2008

Robotic machining from programming to process control

Zengxi Pan
University of Wollongong, zengxi@uow.edu.au

H. Zhang
ABB Corporate Research China

Recommended Citation

Pan, Zengxi and Zhang, H.: Robotic machining from programming to process control 2008.
https://ro.uow.edu.au/infopapers/681

Robotic machining from programming to process control

Abstract

This paper presents the critical issues and methodologies to improve robotic machining performance with industrial robots. A complete solution using active force control is introduced to address various issues arouse during the robotic machining process. Programming complex contour parts without a CAD model is made easy using force control functions such as lead-through and path-learning. The problem of process control is treated with a novel methodology that consists of real-time deformation compensation for quality and controlled material removal rate (CMRR) for process efficiency. Experimental results show that higher productivity as well as better surface quality can be achieved, indicating a promising and practical use of industrial robots for machining applications that is not available at present.

Keywords

force control, robot programming, CMRR, deformation compensation

Disciplines

Physical Sciences and Mathematics

Publication Details

This conference paper was originally published as Pan, Z and Zhang, H, Robotic machining from programming to process control, Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China, 25-27 June 2008. Copyright Institute of Electrical and Electronics Engineers 2008. Original conference paper available here

Proceedings of the 7th
World Congress on Intelligent Control and Automation
June 25 - 27, 2008, Chongqing, China

Robotic Machining from Programming to Process Control

Zengxi Pan
School of Electrical, Computer and Telecommunication
Engineering, University of Wollongong
Australia

Hui Zhang
ABB Corporate Research China
Shanghai, China
Hui.zhang@cn.abb.com

***Abstract*—This paper presents the critical issues and methodologies to improve robotic machining performance with industrial robots. A complete solution using active force control is introduced to address various issues arouse during the robotic machining process. Programming complex contour parts without a CAD model is made easy using force control functions such as lead-through and path-learning. The problem of process control is treated with a novel methodology that consists of real-time deformation compensation for quality and controlled material removal rate (CMRR) for process efficiency. Experimental results show that higher productivity as well as better surface quality can be achieved, indicating a promising and practical use of industrial robots for machining applications that is not available at present.**

***Keywords*—force control, robot programming, CMRR, deformation compensation**

I. INTRODUCTION

Cleaning and pre-machining operations are major activities and represent a high cost burden for casting producers. Machining processes, such as cleaning, milling, grinding, deburring, and saw cutting are promising applications for industrial robot with the drive from foundry automation. From robotic machining point of view, two types of machining processes could be distinguished. The first type, typically cleaning and deburring, usually has a very complex 3D curved cutting path, a crucial cycle time requirement, and relative low surface accuracy. Today, most of the deburring operations are done manually in an extremely noisy, dusty and unhealthy environment. Therefore, automation for these operations is highly desirable. The second type is milling process, in which robot moves in a simple path with lower feed speed (20~30 mm/s), while heavily engaging with the workpiece. The controller must be accurate enough to maintain the surface quality under large and varied machining force. This type of machining is currently conducted by CNC machine, which can be justified economically only for large batch sizes.

This research will propose a robotic machining strategy for foundry industry with small to median batch sizes. The strategy is a complete solution addressing the difficulties for both types of machining applications from programming to process control. Based on an active force control platform, different control strategies are implemented including lead-through, path-learning, CMRR, and deformation compensation to facilitate certain process requirement.

This paper is organized in six sections. Following this introduction section, section two describes several major challenges for robotic machining process. Section three provides the introduction of an active force control platform, which is the foundation for various control strategies. Section four addresses the programming issues for a complex contoured part. With two force control strategies, lead-through and path-learning, robot programming is made easy and efficient. Section five presents two realtime process control techniques. The realtime deformation compensation improves the quality and accuracy of the robotic machining operation, while controlled material removal rate greatly reduces the process cycle time. Experimental results are presented at the end of section four and section five. A summary is provided in section six.

II. CHALLENGES

Robotics based flexible automation is considered as an ideal solution for its programmability, adaptivity, flexibility and relatively low cost, especially for the fact that industrial robot is already applied to tend foundry machines and transport parts in the process. Nevertheless, the foundry industry has not seen many success stories for such applications and installations due to the several major difficulties involved in robotic machining process with a conventional industrial robot. [1]

The first difficulty is the generation of robot motion for a complex workpiece. Concerning robot programming, online programming method has conventionally been carried out by skilled workers guiding the robot through the desired path using a teach pendant, namely the jog-and-teach method. Although the concept is simple, it is not feasible for many machining processes especially for deburring process, which has a great number of teaching points and requires high positioning accuracy. An operator must constantly guide the robot through motions accurately which is usually a very time-consuming task. Offline programming method, which extracts the robot targets from CAD data of a workpiece, is another choice. [2][3] Although off-line programming is more accurate and flexible, it is only cost-effective for large batch sizes. Since it relies heavily on the modeling of the robot and workpiece, additional calibration procedures are usually inevitable to meet the process accuracy requirement.

Today, both on-line and off-line programming methods are still too expensive, time-consuming and difficult. Efficient techniques for automatic robot programming must be applied. We will address this issue by presenting a programming by demonstration (PbD) method which could minimize the burden of robot programming.

The second difficulty is the deformation caused by the interaction force between tool and workpiece, especially for milling process which generates large cutting forces. The stiffness for a typical articulated robot is usually less than 1 N/μm, while a standard CNC machine very often has stiffness greater than 50 N/μm. As a result, force induced deformation is the major source of the inaccuracy of finished surface. A perfect robot program without considering contact and deformation will immediately become flawed as the robot starts to execute the machining task. Unlike multi-axis CNC machine centers, such deformation is coupled and varies even subjected to the same force at different workspace locations. Such coupling results in deformation not only in the direction of reaction force and can generate some counter-intuitive results.

Thirdly, the lower stiffness also presents a unique disadvantage for machining of casting parts with complex geometry, which is non-uniform cutting depth and width. As a result, the machining force will vary dramatically, which induces uneven robot deformation. In general practice, machine tools maximize the material removal rate (MRR) during roughing cycles by applying all of the available spindle power to the machining process. In conventional robot programming and process planning practice, the cutting feed rate is constant even with significant variation of cutting force from part to part, which dictates a conservative cutting feed rate without violating the operational limits. Therefore, it is desirable to maximize MRR and minimize cycle time by optimizing the machining feed speed based on a programmed spindle load.

The fourth difficulty is chatter/vibration occurred during the machining process. [4] Chatter/vibration becomes a more important issue in robotic machining process due to the low stiffness and coupled structure of industrial robots. Robotic engineers and technicians are frustrated to deal with elusive and detrimental chatter issues without a good understanding or even a rule of thumb guideline. Very often, to get their process working correctly, one has to spend tremendous time on trial and error for the sheer luck of stumbling a golden setup or has to sacrifice the productivity by settling on conservative cutting parameters much lower than the possible machining capability.

Most of the existing literature on machining process, such as process force modeling [5], accuracy improvement [6] and vibration suppression [7] are based on the CNC machine. Research in the field of robotic machining is still focused on accurate off-line programming and calibration. In literature, a number of references can be found concerning one or two of the challenges mentioned above. However, system coping with all the above challenges must be available in order to enable a large-scale penetration of robots into the area of machining process. This paper presents the functional structure of such a system

As the chatter analysis was discussed in a separate paper [4], our focus here is to address the first three major issues in robotic machining process: 1)To generate robot program with complex 3D curvature easily without experienced technician and CAD model; 2)To improve the machining quality with the low stiffness, low accuracy robot; 3) To improve the robotic machining efficiency by providing realtime optimization to maximize material removal rate.

III. FORCE CONTROL PLATFORM

The active force control platform is the foundation of strategies adopted to address various difficulties. It is implemented on the most recent ABB IRC5 industrial robot controller which is a general controller for a series of ABB robots. The IRC5 controller includes a flexible teach pedant with a colourful graphic interface and touch screen which allows user to create customized HMI very easily. It only takes several minutes for a robot operator to learn the interface for a specific manufacturing task and it is programming free. An ATI 6 DOF force/torque sensor is equipped on the wrist of the robot to close outer force loop to realize implicit hybrid position/force control scheme. The system setup for robotic machining with force control is shown in Figure 1. The flexible force controller could be configured differently to satisfy various application needs. The block diagram of the force control loop is shown in Figure 2.

Figure 1. System Setup for Robotic Machining with Force Control

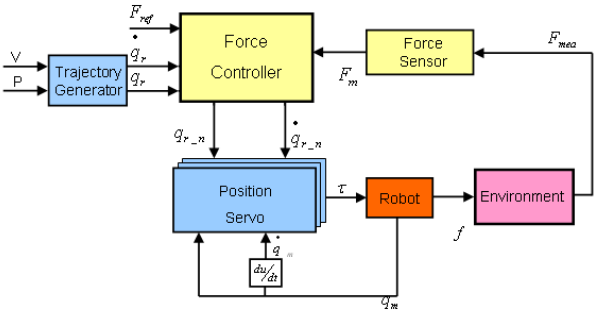

Figure 2. The Force Control Loop

While the conventional position control is realized in joint space, force controller is implemented in Cartesian space. The difference between the reference force and the measured contact force is input to the force controller. If certain direction is set to be under force control, the force deviation calculates a correction to the robot's nominal position and changes the reference position and speed given by robot trajectory generator.

The force controller provides two major functions to make the entire programming process collision free and automatic. First function is lead-through, in which robot is compliant in selected directions (force control directions) and stiff in the rest of directions (position control directions). To change the position or orientation of the robot, the robot operator could simply push or drag the robot with one hand. The second function is called path-learning, in which robot is compliant in normal to path direction to make the tool constantly contact with work piece. Thus, an accurate path could be generated automatically.

During the machining process, the force controller provides two more functions to achieve deformation compensation and CMRR. In both case robot is still under position control, that is, stiff at all directions. Deformation compensation is achieved by update the target position of position loop based on the measured process force and robot stiffness model, while robot feed speed is adjusted to maintain constant spindle power consumption for CMRR. These two strategies are complementary to each other since CMRR adjusts robot speed at feed direction and deformation compensation adjusts the reference target at rest of directions. The detailed control strategies will be explained in section 5.

IV. EASY ROBOT PROGRAMMING

Programming by Demonstration (PbD) aims at solving the persistent problem of programming robot applications. To be a successful strategy, it must satisfy the requirements for potential robot operators, who usually have the knowledge about the machining process and know the basic operations of robot, such as jogging, write simple robot program.

To facilitate the programming process, an artificially tangible tool (dummy tool) with the same dimensions as the real process tool is usually desirable. For example, in the deburring process with an end milling tool, moving the tool with sharp cutting edge along the workpiece surface can create undesirable friction and damage to the part's surface. However, a cylindrical shape with the same dimension would eliminate the problem and greatly enhance the programming experience.

A. Lead-through

Lead-though is the only step requires human intervention through the entire PbD process. The purpose of lead-through is to generate a few gross guiding points. These guiding points will be used to calculate path frame in path-learning as shown in Figure 3. The position accuracy of these guiding points is not critical because these guiding points are not the actual points/targets in the final program and they will be updated in automatic path-learning. However the orientation of these points should be carefully taught since it will determine the path frame and will be kept in the final program.

Theatrically all six DOFs could be released under force control and the user can adjust both position and orientation of the robot tool at the same time. In practice, we found it is almost impossible to adjust the tool orientation accurately by push/pull with a single hand. Thus, a force control jogging mode is created, under which the operator could push/pull the robot tool to any position easily and change the robot tool orientation using joystick on the teach pendent. Since this jogging is under force control, collision won't happen even when the tool is in contact with the workpiece. As the instant position and orientation of the robot tool is displayed on the teach pendant, the operator could make very accurate adjustment on each independent rotation axis.

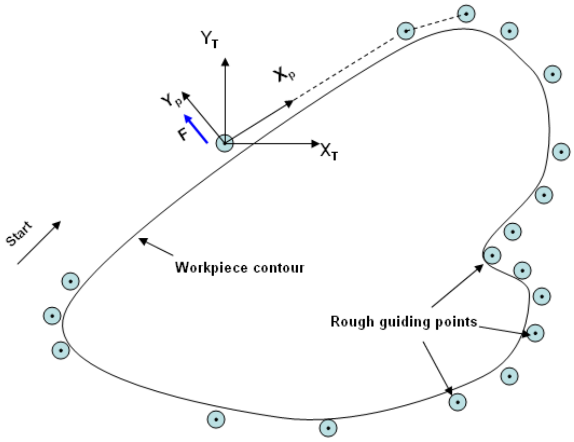

Figure 3. Lead-through and Path learning

B. Automatic Path-learning

A robot program based on gross guiding points taught in lead-through is then generated. This program path, consisted of a group of linear movements from one guiding point to the next, is far different from the actual workpiece contour. The tool fixture would either move into the part or too far away from it.

During the automatic path-learning, the robot controller is engaged in a compliant motion mode, such that only in direction Yp, which is perpendicular to path direction Xp, robot motion is under force control, while all other directions and orientations are still under position control. Further, it can be specified in the controller that a constant contact force in Yp direction (e.g., 20 N) is maintained. Because of this constrain, if the program path is into in the actual workpiece contour, the tool tip will yield along the Y axis until it reaches the equilibrium of 20N, resulting a new point which is physically on the workpiece contour. On the other hand, if the program path is away from the workpiece, the controller would bring the tool tip closer to the workpiece until the equilibrium is reached of 20N.

Since this method uses the path direction of gross guiding points to approximate the actual normal to workpiece contour direction, it is valid only when the normal direction does not change too much between two neighbouring guiding points. As a result, more guiding points need to be taught at sharp corner to limit the approximation error while fewer points are required at the place with small curvature.

While robot holding the tool fixture is moving along the workpiece contour, the actual robot position and orientation are recorded continuously. As described above, the tool tip would always be in continuous contact with the workpiece, resulting a recorded spatial relationship that is the exact replicate between the tool fixture and the workpiece. A robot program generated based on recorded path can be directly used to carry out the actual process. When the robot is executing the actual process, the robot controller is not necessarily to engage any force control behaviour, unless such control would benefit the process in one way or the other.

C. *Post Processing*

After automatic path-learning, the position data logged by the robot controller will be filtered and reduced to generate a robot program. Due to the high dynamic forces, the measurements around sharp corners are often influenced by noise. A threshold for the maximum and minimum acceptable contact force is set up to remove this type of noise. The amount of the targets from automatic path-learning are disproportionately large since the robot controller records the position data as fast as every 4 ms. An approach, namely deviation height method, is used to reduce the redundant points and approximate the contour by straight-line segments. The deviation height limit determined by process requirement is set as the error bound for the reduced robot path. A point will be remained in the path only if there is a certain intermediate point exceeding the deviation height limit. All the intermediate points will be removed from the path. This approach can reduce the length of the point data to 5~10% of the original one. A robot program is then generated in a standard format from the reduced data.

D. *Experimental Results*

With force control integrated in IRC5 controller, PbD method is available for various ABB industrial manipulators. An automatic deburring system using IRB 4400 manipulator is designed to clean the groove of a water pump to guarantee a seamless interface between two pump surfaces.

A 2 mm cutting tool, driven by ultra high speed (~18,000rpm) air spindle is adopted to achieve this task. Since the groove is only about 5 mm wide and has contoured 2D shape, manually teaching a high quality program to clean the complete groove is almost impossible even for very experienced robot operator. Due to the process requirement, the cutting tool is always perpendicular to the surface of water pump. During path-learning, a contact force normal to the edge of 10 N is used, while the velocity is 5mm/s. As shown in Figure 4, the curvature of recorded targets after path learning changes dramatically along the path. The blue points represent the targets in the final cutting program, while the read points represent the offset targets in the test program. The average robot feed speed during the cutting process is about 10 mm/s, while the exact feed speed is determined by the local curvature, which is slower at sharp corner, to ensure a smooth motion throughout the path. The point reduction technique is performed on the filtered measurements. A deviation height of 0.2mm reduced the thousands of points recorded by the robot controller every 40ms to about 300 points.

With this programming strategy, generating a program for a water pump with complex contour, including more than three hundred robot target points, could be completed within one hour instead of several weeks by experienced robot programmer. During this procedure, the operator is only involved in the first step of teaching the gross movement of the robot, while the bulk of the step is automated by the robot controller.

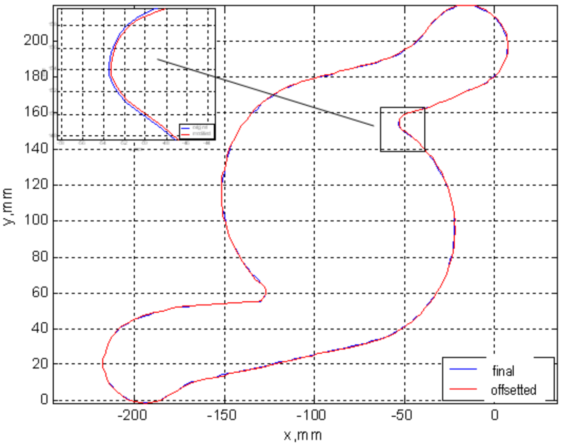

Figure 4. Results from Path-learning

V. PROCESS CONTROL

A. *Robot Deformation Compensation*

The major position error sources in robotic machining process can be classified into two categories, (1) Machining force induced error, and (2) motion error (kinematic and dynamic errors, etc.). The motion error, typically in the range of 0.1 mm, is inherent from the robot position controller and would appear even in non-contact cases. While the machining force in the milling process is typically over several hundreds of Newton, the force-induced error, which could easily go up to 1 mm, is the dominant factor of surface error. Our objective is to estimate and compensate the force induced deformation in realtime to improve the overall machining accuracy.

Since force measurement and subsequent compensation is carried out in 3-D Cartesian space, a stiffness model, which relates the force applied at the robot tool tip to the deformation of the tool tip in Cartesian space, is crucial to realize deformation compensation. For industrial robot the dominant contribution factor for a large deflection of the manipulator tip position is the joint compliance, e.g., due to gear transmission elasticity. Modeling of robot stiffness could be reduced to six rotational stiffness coefficients in the joint space. From the control point of view, this model is also easy to implement, since all industrial robot controllers are decoupled to SISO joint control at the servo level. As a result, the joint deformation could be directly compensated on the joint angle references passed to the servo controller.

For an articulated robot, robot kinematics gives:

$$K_x = J(Q)^{-T} K_q J(Q)^{-1} \tag{1}$$

Where: K_q is a 6×6 diagonal joint stiffness matrix; $J(Q)$ is the Jacobian matrix; K_x is a 6×6 stiffness matrix in Cartesian space. K_x is not a diagonal matrix and it is configuration dependent. If K_q can be measured accurately, the deformation of robot TCP under external force at any location in the workspace could be estimated as,

$$\Delta X = J(Q) K_q^{-1} J(Q)^T \cdot F \tag{2}$$

In this model, the joint stiffness is an overall effect contributed by motor, joint link, and gear reduction units. It is not realistic to identify the stiffness parameter of each joint directly by dissembling the robot; the practical method is to measure it in Cartesian space. While various given payload is applied on the robot tool tip at different robot configurations, the original and deformed positions are measured with ROMER, a portable CMM 3-D digitizer, and the 3-DOF translational deformations are calculated. From, Eq. (2), K_q could be solved by least square method.

The block diagram of real time deformation compensation is shown in Figure 5. After filtering the force sensor noise and compensating the gravity of the spindle and the cutter, the force signal was translated into the robot tool frame. Based on the stiffness model identified before, the deformation due to machining force is calculated in real time and the joint reference for the robot controller is updated accordingly.

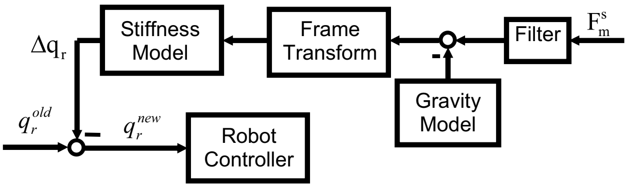

Figure 5. Principle of real-time deformation compensation

B. *Controlled Material Removal Rate*

In pre-machining processes, maximum material removal rates are even more important than precision and surface finish for process efficiency. MRR is a measurement of how fast material is removed from a workpiece; it can be calculated by multiplying the cross-sectional area (width of cut times depth of cut) by the linear feed speed of the tool:

$$MRR = w \cdot d \cdot f \tag{3}$$

Where w is width of cut (mm), d is depth of cut (mm), f is feed speed (mm/s).

Conventionally, feed speed is kept constant in spite of the variation of depth of cut and width of cut. Since most foundry parts have irregular shapes and uneven depth of cut, this will introduce a dramatic change of MRR, which would result in a very conservative selection of machining parameters to avoid tool breakage and spindle stall. The concept of MRR control is to dynamically adjust the feed speed to keep MRR constant during the whole machining process. As a result, a much faster feed speed, instead of a conservative feed speed based on maximal depth of cut and width of cut position, could be adopted.

Since the value of MRR is difficult to measure, the MRR is controlled by regulating the cutting force, which is readily available in realtime from a 6-DOF force sensor fixed on the robot wrist. The challenges for designing a robust controller for MRR is the fact that cutting process model varies to a large degree depending on the cutting conditions. Efforts for designing an adaptive controller will be presented in a separate paper.

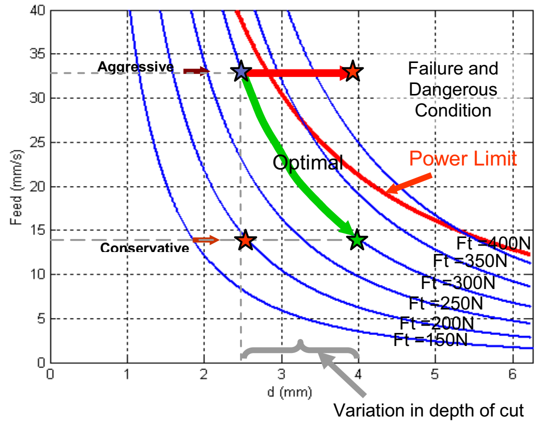

Figure 6. Controlled material removal rate

As the feed speed f is adjusted to regulate the machining force, MRR could be controlled under a specific spindle power limit avoiding tool damage and spindle stall. Also, controlled MRR means predictable tool life, which is very important in manufacturing automation.

The structure of cutting force in a milling operation is represented as linear first-order model:

$$F_c = K \cdot w \cdot d \cdot f \frac{1}{\tau_m s + 1} \tag{4}$$

where τ_m is the machining process time constant. Since one spindle revolution is required to develop a full chip load, τ_m is 63% of the time required for a spindle revolution [8]. The force process gain may be seen as $\theta = K \cdot w \cdot d$, which is sensitive to the process inputs. With the proper selection of reference feed speed f_r and reference force F_r, various controller could be designed to regulate the cutting force F_c, while force process gain θ changes. A simple PI controller is implemented here to demonstrate the effect of CMRR.

C. *Experimental Results*

In the deformation compensation test of milling an aluminum bar, a laser displacement sensor is used to measure the finished surface. The surface error without deformation compensation demonstrates counter-intuitive results; an extra 0.5mm was removed in the middle of the milling path.(Figure 7) Conventional wisdom says that a flexible machine would also cut less material due to deformation, since the normal force during cutting will always push the cutter away from the surface and cause negative surface error. However, in the articulated robot structure, the deformation is also determined by the structure Jacobian, in a lot of cases, a less stiff robot could end up cutting more material than programmed. The coupling of the robot stiffness model explains this phenomenon, the force in feed direction and cutting direction will result in positive surface error in that robot configuration.

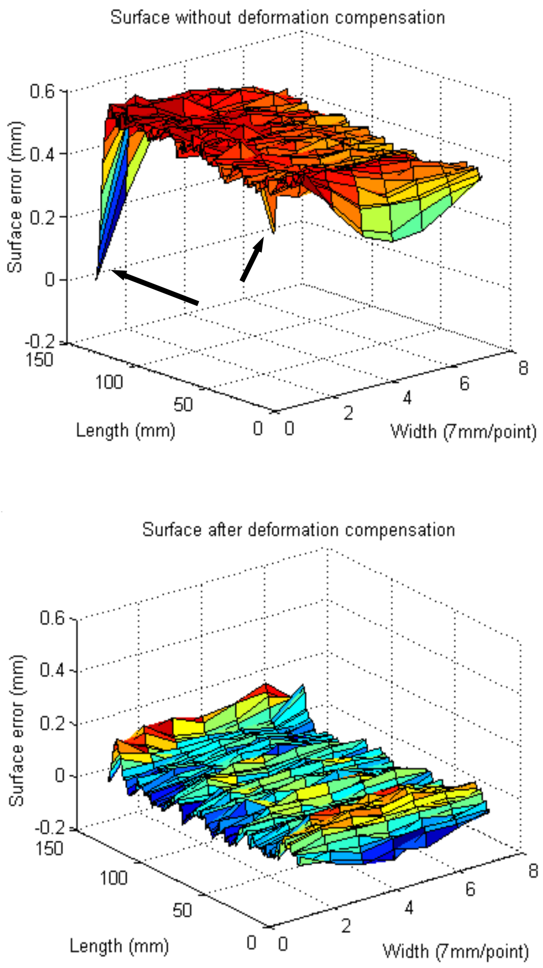

Figure 7. Deformation compensation results

Since the feed force and the cutting force are the major components in this setup, the overall effect will cut the surface 0.5 mm more than the commanded depth. In our definition, negative surface error means less material was removed than the commanded position. The result after deformation compensation shows a less than 0.1 mm surface error, which is in the range of robot path accuracy. Further test conducted on the foundry cylinder head workpiece shows that the surface accuracy improved from 0.9mm to 0.3mm, which is below the 0.5mm target accuracy for pre-machining application.

VI. CONCLUSION

This paper has addressed the critical issues in robotic machining process from programming to process control. Three major contributions, including easy robot programming, online deformation compensation and controlled material removal rate, have been introduced in detail. The complete solution is achieved with force control strategy based on ABB IRC5 robot controller.

Easy robot programming is characterized by two main modules: lead-through and automatic path-learning. Lead-through gives robot operator the freedom to adjust the spatial relationship between the robot tool fixture and the workpiece easily, while robot automatically follow the workpiece contour, record the targets and generate the process program in path-learning. Since the robot programming is generated at actual process setup, no additional calibration is required.

Online deformation compensation is realized based on a robot structure model. Since force induced deformation is the major source of inaccuracy in robotic machining process, the surface quality is improved greatly adopting the proposed method. This function is especially important in milling applications, where cutting force could be as large as 1000 N.

Regulating machining forces provides significant economic benefits by increasing operation productivity and improving part quality. CMRR control the machining force by realtime adjusting the robot feed speed. Various control strategy, including PID, adaptive control and fuzzy logic control, could be implemented depends on different cutting situations

Including the chatter and vibration analysis presented in another paper, these complete set of solutions will greatly benefit the foundry industry with small to medium batch sizes. Dramatic increase of successful setups of industrial robots in foundry cleaning and pre-machining applications will be seen in the very near future.

REFERENCES

[1] Zengxi Pan, "Intelligent robotic machining with force control", Ph.D. dissertation, Stevens Institute of Technology, NJ, USA, Jan 2006.

[2] Luis Basanez, Jan Rosell, "Robotic Polishing Systems—From graphical task specification to automatic programming", IEEE Robotics & Automation magazine, Sep, 2005.

[3] J. Norberto Pires, et, al, "CAD interface for automatic robot welding programming", Industrial Robot: An International Journal, Vol 31, pp.71-76, 2004.

[4] Zengxi Pan, Hui zhang, et, al, "Chatter analysis of robotic machining process", Journal of Material Processing Technology, Volume 173, Issue 3, Pages 301-309, April 2006.

[5] Sung I. Kim, Robert G. Landers, A. Galip Ulsoy, 2003, "Robust Machining Force Control with Process Compensation," Journal of Manufacturing science and engineering, Vol 125, pp. 423-430

[6] Seung-Han Yang, 1996, "Real-time compensation for geometric, thermal, and cutting force induced errors in machine tools," Ph.D. dissertation, The University of Michigan

[7] E. Budak, Y. Altintas, 1998, "Analytical Prediction of Chatter Stability Conditions for Multi-Degree of Systems in Milling. Part I: Modeling, Part II: Applications," Transactions of ASME, Journal of Dynamic Systems, Measurement and Control, vol.120, pp.22-36

[8] L.K. Daneshmend, H.A. Pak, 1986, "Model Reference Adaptive Control of Feed Force in Turning," ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 108, No. 3, pp. 215-222