

CAM-Based Planning, Programming and Execution of Large-Scale Machining Operations by a Robot-Mounted Gantry System

Jason Michel Lambert, Martin De Montigny and Claude Perron
National Research Council Canada

ABSTRACT

This paper examines issues related to planning, programming and execution of machining operations by a robot in the context of machining large parts with complex geometries by a gantry-mounted robotic system. Parts were created from surface data in a CAD/CAM environment. The same environment was used to generate tool paths using a conventional machine tool approach. These paths were converted to robot trajectories and validated using mathematical kinematic models of the robotic system. Validation was performed according to various criteria related to process performance. Associated robot programs were then automatically generated. The manufacturing cell was progressively integrated according to requirements resulting from iterative process characterization. A metrology-based calibration procedure was designed that considerably improved the system's positioning precision. A custom user interface for online process monitoring was created as well as online cell program management capabilities to accommodate the large quantity of trajectory information. The system went from conceptual, to prototype-level to full production mode, thereby covering several Manufacturing Readiness Levels (MRL). The various strategies used in planning the process and programming the system demonstrate the viability of CAM-based approach for robot machining path generation without resorting to the use of OLP packages for validation of program generation. This could create a shift in the dependency on robotic specialists, allowing machine tool experts to carry about the programming of robot machining cells using the conventional machine tool approach.

CITATION: Lambert, J., De Montigny, M. and Perron, C., "CAM-Based Planning, Programming and Execution of Large-Scale Machining Operations by a Robot-Mounted Gantry System," *SAE Int. J. Mater. Manf.* 5(1):2012, doi: 10.4271/2011-01-2651.

INTRODUCTION

The use of robotic systems for machining applications is an emerging trend in aerospace manufacturing. Robots possess characteristics that make them an interesting alternative for tasks that are not part of the core efficiency of machine tools. Their flexibility and reconfigurability allow them to address a great variety of parts within one manufacturing cell configuration, as well as having different operational modes. This makes robotic systems well suited to peripheral and complementary machining tasks such as pre-machining, part finishing and inspection, or a combination of these tasks. Enablers of this trend are stronger, larger payload robots, and the integration of external sensor information like metrology systems, vision and force sensors to augment the system's capabilities. However, despite the improved performance brought about by these new technologies, some

aspects inherent to the use of robotic systems have yet to meet requirements brought about by machining and continuous path applications in general. The stiffness found in robot joints and links create static deformations and vibrations in the presence of contact which can degrade quality of the machining process, especially when the robot is mounted on a gantry system to increase its workspace. Off-Line Programming (OLP) software packages used to perform robot programming and simulation lack the sophisticated, geometric feature-based path generation capabilities of Computer-Aided Manufacturing (CAM) software. Robot trajectory validation is essentially performed visually using graphical simulation, providing limited feedback with respect to system aspects such as configuration-dependent precision and stiffness. Also, the large number of data points typical of surface-referenced tool paths, especially for large parts with complex geometries can be demanding to the point of making the graphical-based validation approach unusable. This paper

presents the development and implementation of a robot path generation and validation methodology that integrates within the information flow of a manufacturing cell programming for machining applications. This methodology was developed out of a need to provide as seamless an integration as possible given the process requirements. The process involved machining large parts with a detailed geometry, which was expected to require sophisticated path generation and result in the management of a large quantity of path data. Furthermore, as the project was initiated at a very conceptual stage and many aspects of the process had to be validated iteratively, the manufacturing cell being progressively assembled. As a consequence, the use of 3rd-party OLP software was deemed inappropriate. A means for trajectory generation and validation had to be devised that could better fit within this context.

PROJECT REQUIREMENTS

Initial requirements for the project came in the form of an inquiry about the possibility of using the Canada's National Research Council Aerospace Manufacturing Center (NRC-AMTC) robotic facilities in order to machine large parts containing elaborate and diverse features. The parts were to be used in reproducing a scaled-down model of section of a coastal line for wind tunnel aerodynamic testing purposes. The final model was estimated to be $8\text{m} \times 15\text{m}$ in size. The many diverse features to be machined came from the elaborate topology of the coastal terrain (mountains, valleys, cliffs) which had to be faithfully reproduced. [Figure 1](#) illustrates the terrain geometry to be machined.

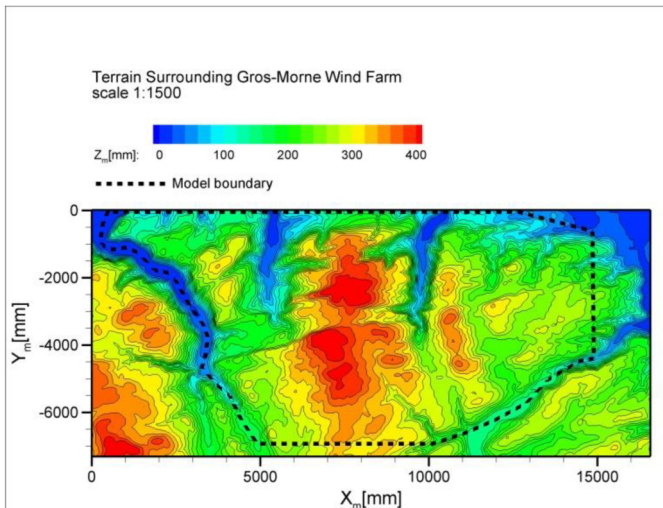


Figure 1. Topological terrain representation

The model was to be machined out of soft, light material that could be easily moved around for assembly in the wind tunnel. Surface finish requirements were such that adequate flow characteristics could be reproduced. In order for this, the model had to incorporate vertical discrete steps 3mm high, each step having to be fairly smooth. Surface finish

requirements were quantitative and had to be evaluated thus. Previous models of similar terrain had been generated using conventional machine tools. This approach proved to be rather tedious and time-consuming due to the model having to be broken down in several small pieces to accommodate the machine tool workspace. This generated considerable setup time that still monopolized the machine tool, and increased model inaccuracies resulting from error accumulation when reassembling the model. In order to palliate these issues, a gantry-based approach was considered. Furthermore, given that the material to be used could be soft enough so as to generate low process forces while machining, and that machining tolerances ($>0.5\text{ mm}$) were fairly relaxed from a machining standpoint, it was possible to envision using a robotized manufacturing solution in performing this task. For logistics purposes, it was decided to section the model in $1.2\text{m} \times 2.4\text{m}$, resulting in 26 sub-parts.

MANUFACTURING CELL DESCRIPTION

The system used to manufacture the parts was specifically designed to address the goal at hand. At the beginning of the project the approach envisioned was purely conceptual that is, the system or its individual components were never used for this purpose. The methodology used to assemble the robotic cell was one based on iterative prototyping, where sub-elements of the global process were sequentially addressed, and the necessary cell elements combined so as to progressively cover all aspects of the process. Even though the cell was integrated in a research environment, the final system was fully capable of entering production mode, and has done so with hardly any modifications right from the first part being machined. Hardware and software elements were closely integrated such that a continuous flow was present at all stages of the process, from path programming right down to execution and monitoring.

ROBOT-GANTRY SYSTEM

The main cell component is the AMTC robot-gantry system ([Figure 2](#)). It is comprised of a 3-degree of freedom (dof) gantry with Cartesian X, Y and Z positioning capabilities. X-displacement is achieved by moving the inverted “u” structure along two parallel rails. Y-displacement is accomplished by moving the tower along the inverted “u” structure, whereas Z-displacement is performed by extending and retracting the telescoping tower. The gantry has a $6\text{m} \times 6\text{m} \times 6\text{m}$ working envelope. At the gantry's end-effector is mounted an inverted 6-dof industrial serial manipulator.

The robot, a KUKA kr-210 with enhanced absolute accuracy, has a 210 kg payload. Its movement can be coordinated with the gantry so that both can take part in executing a given trajectory, or can be independently operated, while the gantry is stationary. The robot is equipped with an ATI delta 6-dof force sensor mounted at the end of its



Figure 2. Robot-Mounted Gantry System and Spindle

wrist. At the end of the robot is a Pushcorp STC1503 spindle and tool changer. The spindle has a 2.23 kW electrical servo motor and has closed-loop speed control from 60 to 15 000 rpm. The spindle was directly mounted on the robot, its rotation axis in-line with the last joint's rotational axis. Two types of cutting tools were used. For roughing, a 2-flute cutter with 25.4 mm diameter was used, whereas for finishing, a 2-flute cutter with 25.4mm radius was used.

CELL ENVIRONMENT

For security purposes as well as to restrain the spread of residual material, an enclosure was built around the robot-gantry workspace. Curtains were laid on top and sides of the enclosure. A front panel was built with transparent panels to provide visual access to the operator. A camera was also located inside the enclosure to provide additional monitoring views. A commercial dust collector was fitted to the spindle using a custom-made nozzle. [Figure 3](#) depicts the cell environment as seen from the operator station

To manage and coordinate operation of the robot, dust collector and spindle, an Allen Bradley SLC-500 PLC was used. Its main functions were to command and monitor spindle speed, activate dust clearing mechanisms and generate emergency states in case of speed discrepancies or spindle overheating.

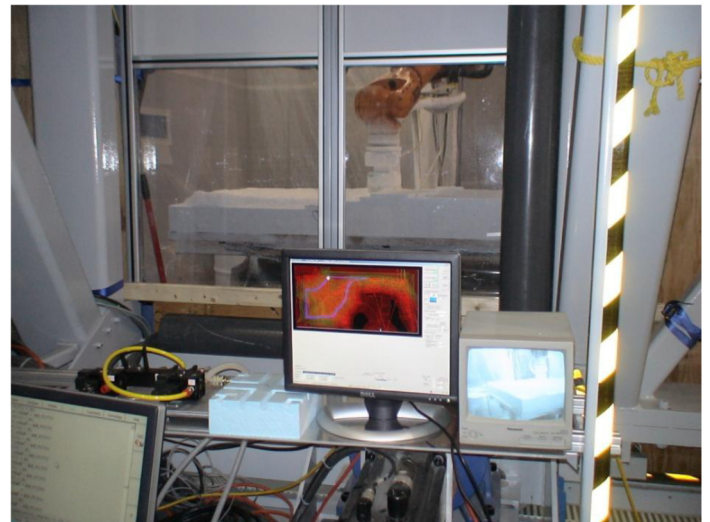


Figure 3. Cell environment

COMMUNICATION & CONTROL

The robot-gantry controller is the central control unit of the cell, making requests and taking input from the other elements. Process parameters are included in the robot programs, and appropriate cell actions are taken based on these parameters. Conversely, feedback on the cell elements are relayed to the robot controller, the latter taking required measures to ensure safety of the process and environment. The main control elements are summarized in [Figure 4](#)

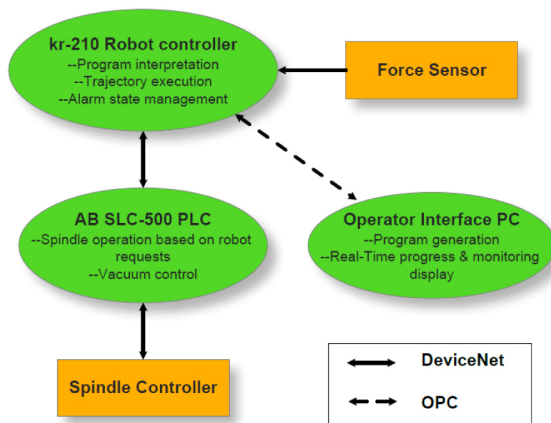


Figure 4. Control & communication channels

The operator interface PC performs program generation and buffered path data transfer to the controller. In addition it provides a graphical monitoring display of the task progress (Figure 5). A map of the part being machined shows the completed area, the area currently being machined, whose path information is held in memory, as well as the remaining area to be machined. Remaining time to completion and average speed are also displayed.

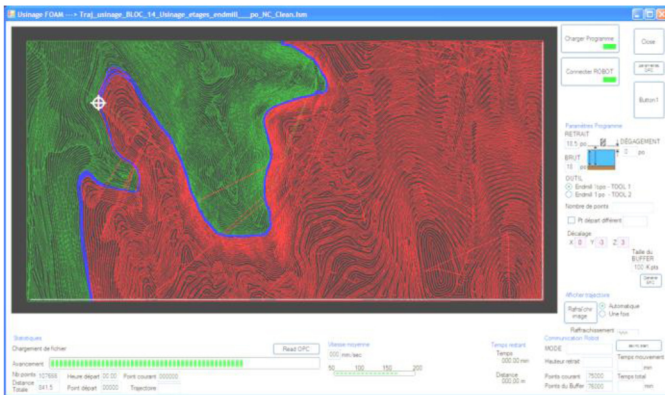


Figure 5. Operator graphical interface

PATH GENERATION AND VALIDATION

Typically, robot manufacturing cell path generation, validation and programming all take place in one integrated environment, the so-called Off-Line Programming (OLP) packages. These packages heavily rely on visual feedback and graphical simulation in order to program a cell. The option of using an Off-Line programming package for this particular project was discarded for the following reasons:

- **Large number of path points:** A machining application requiring many curved paths require a large number of points, if the curve is to be smooth enough to meet tolerances. The fact that the parts were to describe natural terrain features

implied that most of the path would indeed contain many curves. Using a 3-D graphical environment to validate all these points would be both time consuming and resource-intensive in terms of computer memory, so much as to make the approach not feasible.

- **Need for a quantitative measure of closeness to singularities:** Singularities are points in the workspace where the robot loses controllability. Depending on the robot configuration, its ability to execute precise motions can vary, and degenerates at singular configurations. For this machining application, where precision must be maintained along the path, it was necessary to get a measure of this dexterity, to locate points where the manipulator would be more sensitive to control errors. This feature is not present in OLP packages.

- **Inclusion of a path compensation mechanism:** As described below, compensation for the gantry's flexibility needed to be performed on the trajectory points prior to validation.

- **Remove dependence on OLP software for machining applications:** OLP packages tend to be expensive and have a steep learning curve, even for someone with a background in robotics. Given that machining tasks are usually generated by machine tool specialists with no knowledge of robotics *a priori*, the inclusion a sophisticated software tool along with the need for operation of this tool by specialists can compromise viability of a robotized machining approach. Developments for this project were aimed at providing tools for path validation which do not require additional OLP capabilities, and which can eventually be used by people with minimal background in robotics.

Consequently, path generation and validation was implemented as multistage process in Matlab/Simulink, taking as input APTsource code, which is a direct export of most CAM software packages. The Matlab/Simulink environment provides an ideal framework for implementing the necessary algorithms, and has the capabilities required for further expansion. Furthermore, all the developed modules can be compiled and used in a standalone fashion, thereby removing dependency on additional software.

DATA FLOW

Figure 6 illustrates the various stages of the validation process. Firstly, tool paths are created in a conventional manner using, in this case, the Catia's Advanced Machining module. Path points are output in the APTsource format, again by conventional means. This file serves as input via Matlab to a Simulink model. Path points and orientations are converted according to the relative robot/part positions for the specific cell. The model then analyzes each point, to which corresponds a specific robot configuration, in a sequential manner using as a basis the robot's kinematic description. For each point, the following assertions are made: