three-dimensional rendering of the component model which will be transformed into a readable RSL file that the machine will be able to understand. The first step is accomplished by first bringing in a high detailed rendering of the component model intended for AFP which was created using Catia, an industry leading computer aided engineering program for three-dimensional rendering. Catia stores this component model information in the form of an STL file format which contains several coordinate triangles which together make up the complete

rendering of the model. Each coordinate triangle contains its normal direction and three separate point vector which make up the surface triangle. By utilizing the STL's systematic rendering process of the component, we then import the STL file into our in-house Java based interpreting program. This program then interprets the STL's rendering of the model using two numerical method approaches designed specifically for generating computer simulated AFP trials. The first numerical method used begins by creating points along the surface of the shape. To do this, we use equidistant plane slices to intersect the triangular plane edges which will be where the points for path generation will reside [9]. The second numerical method used involves the generating of path vectors using these equidistant points about the surface of the object and both method coordinate systems are based on the Frenet frame. This portion of the code will provide a path vector at each of the surface points that the slices create. The average of 2-3 of these vectors creates the pathing for the computer simulated AFP trials. However, to engage a KUKA robot, this newly generated pathing (with its normal referencing) must finally be converted into a unique RSL file which the machine will understand. The RSL file will also allow for the component model to be moved to start a new line of carbon fiber placement while the robot simulation program itself will allow for the testing of code viability.

2.1. Model analysis

The generation of the tool path for the roller mounted to the robotic arm begins with an STL model of the part of interest. Using our in-house Java based interpreting program, the STL model is first imported into memory for reading and writing capabilities. It is by doing this first that we may then later apply numerical methods to the patterned information within the STL to create a more usable set of information based on user input. User inputs include slicing angle, also known as placement angle or fiber angle, and number of sections. A slice is a term used to describe the angled placing an iterative plane through a model which will be used for the linear mapping of the object's surface. In additive manufacturing such as fiber placement, this can enable the use to reinforce both the materials and the structures being created [4]. The number of sections is the division of the objects width in relation to its given slicing angle of rotation. This allows for the determining of each consequent slice angle to be both equidistant and parallel. Generally, good slicing angles run the length of the object but may vary depending on the curvature of the component model. This section slicing methodology is important because it allows the user to adapt the tool's
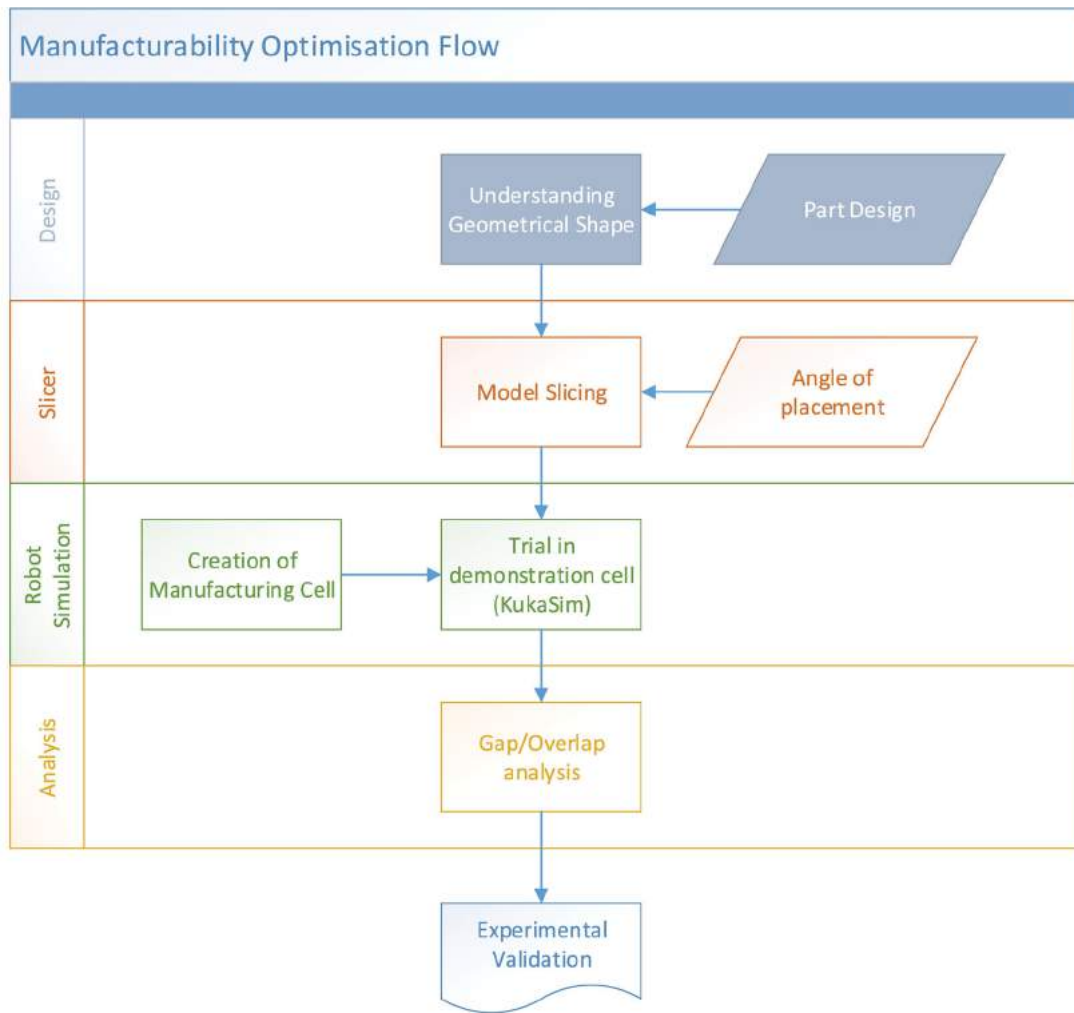
Manufacturability Optimisation Flow

Figure 1. The manufacturability study flow for the part in question.

tow width for each automated lay-up process; avoiding both tow overlaps and tow gappage [2]. Our simulations ranged from a single tow width of ¼ in to an eight tow width of 2 in which is within the range of the AFP industry's technological capabilities [8]. This range is vital to the efficiency of coverage in this study because most of the parts to be processed for this study exceed 10 feet in length and 5 feet in width. Each facet that makes up the mesh contains three vectors which define the 3 points of the triangle, as well as a vector that defines the outward-pointing face normal of the facet. In order to generate the slices, the code searches for the facets which intersect the plane, and a line is found across this intersection. A set line is the summation of each of these lines about each individual defined plane. The total number of set lines, which are found at this intersection, defines the cross section of the model. Figure 2 shows a sliced rendering of the component model. Notice how each slice is depicted by the set lines which run across the surface

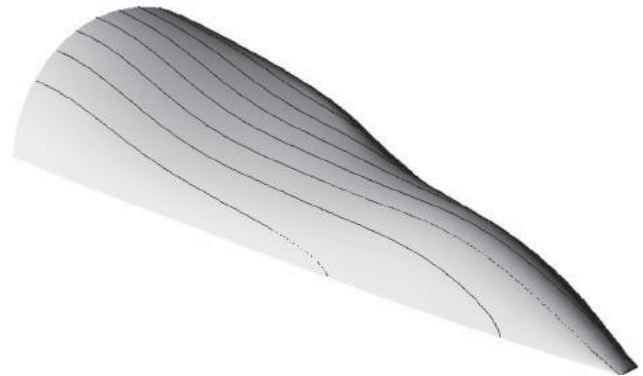of the component and used as a generated pathing for an AFP tool.

Figure 2. The black lines indicate the generated pathing along the surface of the model which was divided into 8 equally spaced sections at 0 degrees for demonstration.

After the initial generation of the slice, which contains the direction vector (Z-axis) orthogonal to the surface at the vector location of the point. Now the X and Y axes must be generated for the points to allow for a smoother orientation of the machine head throughout the tool path [3]. The X-axis is defined by averaging the directions from the surrounding points to the point of interest. The point of interest is defined to be the directional vector from which an averaging of a previous point and a future point are made before normalizing the length of the vector. The X-axis represents the gradient on the surface at this vector, and points towards the next vector along the path. The Y-axis, simply generates from the cross product of X, Z axis, and it is orthogonal to both.

2.2. Generating the robot scripting language (RSL)

The information required to generate the robotic motion instructions along the surface of the object have been provided to this RSL generation module from the path generation portion of the code. To define a motion point for the robot, the vector location of the point and the XYZ axes must be defined. The XYZ axes define the orientation for the robot's tool at the point along the path. The RSL generation module contains methods which simplify the generation of RSL. The methods use the necessary inputs, primarily vector location and directional axes, and reformats it with the RSL markup syntax, in order to generate the final functional code. Additional instructions are later added automatically upon the first time opening and running of the new RSL file in the KUKA Sim Pro 2.2 program. These additional instructions extend to Boolean inputs and outputs which serve to create a more realistic simulation, adding functionality beyond blindly following a provided path. They account for motion routines between individual sections of the paths, as well as changing the configuration of the robot and manipulating the available binary I/O ports. The binary ports, while not required for this approach, can be used to communicate with other machines within the workcell, as well as enable tracing of tool's motions for visual inspections of its movements. The types of motions available to control the robot are linear and point to point. The linear motion directly translates the tool data, which is the point defined at the end of the tool, as well as the tools orientation, along the path between the current position and the specified coordinate. This point to point method involves robot pathing movements, from start to finish, which are generated by the KUKA Sim Pro 2.2 software upon opening the created RSL file. Depending on the workcell setup in use, this point to point method generated may result in unpredictable and undesirable movements.

A limitation of the RSL is that the position along the rail, also known as the external axis, is not internally computed by the robot's motion controller. Therefore, the code developed for converting the path locations into motion points for the robot needs to compute the optimal travel distance along the external axis in order to complete the instructed motion.

The base of the robot may be aligned with the end effector, or the base may be staggered from the head up to the length of the arm itself. The variation in this staggering along the external axis allows the robot to reach extreme locations on the surface, which may be very close to the rail or distant points. In general, the variation along the external axis significantly expands the working envelope available to the robot. More distant points may also be brought into the working envelope of the robot by manipulation of the surface's orientation with regards to the rail.

The available manipulation was the simulated effect of a mandrel for the testing. A connection between the robot and the part was created, and the robots code could send a signal which would initiate a partial rotation of the surface. Rotation of the particular part was performed about its long axis, and the rotation could be stepwise or continuous depending on the pathing requirements.

2.3. Workcell organization

Several workcell organizations were prepared to account for the limitations between each object's physical boundaries and the optimal degrees of freedom between the robot and the given model. Limitations for the robot include both its movement path and its reach about the model's surface. Movement path limitations include the robot's general restriction to a finite linear track and arm movements which do not collide with the track, the model, itself, or go beyond the designed tolerable axes of motion for the robot. For reach limitations, it was vital to account for the fiber placement processes needed to be applied at an angle which is normal to the surface of the object.

To address this specific reach requirement, several simulated setups were created in KUKA Sim Pro 2.2 for our approach, in which inner and outer boundary tests were systematically performed. First, it was determined that the initial orientation of the object, relative to the robot, would need to run parallel to the robot's track to utilize the robot's linear range. Second, the model was assessed statically and dynamically (rotating about a single axis), at varying angles, degrees of rotations, and general proximity from the robot to determine feasibility. In Figure 5, KUKA Sim Pro 2.2 simulates a demonstration
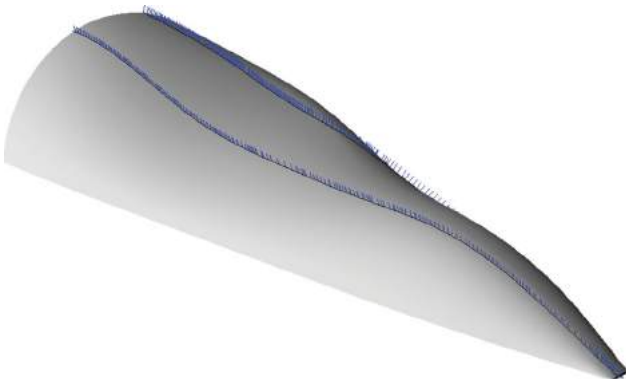
Figure 3. The blue lines represent the normal axis for each point defined about two arbitrarily selected paths from Figure 2 along the surface of the model.
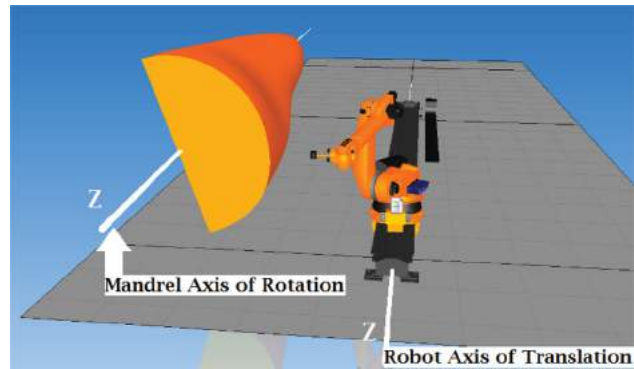

Figure 5. Defining both the mandrel's axis of rotation and the robot's axis of translation in parallel of one another.

workcell in which the mandrel would perform a partial rotation about its long axis which would be positioned in parallel with the robot's rail. This optional mandrel rotation would enable the robot to potentially reach a wider range of consecutive paths.

Each workcell was populated by a KR-120P KUKA Robotic arm, which rode upon a KL-10000, a 10 meter rail. A simple placeholder tool was constructed and mounted onto the KR's end effector to simulate the use of a fiber placement type device. Motion instructions

for pathing along the surface were generated using an approximately 10 meter long, conical shape, with a roughly 3 meter radius. The generation of the path is restricted to the outer surface of the model and also incorporates predetermined aspects of the simulation, including workcell configuration and the parameters specific to the robot's orientations and limits [7]. Below is an example of actual motion pathing performed by a robot for horizontal pathing while the mandrel rotates about the component model's long axis. Each path is a series of points as a reference from which the robot is able to orient

Figure 4. Pictures of the developed slicer applied on a portion of the investigation part

itself without further manipulation of points during the simulation. This method of displaying the points, shown in green, enables pre-run visual confirmation of the validity of the entire pathing program. Figure 6 demonstrates how each path is coordinated with a specific rotation of the shape, and will match smoothly against the surface during the completion of the particular path.
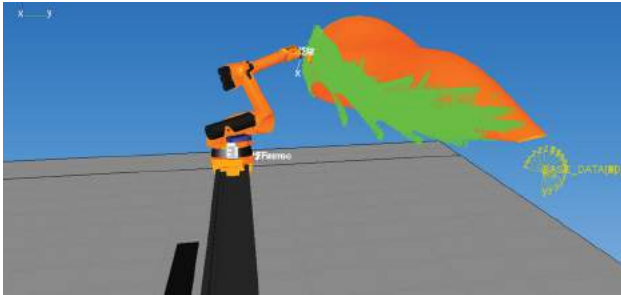

Figure 6. Simulation of viable robot tool pathing, shown in green, along the rotated length of the component model.
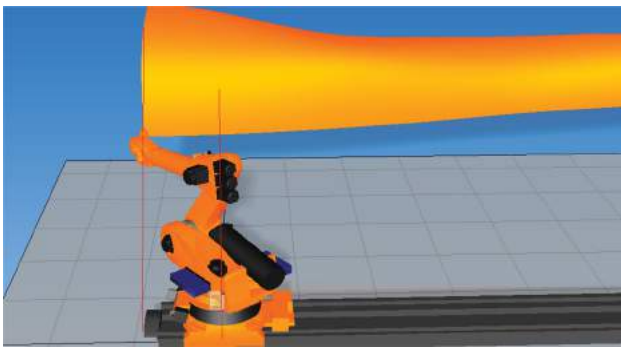

Figure 7. Example of the staggering between the head and the base. The use of the staggering allows the shape to be placed closer to the model, enabling the tool to reach points closer to the rail as well as more distant points. Here the robot uses a 1000 mm stagger to access the point.
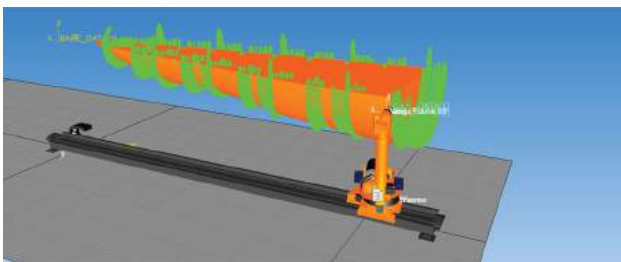

Figure 8. Example of a top down configuration which relied heavily on the variation of the external axis in order to avoid crashing against the rail, as well as manipulating the robot's working envelope to reach more points which lie further away from the rail, necessitating longer reach from the robot.
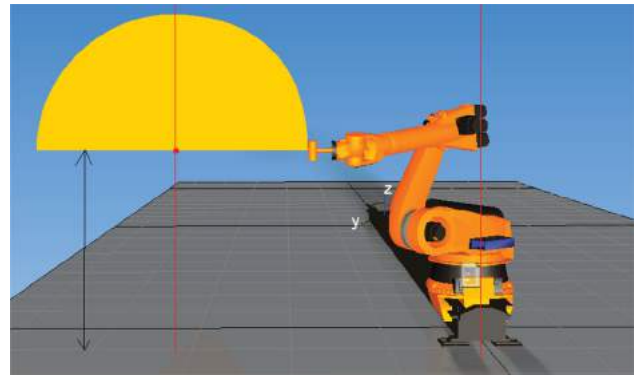

Figure 9. Typical dimensions for the offset configuration, the part location relative to robot base: the long axis of model is 3200 mm from the rail. The axis or rotation for the model is 2000mm above the floor.
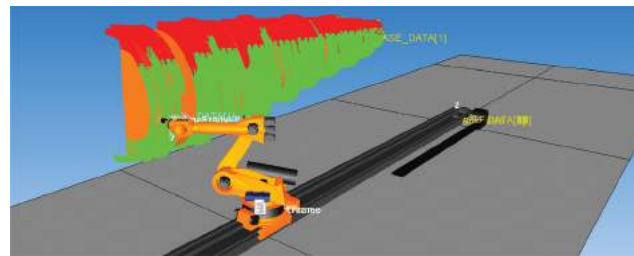

Figure 10. Demonstration of the safety points, shown in red, represent areas either out of reach to the robot or a potential collision between the robot and the part within the workcell during execution. After each pass, the robot backs out from the surface as it moves to the beginning of the next path, and the mandrel rotates the part to allow better accessibility.

The primary use of the mandrel was during zero-degree pathing along the long axis of the part. Stepwise rotation was used to bring each path into the robot's reach. The rotations would be signaled after the current path had been finished, and the robot removes itself from the surface. After the completion of the parts rotation, the robot would touch down to the surface and continue following the paths.

Rotational control of the mandrel was also attempted for non-zero angles, however, such a configuration is not suited for stepwise rotations. Instead, the mandrel would be set to a constant rotation rate, and the head would move accordingly for the path, all the while maintaining a constant height. Such a configuration would ensure full coverage of the surface, however, a great deal of coordination between the speed of the robot and the mandrel rotation speed would be necessary to ensure continuous contact throughout the entire path, as well as ensuring a constant speed between the tool data and the surface (similar to actual AFP processes). Brief ventures into
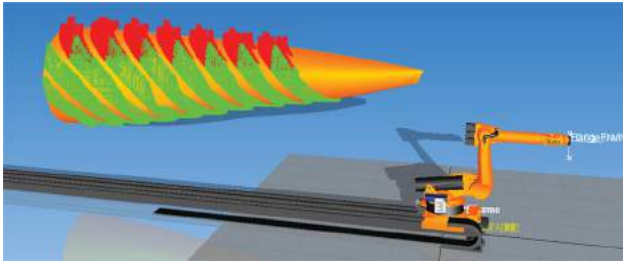
Figure 11. Demonstration of 0° and -45° pathing along the surface. Green areas indicate the possible motion points, while red areas are nonviable. These extreme points require physical manipulation of the workcell configuration to bring them into reach of the tool. The most likely solution would be the use of a mandrel to rotate the part for better accessibility.
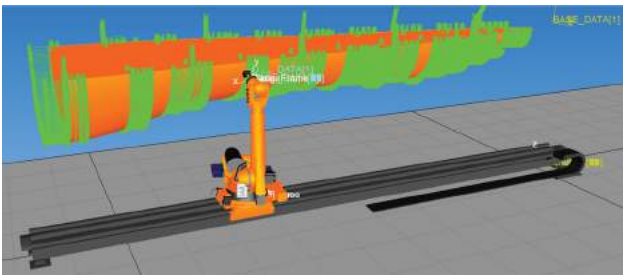

Figure 12. Positioning the part above the rail enables complete coverage of the surface by the tool. The configuration's success results from the reduction of extreme points, either too near or distant.

such a configuration encountered problems with timing between speed of path coverage and the mandrels rotation. As a result, the rotation of the mandrel would outpace the speed of the tool, such that all coordination would be lost. This challenge highlights the limitations held within the engineering software's inability to variate rotational speeds of the component model based on the robot's position and could be addressed in later work.

2.4. Path evaluation

The two pre-requisites for path evaluation, are the generation of the robotic instructions as well as the modeling and configuration of the workcell that the robot interacts with. The tool used for path evaluation was KUKA Sim Pro 2.2. The robotic instructions generated by the initial code are transferred to KUKA Sim Pro 2.2. The KUKA software provides many useful tools for evaluating and improving programs. In terms of evaluation, KUKA's software simulates the robot's motor controllers, such that it can internally determine whether or not specific motion points are within the robot's reach. The software also allows for visual inspection of the motion pathing, as the robot's motions are simulated within the software,

such that anomalies may be noted and used to improve the code for generating the robotic instructions. The particular anomalies encountered throughout testing varied from flipping of the robot's end effectors as joints hit their rotational limits, in addition to the "crashing" of the arm into different components within the workcell. The general accessibility of the path by the robot and the presence of the physical anomalies are used to evaluate the overall viability of the path generated.

3. Concluding remarks

Through this work, several accomplishments and further challenges exist. In regards to the double-curved freeform tool pathing, this study succeeded in first reading a component part model into memory, allowing the user to input both the tooling angles and the number of sections, while finally creating a usable RSL. This was accomplished through the creation of an in-house tool path generating software. Regarding workcell optimization, this study determined that the length of the part should be running parallel to the robot's track, there by translating the part within the X and Y axis to reduce or remove safety points, and demonstrate repeatability through simulations. However further challenges exist in regards to eventually reducing or removing both tool path overlap and tool path gappage in how it may negatively impact the laying of fibers about a part [2]. Additional modifications for the program would also be needed in the way of accounting for the width of the tool being used in the context of a selected manufacturing process [8]. In the case of fiber tows, both the width of the tows and the number of tows would need to be definable by the user. The last challenge for further development is improving computational efficiency because component parts may increase in both size and complexity and must be optimized. [1].

Acknowledgment

ORCID

J. Halbritter ⓘ http://orcid.org/0000-0001-5747-2875
Ramy Harik ⓘ http://orcid.org/0000-0003-1452-9653
A. Zuloaga ⓘ http://orcid.org/0000-0002-7461-9610
M. vanTooren ⓘ http://orcid.org/0000-0002-3806-5294

References

[1] Backer, W.-D.; Kirkpatrick, M.; Harik, R.; Tarbutton, J.: Automated reconstruction of continuous robotic motion from g-code patterns, Online Proceedings CAD'16, Vancouver, Canada, 316–322, 2016. http://www.cadconferences.com/cad16_316-xxx.html#.V4NTxI5U_xk

[2] Bruyneel, M.; Zein, S.: A modified fast marching method for defining fiber placement trajectories over meshes, Computers & Structures, 125(1), 2013, 45–52. http://dx.doi.org/10.1016/j.compstruc.2013.04.015

[3] Debout, P.; Chanal, H.; Duc, E.: Tool path smoothing of a redundant machine: application to automated fiber placement, Computer-Aided Design & Applications, 43(2), 2011, 122–132. http://dx.doi.org/10.1016/j.cad.2010.09.011

[4] Doherty, S.; Backer, W.-D.; Bergs, A.-P.; Harik, R.; Tooren, M.-V.; Rekleitis, I.: Selective directional reinforcement of structures for multi-axis additive manufacturing, The Composite and Advanced Materials Expo, Anaheim, California, 2016. http://www.thecamx.org/wp-content/uploads/CAMX-2016-Sessions.pdf

[5] Khennane, A.: 8 - Filament winding processes in the manufacture of advanced fibre-reinforced polymer (FRP) composites, Woodhead Publishing Series in Civil and Structural Engineering, Sawston, Cambridge, 2013, 187–206. http://dx.doi.org/10.1533/9780857098641.2.187

[6] Lopes, C.: Damage and failure of non-conventional composite laminates, Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2009. http://www.lr.tudelft.nl/?id = 27577&L = 1

[7] Marsh, G.: Automating aerospace composites production with fibre placement, Reinforced Plastics, 55(3), 2011, 32–37. https://dx.doi.org/10.1016/s0034-3617(11)70075-3

[8] Tooren, M.-V.; Elham, A.; Harik, R.: Optimal variable stiffness distribution for a composite plate with a row of holes subjected to tension/shear load cases. 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Dallas, Texas, 1–14, 2015. http://dx.doi.org/10.2514/6.2015-2488

[9] Zhang, Y.; Backer, W.-D.; Harik, R.; Bernard, A.: Build orientation determination for multi-material deposition additive manufacturing with continuous fibers. The 26th CIRP Design Conference, Stockholm, Sweden, 2016. http://cirpdesign2016.org/program-2/