CAD DIRECTED ROBOTIC DEBURRING

K. M. Murphy, R. J. Norcross, and F. M. Proctor
Robot Systems Division, National Bureau of Standards

ABSTRACT

Robots prove advantageous for the deburring of machined parts because they are more repeatable than humans and suffer no fatigue. However, most deburring robots are programmed manually, requiring a large investment in programming time. At the National Bureau of Standards' Automated Manufacturing Research Facility, research is being conducted on techniques to automate robot programming. A technique has been developed and demonstrated in the Cleaning and Deburring Workstation which uses CAD geometry data to automatically generate robot deburring paths. Using a graphics interface, an operator specifies the edges on a part to be deburred, the deburring tools to be used, and the speeds, feed rates, and contact forces desired. Deburring paths are generated and sent to a PUMA 760 robot controlled by the NBS Real-Time Control System. The robot uses a two-pass technique for deburring. On the first pass, the robot uses force feedback to correct the deburring path points to account for robot kinematic errors, tool wear, and minor part misplacement. On the second pass, the robot follows the corrected path, deburring the part. This paper describes the techniques, algorithms and data formats used in this robotic deburring system.

INTRODUCTION

Deburring is the removal of rough or thin ridges formed during a machining process. Typically done manually, deburring is accomplished by applying tools such as brushes or files to the edges of the part to be deburred. Manufacturers are continually plagued by high job turnover, broad variation in quality, and low employee esteem associated with the tedious process of manually deburring job-shop quantities of machined parts. Consistent with this is the observation that the cost of manual deburring can reach 30% of the total production cost.

Robotic deburring has slowly found its way into commercial applications involving the production of large quantities of single part types, and where there exist stringent requirements for reproducible results. The ability of robots to work continuously suits them to the former task, while the precision to which they can retrace recorded motions suits them to the latter. Unfortunately, most businesses find it impossible to justify the recurring costs of programming the robot to deburr new parts, an impasse that has hampered the proliferation of robots in the workplace. The National Bureau of Standards has chosen to tackle this problem by developing methods to automatically generate, from geometry data, the sequence of robot motions required for deburring.

This work was partially funded by the U. S. Navy MANTECH Program.

The Automated Manufacturing Research Facility (AMRF) is a research testbed consisting of three machining workstations, an inspection workstation, a distributed material handling workstation, and a cleaning and deburring workstation (CDWS) [1]. The CDWS contains two cooperating robots with a rotary vise for part fixturing placed in their common volume. A Unimate 2000 six-axis hydraulic robot is equipped with a gripper for part handling. A PUMA 760 six-axis electric robot is fitted with a quick change which allows for selection between a chamfering tool, an end brush, and a hole brush. The abrasive-loaded nylon brush material limits the deburring to soft metals such as aluminum and brass. Harder materials require carbide tools, rotary files, or grinding stones.

One of the principal advances of the workstation is a system that generates accurate deburring paths from CAD data. The process is shown in figure 1. A graphic representation of a part is generated from geometry data and presented to a user who selects the edges to be deburred and the desired deburring parameters. A robot path planner generates an initial deburring path for the robot. This path is not suitable for deburring due mainly to robot inaccuracy and is automatically corrected by the robot using force feedback.

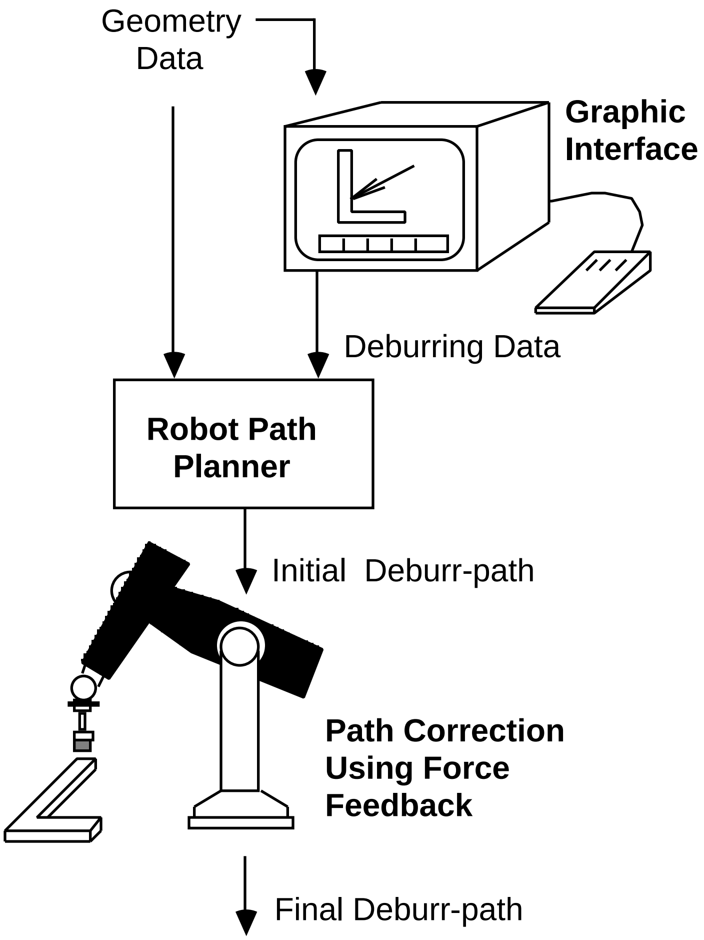


Figure 1. CAD Directed Robotic Deburring. An operator specifies how edges are to be deburred. A path planner generates an initial path which is corrected by the robot to account for inaccuracies.

Three types of data define and limit the automatic generation of robot paths. Geometry data provides a description of the workpiece and is used by graphic interface routines to develop deburring and fixturing data. Deburring data indicates which deburring techniques are to be used with selected edges. Fixturing data provides coordinate transformations between the workpiece, vise, and robot frames.

The robot path planner uses the geometry, deburring and fixturing data to develop deburring paths. This process occurs in three steps: related edges are grouped, robot poses (the positions and orientations of the tool) for each edge are generated, and the robot poses are formatted to form a deburr-path.

Unfortunately, these paths are not suitable for deburring due to several sources of inaccuracy. The inaccuracy of the robot contributes most significantly. Although most robots have high *repeatability*, the ability to return to the same location when given the same coordinates, the precision with which a computed point can be attained, the *accuracy*, is low. Errors in machining, large part tolerances, and tool wear also contribute to discrepancies between the real world and the software models. To compensate for inaccuracy, the robot corrects the computed path by approaching each endpoint half an inch back along the tool axis, driving its tool forward while reading a wrist force sensor until a desired contact force is met. The coordinates of this point replace the computed coordinates. When the entire set of endpoints on the paths have been corrected for robot inaccuracy, the paths are traced with the selected tool to accomplish deburring.

This paper presents the data structure, path planning, and correction routine used in the implementation of CAD Directed Robotic Deburring.

DATA

The robot path planner requires geometry, deburring, and fixturing data. Geometry data is derivable from the AMRF's part model database report [2], or may be manually entered. Deburring data associates edges with deburring techniques. Fixturing data provides coordinate transformations between the workpiece and the part handling gripper and vise. Both the deburring data and the fixturing data are created by an operator using an interface that presents graphic representations of the part and fixtures from geometry data. This interface allows the user to specify which deburring techniques are to be used on selected edges and how a part is to be gripped by the robot and vise. The user manipulates a drawing of the workpiece, a collection of graphic buttons, and text windows to create and modify the data. Creating the geometry, deburring, and fixturing data for a simple workpiece takes an average of one to two hours.

The geometry data used locally is a small subset of the part model data available throughout the AMRF. The geometry data includes the topology (the relationship between points, edges, faces and shells) and geometry (the location of vertices, curves, and surfaces). Edges are defined as the intersection of surfaces. Although several surface types are supported by the AMRF part model, the current implementation at the workstation recognizes only two types: PLANEs and CYLINDERs. The intersection of these two surface types represent 90% of the edges on workpieces handled at the workstation. Other edges can be approximated. A PLANE is defined by the surface normal, oriented out of the material, and the distance of the plane from the origin. The surface normal is represented as a unit vector and the distance from the origin is the dot product of the surface normal and a vector from the origin to any point on the plane. A CYLINDER is defined by a point on the cylinder's center axis, the orientation of the axis, the radius of the cylinder, and a flag that specifies whether the cylinder is a hole (material removed) or a rod (material remaining).

The deburring data consists of an unordered list of edges associated with deburring techniques. The edge selections are stored as pointers into the geometry data. The deburring techniques specify tools and their parameters. The current deburring tools include an end brush, a hole brush, and a chamfering tool. The deburring parameters that can be specified include feed rate, tool speed, interference force, and tool orientation to the surface. Each edge also contains a pointer into the fixturing data which specifies the part orientation when the edge was selected.

The fixturing data contains the transformation of the origin of the workpiece relative to the vise, as well as the expected vise opening. This data is used in conjunction with the deburring data to ensure that the edges to be deburred are accessible and to generate approach and depart poses for the deburring tool. The fixturing data is also used to generate paths for the part handling robot, following methods similar to those discussed here.

PATH PLANNING

The robot path planner is a software module that automatically generates *deburr-paths* from the geometry, deburring, and fixturing data. A deburr-path is a set of edges which can be deburred without refixturing the part or changing the deburring technique. When formatted for the robot controller, a deburr-path consists of a header, which specifies the path name, part location, and deburring technique, and a list of path points that give step-by-step instructions for deburring the edges of the part. Generating the deburr-paths requires three steps: edge organization, pose generation, and robot interface formatting.

Edge Organization

The deburring data developed by the graphic interface is an unordered list of edges. Although these edges may be deburred in this initial random order, grouping them according to part orientation and deburring technique reduces processing time by eliminating unnecessary tool changes and approach and depart trajectories.

Edges with the same part fixturing requirements, deburring tool, and roughly equivalent deburring parameters are grouped by the path planner into deburr-paths. The path planner then divides each deburr-path into one or more *loops*. A loop is a set of edges that can be deburred without lifting the tool from the part, and are built by sequentially linking edges with similar endpoints. The similarity requirement depends on the deburring tool. For example, the end brush has a half-inch radius on its working surface. Due to this large surface, endpoints which fall within 0.15 inch of each other are considered similar and their edges can be placed in the same loop. In contrast, a chamfering tool cannot deburr sequential edges, and its loops always consist of a single edge.

Pose Generation

Once edges are organized, the path planner creates a sequence of poses which define the trajectory for each deburr-path. The poses contain position and orientation information in the part's coordinate frame. During deburring, the robot will move between these poses in straight-line motion. There are two types of poses in a deburr-path: *vertex* poses, at which the tool makes contact with the part, and *goto* poses, used for approach and depart trajectories.

In the conversion of edges to poses, the path planner considers each edge separately. For each edge type, and for each tool, there is a routine for converting an edge into a sequence of poses. The path planner supports two surface types and three deburring tools. These form nine edge/tool combinations and require nine conversion functions. However, since the chamfering tool and hole brush are used only on edges defined by the intersection of planes and cylinders (material removed), the number of required functions is reduced.

Edges defined by the intersection of planes with planes, or planes with cylinders (material remaining) are deburred with the end brush. In the former case, the robot traces the edge along a straight line connecting the endpoints. The trajectory for this edge consists of two vertex poses located at the endpoints. In the latter case, the trajectory consists of many intermediate vertex poses located along the arc, which is approximated by several chords. The tool orientation is offset from the normal of the first machined surface, as shown in figure 2. This concentrates the effect of the tool on the burr. The offset is specified by the operator as a deburring parameter.

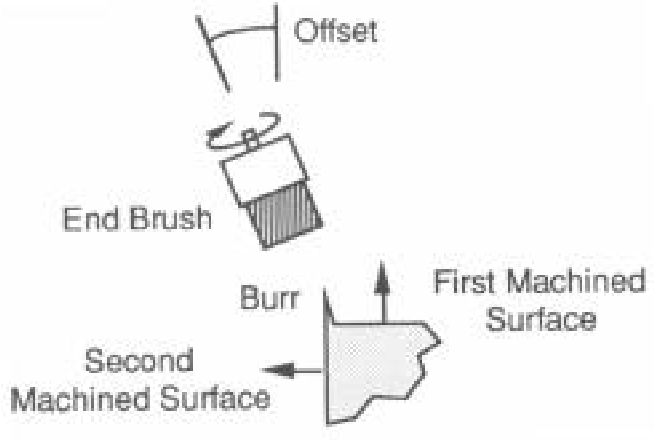


Figure 2. Tool Orientation. A burr occurs at the intersection of two machined surfaces, typically in the plane of the second. To effectively remove the burr, the orientation of the tool is offset from the normal of the first machined surface.

In addition to the vertex poses on the workpiece, the robot requires safe approach and depart trajectories. The goto poses provide these trajectories. The robot begins three inches above the first vertex pose of each loop, proceeds vertically for two-and-a-half inches, then approaches the workpiece in the same orientation as the vertex pose. The robot always approaches from above the workpiece, insuring an unblocked trajectory. The depart trajectory is determined similarly for the last vertex pose of each loop.

Robot Interface Formatting

The generated poses must be placed in a format suitable for the deburring robot controller. The PUMA 760 robot is controlled by the NBS-developed Real-Time Control System (RCS), which consists of a database and five hierarchical control levels [3]. The RCS database requires that deburr-paths begin with path header information (-path-), followed by a list of path points (-ppt-). The general format for a deburr-path is:

```
   -path-  deburr  part name, path number,             ; path identifier
                   part location,                      ; location in fixture
                   tool type, speed, force, feed rate  ; deburring technique

   -ppt-   vise    open/closed, rotation position      ; set vise location
   -ppt-   goto    approach pose                       ; start above first edge
   -ppt-   tool-on                                     ; turn tool on
   -ppt-   vertex  pose                                ; locations in part frame
   -ppt-   vertex  pose
             .       .
             .       .
             .       .
   -ppt-   vertex  pose
   -ppt-   goto    depart pose                         ; back away from last edge
   -ppt-   tool-off                                    ; turn tool off
```

The header specifies the deburr-path label, part location, and deburring technique, while the list of path points gives step-by-step instructions for deburring. There are several types of path points. A *vertex* path point specifies a vertex pose which positions the tool on an edge. During deburring, the robot moves from one vertex pose to the next in straight-line motion according to the trajectory parameters given in the header. These vertex poses must be corrected to account for the robot's inaccuracy. A *goto* path point specifies an approach or depart pose. These poses do not need to be corrected since they occur in free space where robot accuracy is not critical. A *vise* path point indicates the vise position, while *tool-on* and *tool-off* path points actuate the tool.

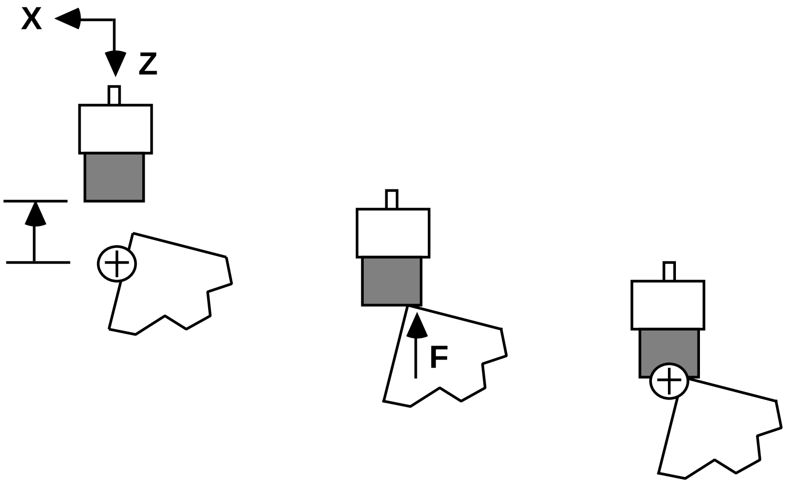
PATH CORRECTION

Most industrial robots are poorly suited for executing trajectories computed offline. A typical robot's accuracy (the ability to go to a computed position in space) is considerably worse than its repeatability (the ability to return to a previously recorded position). The deburring robot's repeatability is 0.005 inch, while errors in accuracy of 0.25 inch may occur for motions about the part. Calibration can increase the robot's accuracy; however, system inaccuracies such as part misplacement remain and require correction. Alternatively, the deburring process can be made significantly forgiving, or the paths can be adjusted online. At the workstation, a combination of soft tooling (abrasive-loaded brushes) and path correction accommodate errors for the end brush.

After a path is generated and stored in the robot's database, the workstation controller commands the robot to correct the path. During this procedure, the value of the vertex poses are changed to a new value, as shown in figure 3. The robot moves to a position half-an-inch back along the tool-Z axis from the unaltered vertex pose (i). The wrist force sensor is nulled to account for gravity, and the robot moves toward the part along the Z axis until it contacts the part and develops a commanded force (ii). The robot stores the corrected position in place of the old vertex pose (iii) and backs away from the part.

This method only corrects errors in the tool-Z direction. For the case of the end brush, the area of the brush face compensates for errors in the X and Y directions, which are much smaller than the brush radius. Similarly, the chamfering tool and hole brush are compliant in the XY plane. When the radius of the hole is larger than the errors in the X and Y axes, these tools become self-centering.

Modifications of the force-sensing technique for use with holes would further increase the performance of hole deburring.



(i) (ii) (iii)

Figure 3. Path Correction. The endpoint calculated by the path planner is not correct due to system inaccuracies. To correct the point, the robot starts back from the initial point (i), moves in until the commanded force is obtained (ii), and records the new point (iii). When each vertex pose in the deburr-path is corrected, the path is used to deburr all parts of that type. To account for tool wear and part variation, the path is corrected again after a period determined by the workstation controller.

SUMMARY

A robotic deburring workstation provides improved flexibility and repeatability over other deburring alternatives. However, when used on a wide variety of parts, each produced in small numbers, these advantages are offset by the labor required to accurately program the robot for each part. Researchers at the Cleaning and Deburring Workstation are developing techniques which significantly reduce the manual programming requirements of robotic deburring. A graphic interface allows an operator to quickly associate edges with the deburring tools, speeds, feed rates, and contact forces used to deburr those edges. The resulting deburring data is combined with geometry and fixturing data to automatically generate robot deburring paths. A PUMA 760 robot, controlled by NBS's Real-Time Control System, then uses a two-pass technique for deburring. On the first pass, the robot uses force feedback to correct the automatically generated path points for tool wear, minor part misplacement, and the robot's kinematic errors. On the second pass, the robot follows the corrected path, deburring the part. This approach has proven effective for deburring aluminum and brass parts with abrasive brushes.

REFERENCES

1. J. Simpson, R. Hocken, and J. Albus, "The Automated Manufacturing Research Facility of the National Bureau of Standards", Journal of Manufacturing Systems 1 (1) (1983).

2. T. Hopp, "AMRF Database Report Format: Part Model", National Bureau of Standards Internal Report 87-3672 (September 30, 1987)
3. A. Barbera, J. Albus, M. Fitzgerald, and L. Haynes, "RCS: The NBS Real-Time Control System", Robots 8 Conference and Exposition, Detroit, MI (June 1984).