ABSTRACT

BREHL, DAVID EDWARD. 3-D Microstructure Creation Using Elliptical Vibration-Assisted Machining (EVAM). (Under the direction of Thomas A. Dow, Ph.D.)

In elliptical vibration-assisted machining (EVAM), a diamond cutting tool is made to move in a micrometer-scale elliptical path at high frequency. This oscillation is superimposed on the normal feed motion of the tool. Compared to diamond turning and spindle-type micromachining processes, EVAM has several advantages for making millimeter-scale and smaller structures. These include smaller machining forces, reduced vibration, elimination of runout error, and extended diamond life when cutting ferrous or brittle materials. To date EVAM has been used mostly to make binary and low aspect ratio / low relief features with vertical dimensions of only a few micrometers. The current research explored EVAM's ability to make 3-D microstructures with geometry typical of micro-optical, micro-fluidic, and MEMS devices.

The Ultramill EVAM tool previously developed at North Carolina State University was employed in this work. A new hydrostatic oil-bearing Y-axis was installed on the Nanoform diamond turning machine which improved surface finish by increasing stiffness in the depth of cut direction. To suppress thermal upsets which could cause form error, the Ultramill's original gravity-driven cooling system was replaced with a closed-circulation arrangement. At the small feed velocities used for microstructure fabrication, surface roughness after the cooling system conversion was found to be caused principally by coolant pump pulsations which partially offset the improvement gained from the stiffer Y-axis.

Existing 2-D kinematic and theoretical surface roughness relationships for EVAM were expanded to the general case of 3-D surfaces. Commercial CNC motion planning software could not calculate the complicated cutter compensation required by EVAM's elliptical tool path, so an innovative method was developed for motion program generation, based on surface morphology methods used in image processing and contact probe microscopy.
A variety of 3-D parts were raster-cut with the Ultramill. Round-nosed diamond tools were used to make features with convex, concave, tilted planar and sculpted 3-D geometry. Positive features had heights up to 20 μm, and negative features depths as great as 130 μm. Height-to-width aspect ratios of 0.15 were achieved, compared to 0.01 for most previous parts. EVAM's ability to make functional components was demonstrated by fabrication of millimeter-scale reflecting optical surfaces. These included concave spherical elements with a form error of 62 nm RMS. A complex off-axis segment of an ellipsoidal surface was also made, as might be used in a free-space fiber optic beam splitter. This part had estimated centerline form error of ~480 nm RMS limiting its functional performance. A significant cause of this form error was found to be axis squareness error, rather than being assignable to the EVAM process.

Sharp-nose and dead-sharp diamond tools were used to machine channels and pockets with steep straight sidewalls. Such tool geometries have not been previously discussed in the EVAM literature. Tetrahedron arrays from 5 μm to 80 μm tall were also cut with a dead-sharp tool, in copper and stainless steel. Severe burr appeared in stainless steel parts, unlike when a round-nosed tool was used to machine the same material. Tool cutting edge wear was also observed after machining stainless steel for a 2 m raster distance with the sharp nose tool, while negligible wear was noted at this distance for a round-nosed tool.
3-D Microstructure Creation Using Elliptical Vibration-Assisted Machining (EVAM)

by

David Edward Brehl

A dissertation submitted to the graduate faculty of North Carolina State University in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

Mechanical Engineering

Raleigh, NC

2014

APPROVED BY:

____________________________  ______________________________
Gregory A. Buckner           Ronald O. Scattergood

____________________________  ______________________________
Thomas A. Dow                Paul I. Ro
Chair of Advisory Committee
BIOGRAPHY

Dave Brehl grew up outside the city of Pittsburgh and considers it and the state of Pennsylvania to be his home, though he has not lived there for some years. He earned a B.S. in Mechanical Engineering from the University of Notre Dame (South Bend, IN) in 1981, and a masters’ in mechanical engineering from Stevens Institute of Technology (Hoboken, NJ) in 2003. Prior to coming to North Carolina State University to work on his doctorate, Dave gained approximately 20 years’ professional experience in a broad range of development, manufacturing, and project engineering roles. His employers during this time included Babcock and Wilcox (Barberton OH), BOC Gases (Murray Hill, NJ) and AERCO International (Northvale, NJ). His professional focus in this part of his career was on thermal/fluid process equipment, and included such product lines as fluidized-bed combustion systems, air separation equipment, and high-efficiency commercial water heaters. A desire to move into advanced technologies and manufacturing processes, and to work on something where the accuracy of third digit after the decimal point actually mattered, led him to the Precision Engineering Center at NC State University in 2004. He looks forward to applying the principles of precision engineering and the skills learned at NCSU throughout the second half of his career. He currently lives in Santa Barbara, CA, and runs a diamond turning group for Raytheon Vision Systems. He looks forward to continuing his career in a development role, and maintains a dream of taking a product all the way from the napkin sketch thru to shipping the first production unit.
ACKNOWLEDGEMENTS

I would like to thank everyone who has given me guidance and instruction during my time at NC State University. The whole team at the Precision Engineering Center provided invaluable help, and I greatly enjoyed working in a research environment with such excellent facilities and exposure to such a wide variety of projects and disciplines. The following people deserve a personal thank-you:

- Dr. Thomas Dow, my advisor, for giving me the chance to learn so much at the PEC,
- Drs. Greg Buckner, Ron Scattergood, and Paul Ro, my committee members.
- Alex Sohn and Ken Garrard, Research Engineers at the PEC, for all their help, knowledge, and support.
- Dr. Dow, Alex, Matthew Cerniway, Nobu Negishi, and Brett Brocato for the work on the Ultramill 1999-2005. Standing on the shoulders of giants etc.
- Brandon Lane, for keeping me updated on his work with EVAM after I left the PEC in 2008.
- Karl Falter, PEC MS ’89, for his expertise with the Nanoform and for performing the Moore Y-axis integration.
- Dr. Dow once more, for letting me have the experience of co-organizing the 2007 spring ASPE special topic meeting on vibration-assisted machining.
- Ken Garrard, again, for his early suggestion of using morphology methods in motion-program planning, and for MATLAB routines for analyzing Zygo New View data.
- Dr. John Ellis-Monaghan of IBM (PhD NCSU 1995) for the initial suggestion in 2003 to go back to grad school. I think I am grateful for this.
- Robin Fogel, Kathleen Flynn MD, Janet Hachen, George Rufener, and Gary Bennett, for making me finish it.
• Steve Brehl my brother, for dragging me back to ND at a critical time
• Steve Brehl my cousin, for his general support and interest.
• Marianne Brehl, for the timely reminder “finished is maybe more important than good”

This project was sponsored by the National Science Foundation, under NSF contract # DMI-0433215, Monitor: George Hazelrigg.

Additional financial support was received from Sandia National Laboratories, Oak Ridge National Laboratory, and Panasonic Boston Laboratory.

Chardon Tool (Chardon, OH) provided diamond tools.
# TABLE OF CONTENTS

LIST OF TABLES ................................................................................................................................. xi  

LIST OF FIGURES ................................................................................................................................. xii

1 INTRODUCTION ................................................................................................................................. 1

1.1 OVERVIEW ......................................................................................................................................... 1

1.2 MICROSTRUCTURE CREATION USING MECHANICAL MICROMACHINING .............................. 2

1.2.1 Mechanical Micromachining ......................................................................................................... 2

1.2.2 Mechanical Micromachining Issues ............................................................................................. 4

1.3 VIBRATION-ASSISTED MACHINING ................................................................................................. 5

1.4 REVIEW OF MICROSTRUCTURE FABRICATION USING VAM .................................................. 11

1.4.1 VAM Microstructure Examples .................................................................................................. 12

1.4.2 EVAM Experience at the Precision Engineering Center ............................................................... 14

1.5 OBJECTIVES OF THE CURRENT RESEARCH .............................................................................. 16

2 3-D EVAM KINEMATICS AND THEORETICAL SURFACE ROUGHNESS ... 18

2.1 COORDINATE SYSTEM .................................................................................................................... 18

2.2 EQUATIONS OF MOTION ................................................................................................................ 19

2.3 EVAM CYCLE PARAMETERS ........................................................................................................ 27

2.4 THEORETICAL EVAM SURFACE ROUGHNESS ............................................................................ 28

3 EQUIPMENT DESCRIPTION and IMPROVEMENTS ................................................................. 32

3.1 ULTRAMIL-NANOFORM SYSTEM ................................................................................................. 32

3.2 ULTRAMIL EVAM TOOL ................................................................................................................ 33

3.3 DIAMOND TOOLS AND TOOLHOLDER ........................................................................................ 37
3.4 NANOFORM DIAMOND TURNING MACHINE

3.4.1 Description

3.4.2 Y-Axis Upgrade

3.4.3 Video Zoom Cameras

4 ULTRAMILL THERMAL STABILITY IMPROVEMENT

4.1 INTRODUCTION

4.1.1 Ultramill Temperature Variation

4.1.2 Heat Generation by Piezoelectric Actuators

4.2 OPEN- and CLOSED-CIRCULATION COOLING

4.3 FORM ERROR IMPROVEMENT WITH CLOSED-CIRCULATION COOLING

4.4 EXPERIMENT SETUP

4.5 ULTRAMILL COOLING SYSTEM EXPERIMENTS

4.5.1 Overview

4.5.2 Coolant System Flow Rate

4.5.3 Energy Balance Test

4.5.4 Temperature Tests

4.5.5 Long-Duration Temperature Stability

4.5.6 Capacitance Gauge Tests

4.5.7 Machined Flats

4.6 SUMMARY of ULTRAMILL COOLING SYSTEM EXPERIMENTS

5 SINGLE-GROOVE MACHINING EXPERIMENTS

5.1 PREVIOUS ULTRAMILL SURFACE ROUGHNESS RESULTS

5.2 SINGLE-GROOVE CUTTING EXPERIMENTS
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3</td>
<td>FLYING WING</td>
<td>7.3.1 Part Description, Motion Program, and Machining</td>
<td>121</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.3.2 Metrology Results</td>
<td>122</td>
</tr>
<tr>
<td>7.4</td>
<td>SCULPTED 3-D GEOMETRY (&quot;POTATO CHIP&quot; PART)</td>
<td>7.4.1 Part Description, Motion Program, and Machining</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.4.2 Metrology Results</td>
<td>130</td>
</tr>
<tr>
<td>8</td>
<td>OPTICAL SURFACE FABRICATION: CONCAVE SPHERICAL REFLECTOR</td>
<td>8.1 PART DESIGN</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.2 PART FABRICATION</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.2.1 Motion Program and Machining Strategy</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.2.2 Machining Operations</td>
<td>135</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.3 METROLOGY RESULTS</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.3.1 Form Error</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.3.2 Surface Roughness</td>
<td>142</td>
</tr>
<tr>
<td>9</td>
<td>OPTICAL SURFACE FABRICATION: FIBER OPTIC BEAM BENDER</td>
<td>9.1 INTRODUCTION</td>
<td>147</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.1.1 Beamsplitter Concepts</td>
<td>147</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.2 DEMONSTRATION REFLECTOR</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.2.1 Beam Bender Concept and Test Setup</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.2.2 Optical Fiber Specification</td>
<td>151</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.2.3 Optical Surface Design</td>
<td>152</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.3 REFLECTOR FABRICATION</td>
<td>154</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.3.1 Machining Setup and Motion Program</td>
<td>154</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.3.2 Machining Operations</td>
<td>157</td>
</tr>
</tbody>
</table>
# Table of Contents

9.4 PERFORMANCE TEST .................................................................................................................. 159
9.5 METROLOGY RESULTSS ............................................................................................................. 163
  9.5.1 Form Error.......................................................................................................................... 163
  9.5.2 Surface Roughness .............................................................................................................. 168
9.6 ERROR ANALYSIS of BEAM BENDER REFLECTOR........................................................... 169

10 3-D MICROSTRUCTURES MADE USING SHARP-NOSE TOOLS.............. 173
10.1 INTRODUCTION ....................................................................................................................... 173
  10.1.1 Limitations of Round-Nosed Tools ..................................................................................... 173
  10.1.2 Sharp-Nose Tools .............................................................................................................. 174
  10.1.3 Sharp-Nose Tool Cutting Edge Velocity ............................................................................ 176
10.2 TETRAHEDRON ARRAYS ......................................................................................................... 178
  10.2.1 Part Design and Fabrication ............................................................................................. 178
  10.2.2 Tetrahedron Results – SEM Imagery of Copper Parts ....................................................... 182
  10.2.3 Tetrahedron Results – Burr Formation ............................................................................ 184
  10.2.4 Tetrahedron Results – Tool Wear .................................................................................... 187
10.3 ADDITIONAL MACHINING EXPERIMENTS WITH SHARP-NOSE TOOLS ................. 190
  10.3.1 Grooves Made with 40º Nose-Angle Tool ......................................................................... 190
  10.3.2 Microchannel Stream Combiner ...................................................................................... 191

11 CONCLUSIONS and RECOMMENDATIONS FOR FUTURE WORK ........... 196
  11.1 CONCLUSIONS ....................................................................................................................... 196
  11.2 RECOMMENDATIONS FOR FUTURE WORK ........................................................................ 203

12 REFERENCES ............................................................................................................................... 206
13 APPENDICES ............................................................................................................. 212

13.1 APPENDIX A: MODELING OF EVAM SURFACE FEATURES ON A SLOPED SURFACE ............................................................................................................. 213

13.2 APPENDIX B: MACHINING PROCEDURE FOR ULTRAMILL ......................................................... 225

13.3 APPENDIX C: ULTRAMILL ELLIPTICAL TOOLPATH VERIFICATION ........................................ 229

13.4 APPENDIX D: ULTRAMILL VIDEO MICROSCOPE CAMERAS ................................................. 232

13.5 APPENDIX E: TEMPERATURE MEASUREMENT FOR ULTRAMILL THERMAL EXPERIMENTS ............................................................................................................. 245

13.6 APPENDIX F: LABVIEW BLOCK DIAGRAM FOR ULTRAMILL THERMAL EXPERIMENTS ............................................................................................................. 248

13.7 APPENDIX G: ULTRAMILL ENERGY BALANCE ........................................................................ 251

13.8 APPENDIX H: THEORETICAL INVESTIGATION of THE MINIMUM CHIP THICKNESS EFFECT IN EVAM ............................................................................................................. 258

13.9 APPENDIX I: PROGRAM LISTING FOR UPFEED KNIFE-EDGE PART ................................. 276

13.10 APPENDIX J: PROGRAM LISTING FOR FLYING WING PART .................................................. 278

13.11 APPENDIX K: PROGRAM LISTING FOR “POTATO CHIP PART” (DOUBLE-Sinusoidal Grooves) ............................................................................................................. 287

13.12 APPENDIX L: PROGRAM LISTING FOR SPHERICAL REFLECTOR ........................................ 294

13.13 APPENDIX M: DERIVATION of ELLIPSOID REFLECTOR PROFILE ....................................... 306

13.14 APPENDIX N: ELLIPSOID ROTATION TO MACHINING ORIENTATION ............................... 311

13.15 APPENDIX O: BEAM BENDER REFLECTOR MOTION CONTROL PROGRAM ............... 314

13.16 APPENDIX P: TETRHEDRON TOOL ANGLE DERIVATION ..................................................... 328

13.17 APPENDIX Q: CENTERING ULTRAMILL TO SPINDLE AXIS .............................................. 332
LIST OF TABLES

TABLE 3-1. ULTRAMIL-NANOFORM SYSTEM CONFIGURATIONS…………………………………………33
TABLE 3-2. PIEZOELECTRIC STACK ACTUATOR CHARACTERISTICS………………………………36
TABLE 3-3. UPFEED SURFACE FINISH COMPARISON BETWEEN AIR-BEARING AND
HYDROSTATIC OIL-BEARING Y-AXIS……………………………………………………………………42
TABLE 4-1. ESTIMATED PIEZO STACK THERMAL OUTPUT USING EQUATION 4-1 AND K=0.24
(TOTAL FOR 2 STACKS)…………………………………………………………………………………………48
TABLE 4-2. EFFECT OF COOLANT FLOW RATE ON ULTRAMIL TEMPERATURE
OSCILLATIONS……………………………………………………………………………………………………58
TABLE 4-3. TEMPERATURE AND TARGET POSITION VARIATION FOR CAP GAUGE TESTS………62
TABLE 4-4. FIGURE ERROR AND SURFACE ROUGHNESS FOR FLATS MADE USING CLOSED-
CIRCULATION COOLING……………………………………………………………………………………………65
TABLE 5-1. MACHINING CONDITIONS AND MATERIAL PROPERTIES FOR GROOVE PARTS
MANUFACTURED USING OPEN-CIRCULATION COOLING……………………………………………………71
TABLE 6-1. ERROR CAUSED BY STRUCTURING ELEMENT DISTORTION ON SLOPED
SURFACE………………………………………………………………………………………………………………112
TABLE 7-1. SURFACE ROUGHNESS OF SHALLOW SLOPES ON “FLYING WING” PART………127
TABLE 8-1. MACHINING CONDITIONS FOR CONCAVE SPHERICAL REFLECTOR………………134
TABLE 8-2. AREA-BASED SURFACE ROUGHNESS FOR SPHERICAL REFLECTOR SURFACE MADE
IN HARD-PLATED COPPER…………………………………………………………………………………………145
TABLE 8-3. AREA-BASED SURFACE ROUGHNESS FOR SPHERICAL REFLECTOR SURFACE MADE
IN ALUMINUM…………………………………………………………………………………………………………145
TABLE 9-1. MACHINING CONDITIONS FOR BEAM-BENDER REFLECTOR………………156
TABLE 9-2. RASTER PASSES, CROSSFEED SPACING, AND DEPTH OF CUT FOR REFLECTOR
SURFACE………………………………………………………………………………………………………………157
TABLE 9-3. BEAM BENDER PERFORMANCE…………………………………………………………161
TABLE 9-4. AREA-BASED SURFACE ROUGHNESS FOR ELLIPSOIDAL REFLECTOR SURFACE……161
TABLE 9-5. PREDICTED DEVIATION FROM DESIGN SURFACE CAUSED BY NANOFORM
SQUARENESS AND MOTION PROGRAM ERRORS…………………………………………………………171
TABLE 10-1. SCHEMES FOR GROOVE CUTS FOR THE TETRAHEDRON ARRAYS………………181
TABLE 10-2. PROPERTIES OF MATERIALS USED FOR TETRAHEDRON FABRICATION
EXPERIMENTS……………………………………………………………………………………………………….185
TABLE D-1. Y-AXIS CAMERA PERFORMANCE……………………………………………………………………209
LIST OF FIGURES

FIGURE 1-1. 2 TYPES OF VIBRATION-ASSISTED MACHINING. LEFT 1-D VAM RIGHT ELLIPTICAL VIBRATION-ASSISTED MACHINING (EVAM)..........................................................................................6

FIGURE 1-2. COMPARISON OF 1-D VAM AND EVAM WHEN CUTTING A DOWNWARD TILTED SURFACE........................................................................................................................................8

FIGURE 1-3. INSTANTANEOUS TOOL FORCE MEASUREMENTS FOR VAM. (TOP) PEAK 1-D VAM FORCES ARE THE SAME AS THE CONVENTIONAL FORCES. (BOTTOM) PEAK 2-D VAM FORCES ARE SMALLER THAN CONVENTIONAL TOOL FORCES.................................................................9

FIGURE 1-4. EFFECT OF DIAMOND TOOL WEAR FOR 1-D AND EVAM WHEN DIAMOND TURNING STEEL. (LEFT) AVERAGE TOOL FORCES AS A FUNCTION OF CUMULATIVE CUTTING DISTANCE. (RIGHT) SURFACE ROUGHNESS AS A FUNCTION OF CUMULATIVE CUTTING DISTANCE.........................................................................................................................10

FIGURE 1-5. MOLDS MACHINED IN HARD STEEL USING EVAM [48].............................................13

FIGURE 1-6. MICRO-CHANNELS IN MACHINED IN PLATED COPPER USING THE ULTRAMILL EVAM SYSTEM AND 50-µM NOSE RADIUS ROUND-NOSED DIAMOND TOOL. (LEFT) 3 µM DEEP CHANNELS (CENTER) BURR-FREE EDGE DETAIL OF 1.5 µM DEEP CHANNEL (RIGHT) EXIT DETAIL OF CHANNEL. [3].................................................................................................................................13

FIGURE 1-7. SEM IMAGE AND INTERFEROGRAM OF 200 µM SQUARE ANGSTROM SYMBOL……15

FIGURE 1-8. INTERFEROGRAM AND SEM IMAGE OF THUNDERBIRD MADE IN 17-4 PH STAINLESS STEEL [3]..........................................................................................................................15

FIGURE 2-1. COORDINATE SYSTEM USED FOR EVAM................................................................19

FIGURE 2-2. MOTION OF TOOL TIP IN EVAM..............................................................................21

FIGURE 2-3. IMPORTANT ANGLES WHEN MACHINING ON A SLOPED SURFACE......................24

FIGURE 2-4. DEPTH OF CUT FOR EVAM ON AN INCLINED SURFACE........................................26

FIGURE 2-5. FORMATION OF SURFACE FEATURES IN EVAM.........................................................29

FIGURE 2-6. THEORETICAL PV SURFACE ROUGHNESS AS A FUNCTION OF UPFEED INCREMENT F UP AND FEED MOTION ANGLE Z......................................................................................31

FIGURE 3-1. ULTRAMILL CONCEPT............................................................................................34

FIGURE 3-2. CUTAWAY VIEW OF THE ULTRAMILL, SHOWING ITS FUNCTIONAL COMPONENTS................................................................................................................................35

FIGURE 3-3. PIEZOELECTRIC STACKS USED IN THE ULTRAMILL. (LEFT) TRIANGULAR CROSS SECTION. (RIGHT) HALF-ROUND CROSS SECTION. IN BOTH PHOTOS, THE CERAMIC HALF-ROUND PIVOT PINS ARE NOT YET INSTALLED.........................................................36
FIGURE 3-4. SEM IMAGES OF SINGLE-CRYSTAL DIAMOND TOOLS. (A) ROUND-NOSE GEOMETRY. TOOL SHOWN HAS 1 MM NOSE RADIUS. (B) SHARP NOSE TOOL. TOOL SHOWN IS “DEAD SHARP” WITH A NOSE ANGLE OF APPROXIMATELY 70º BETWEEN THE CUTTING EDGES..............................................................38
FIGURE 3-6. ULTRAMILL INSTALLED ON NANOFORM DTM, WITH OIL-BEARING Y-AXIS. (LEFT) PRIOR TO INSTALLATION OF Y-AXIS CAMERA (CONFIGURATION II). (RIGHT) Y-AXIS CAMERA INSTALLED (CONFIGURATION III-A/B). SIDEVIEW TOUCOFF CAMERA REMOVED IN THIS VIEW..........................................................40
FIGURE 3-7. ULTRAMILL BOLTED TO Y-AXIS CARRIER PLATE ...........................................40
FIGURE 3-8. ACCELEROMETER LOCATIONS FOR Y-AXIS TESTING.................................43
FIGURE 3-9. OSCILLOSCOPE RESPONSE, ACCELEROMETER 1, FOR OIL-BEARING Y-AXIS WITH ULTRAMILL OPERATING AT 1000 HZ...........................................................43
FIGURE 4-1. OPEN-CIRCULATION COOLING ARRANGEMENT ........................................50
FIGURE 4-2. CLOSED-CIRCULATION COOLING CONFIGURATION.................................50
FIGURE 4-3. CROSSFEED PROFILES FOR “POTATO CHIP” PART DESCRIBED IN CHAPTER 7, OBTAINED BY WHITE-LIGHT INTERFEROMETRY. (LEFT) PROFILE FOR OPEN-CIRCULATION COOLING SHOWS FORM ERROR BECAUSE OF ULTRAMILL TEMPERATURE VARIATION. (RIGHT) CLOSED-CIRCULATION COOLING NEARLY ELIMINATES FORM ERROR BY IMPROVING TEMPERATURE STABILITY........................................51
FIGURE 4-4. THERMOCOUPLE INSTALLATION LOCATIONS ON ULTRAMILL...............52
FIGURE 4-5. (LEFT) VIDEO-MICROSCOPE PICTURE OF CAPACITANCE GAUGE AND ALUMINUM TARGET INSTALLED ON TOOLHOLDER IN PLACE OF DIAMOND TOOL. (RIGHT) TARGET MOTION RELATIVE TO CAPACITANCE GAUGE........................................54
FIGURE 4-6. ULTRAMILL TEMPERATURES FOR OPEN-CIRCULATION COOLING CONFIGURATION.................................................................57
FIGURE 4-7. ULTRAMILL TEMPERATURES FOR MACHINING TEST USING CLOSED-CIRCULATION COOLING. COOLANT FLOW WAS REDUCED FROM 0.4 L/MIN TO 0.16 L/MIN AT 38 MINUTES’ ELAPSED TIME.......................................................57
FIGURE 4-8. ULTRAMILL TEMPERATURES RecorderDURING 3-HR LONG MACHINING RUN. AFTER BLOCK TEMPERATURES TC8 AND TC10 REACH STEADY STATE AT 80 MINUTES, THEY SHOW VARIATION OF LESS THAN 0.1ºC..............................................59
FIGURE 4-9. OPEN-CIRCULATION CAPACITANCE GAUGE TEST.................................61
FIGURE 4-10. CLOSED-CIRCULATION CAPACITANCE GAUGE TEST.............................61
FIGURE 4-11. FORM ERROR BASED ON A 0.5 MM LONG X 0.4 MM WIDE AREA IN THE CENTRAL REGION OF THE RASTERED FLAT.................................................................64
FIGURE 6-8. STRUCTURING ELEMENT GENERATED BY MORPH3D PROGRAM. ELEMENT SHOWN HERE IS BASED ON EQUATION 6-1 USING A 1 MM TOOL RADIUS AND 22 μM X 4 μM VIBRATION ELLIPSE.

FIGURE 6-9. VALIDATION CASES FOR TESTING THE MATLAB MORPHOLOGY MOTION PROGRAM.

FIGURE 6-10. ACCURACY OF MORPHOLOGY METHOD FOR CONCAVE SPHERICAL SURFACE PROBED BY HEMISPHERICAL STRUCTURING ELEMENT. “DEVIATION” IS THE DIFFERENCE IN Z COORDINATES BETWEEN THE THEORETICAL AND PROGRAM-GENERATED IMAGE SURFACE.

FIGURE 6-11. MINIMUM DEVIATION Δ AS A FUNCTION OF CROSSFEED POSITION FOR 500-μM RADIUS CONCAVE SPHERICAL SURFACE (SEE FIGURE 6-10).

FIGURE 6-12. ACCURACY OF MORPHOLOGY METHOD FOR CONVEX SPHERICAL SURFACE PROBED BY HEMISPHERICAL STRUCTURING ELEMENT. “DEVIATION” IS THE DIFFERENCE IN Z COORDINATE BETWEEN THE THEORETICAL AND PROGRAM-GENERATED IMAGE SURFACE.

FIGURE 6-13. ACCURACY OF MORPHOLOGY METHOD FOR SINUSOIDAL SHEET PROBED BY STRUCTURING ELEMENT BASED ON ELLIPSOIDAL FEED PATH. “DEVIATION” IS THE DIFFERENCE IN Z-COORDINATE BETWEEN THE THEORETICAL AND PROGRAM-GENERATED IMAGE SURFACE.

FIGURE 6-14. DISTORTION OF ELLIPTICAL VIBRATION PATH CAUSED BY UPFEED AND VERTICAL FEED MOTION.

FIGURE 6-15. TEST CASE FOR CHECKING SENSITIVITY TO MACHINING TILTED SURFACES.

FIGURE 6-16. DEVIATION BETWEEN DESIRED AND ACTUAL SURFACE PRODUCED FOR TILTED SURFACE CASE SHOWN IN FIGURE 6-14. FEED MOTIONS DISTORT THE STRUCTURING ELEMENT FROM IDEAL ELLIPTICAL VIBRATION PATH.

FIGURE 7-1. LAYOUT SKETCH SHOWING CROSS-SECTION OF THE “KNIFE-EDGE” FEATURE. NOTE BACKFEED STEP TO CREATE THE FEATURE, AS WELL AS THE CURVED LOWER PORTIN OF THE LEADING EDGE SLOPE DUE TO THE ELLIPTICAL VIBRATION PATH.

FIGURE 7-2. SEM IMAGE OF “KNIFE EDGE” FEATURE.

FIGURE 7-3. WHITE LIGHT INTERFEROGRAM AND PROFILE OF KNIFE-EDGE PART.

FIGURE 7-4. DESIGN SKETCH FOR THE “FLYING WING” FEATURE.

FIGURE 7-5. WHITE-LIGHT INTERFERORGRAMS OF THE “FLYING WING” PART, WITH PROFILES AT TWO LOCATIONS SHOWING CONCAVE FORM ERROR IN THE STEEPLY SLOPED SURFACES.
FIGURE 7-6. SEM IMAGE OF “FLYING WING” FEATURE. (LEFT) OVERALL PART (RIGHT) DETAIL SHOWING CONCAVE SLOPE AND LINEAR FEATURE ERRORS..................................................123


FIGURE 7-8. PIEZO STACK TEMPERATURE WHILE MACHINING FINISH PASS OF THE FLYING WING......................................................................................................................126

FIGURE 7-9. “POTATO CHIP” PART WITH 3-D SCULPTED GEOMETRY..................................................129

FIGURE 7-10. WHITE-LIGHT INTERFEROGRAM OF THE “POTATO CHIP” PART. PART SHOWN WAS MADE USING CLOSED-CIRCULATION FOR THE ULTRAMILL.................................129

FIGURE 7-11. UPFEED AND CROSSFEED PROFILES OF “POTATO CHIP PART” EXTRACTED FROM WHITE-LIGHT INTERFEROMETRY DATA AND PLOTTED USING MATLAB............................................................131

FIGURE 8-1. DESIGN DIMENSIONS OF SPHERICAL REFLECTOR................................................................133

FIGURE 8-2. CENTERLINE CROSS SECTION OF CONCAVE SPHERICAL REFLECTOR.........................133

FIGURE 8-3. TOOLPATHS FOR 2 RASTER PASSES ON THE SPHERICAL REFLECTOR.....................136

FIGURE 8-4. MACHINING SCHEME FOR THE CONCAVE SPHERICAL REFLECTOR..........................136

FIGURE 8-5. GPI INTERFEROGRMS OF SPHERICAL REFLECTORS MADE WITH THE ULTRAMILL. PLOTS SHOW DEVIATION OF THE SURFACE FROM THE DESIGN RADIUS OF CURVATURE (4.5 MM CONCAVE)...............................................................138

FIGURE 8-6. HOW X-Y SQUARENESS ERROR CAUSED ASTIGMATISM.............................................139

FIGURE 8-7. PREVENTING CROSSFEED TRENCH AT BASE OF CONCAVE PART (TOP) FRICTION IN Z-AXIS BALLSCREW NUT DURING REVERSAL OF DIRECTION CAUSES IT TO LAG BEHIND X-AXIS AT BOTTOM OF PART (BOTTOM) SLOWING DOWN X-AXIS AFTER Z-AXIS REVERSAL KEEPS THE TWO AXES SYNCHRONIZED..................................................140

FIGURE 8-8. WHITE-LIGHT INTERFEROMETRY PROFILES AT BOTTOM OF THE CONCAVE REFLECTOR (BEST-FIT SPHERE REMOVED). (LEFT) EFFECT OF Z-AXIS FRICITION ON MACHINES SURFACE (RIGHT) SUPPRESSION OF “TRENCH” FEATURE BY SLOWING DOWN X-AXIS AT REVERSAL POINT. HIGH-FREQUENCY FEATURES ARE CAUSED BY ULTRAMILL VIBRATION DUE TO PRESSURE PULSATIONS WITH CLOSED-CIRCULATION COOLING..................................................................................................................142

FIGURE 8-9. LOCATION OF SURFACE ROUGHNESS MEASUREMENT ZONES.....................................145

FIGURE 8-10. WHITE-LIGHT INTERFEROGRMS FOR SPHERICAL REFLECTORS, SHOWING DIFFERENCE IN MACHINED SURFACE BETWEEN (A) HARD-PLATED COPPER AND (B) ALUMINUM.................................................................................................................146
FIGURE 9-1. SCHEMATIC OF LIGHT PROPAGATING THROUGH AN OPTICAL FIBER BY TOTAL INTERNAL REFLECTION, AND ANGLE OF ACCEPTANCE AND DIVERGENCE

FIGURE 9-2. FREE-SPACE BEAM SPLITTER WITH REFRACTING ELEMENTS TO COLLIMATE LIGHT FROM THE SOURCE FIBER AND TO FOCUS THE BRANCH BEAMS INTO THE RECEIVING FIBERS

FIGURE 9-3. FIBER OPTIC BEAM SPLITTER, USING 3-D CURVED REFLECTOR

FIGURE 9-4. CONCEPTUAL SKETCH SHOWING PHYSICAL IMPLEMENTATION OF REFLECTION BEAM SPLITTER

FIGURE 9-5. BEAM BENDER TEST SETUP FOR DEMONSTRATING THE FOCUSING REFLECTOR CONCEPT

FIGURE 9-6. REFLECTION PROPERTY OF AN ELLIPSE. ANY RAY ORIGINATING AT FOCUS $F_1$ (BLUE RAYS) IS REFLECTED AND PASSES THROUGH THE OTHER FOCUS, $F_2$ (RED RAYS)

FIGURE 9-7. LAYOUT OF ELLIPTICAL CENTERLINE PROFILE OF REFLECTOR SURFACE FOR THE BEAM BENDER

FIGURE 9-8. (LEFT) WORKPIECE SETUP FOR MACHINING BEAM BENDER REFLECTOR (RIGHT) DESIGN SURFACE IS ROTATED THROUGH ANGLE $\Psi$ TO ALIGN IT FOR MACHINING

FIGURE 9-9. PREPARATION AND INSTALLATION OF THE ALUMINUM WORKPIECE USED TO MAKE THE BEAM BENDER REFLECTOR. (A) MILLING 45° CHAMBERS. (B) WORKPIECE WITH CHAMBERS IN THE WORKHOLDER. (C) WORKPIECE AND WORKHOLDER INSTALLED ON THE NANOFORM. A FEDERAL GAUGE IS USED TO ALIGN THE CHAMFERED WORK SURFACE WITH THE PLANE FORMED BY THE DTM’S X AND Y AXES

FIGURE 9-10. TEST SETUP FOR THE BEAM BENDER PART

FIGURE 9-11. INSTALLATION OF ALUMINUM WORKPIECE CONTAINING BEAM BENDER REFLECTOR, SOURCE, AND RECEIVING FIBERS

FIGURE 9-12. SKETCHES SHOWING EXPECTED AND ACTUAL IMAGES OBTAINED BY PROJECTING THE REFLECTED BEAM ONTO A WHITE CARD. THE REFLECTOR HAS FIGURE ERROR, SO THE ACTUAL IMAGE IS TOO LARGE FOR THE BEAM BENDER TO WORK AT HIGH EFFICIENCY

FIGURE 9-13. COMPOSITE INTERFEROGRAM OF REFLECTOR, OBTAINED BY USING NEW VIEW WHITE-LIGHT INTERFEROMETER TO STITCH 72 HIGH-MAGNIFICATION DATA SETS

FIGURE 9-15. REFLECTOR CENTERLINE PROFILE DETAILS (CALLED OUT FIGURE 9-14). ........166
FIGURE 9-16. DEVIATION BETWEEN DESIGN AND MEASURED REFLECTOR SURFACES ALONG CENTERLINE, USING LEVELING AN ALIGNMENT SHOWN IN FIGURES 9-14 AND 9-15...........167
FIGURE 9-17. MEASUREMENT ZONES FOR SURFACE ROUGHNESS........................................168
FIGURE 9-18. ESTIMATED DEVIATION PROFILES BETWEEN DESIGN AND PREDICTED
REFLECTOR SURFACE, FOR COMBINED SQUARENESS AND VIBRATION PATH ERRORS.
NOTE APPARENT TOOL FEED DIRECTION IS DRAWN OPPOSITE OF THAT FOR CENTERLINE
PROFILES IN FIGURES 9-14 AND 9-15.................................................................172
FIGURE 10-1. COMPARISON OF GROOVES OF EQUAL DEPTH MADE BY ROUND-NOSED TOOLS
WITH NOSE RADIUS OF 1 MM AND 50 µM...............................................................173
FIGURE 10-2. (LEFT) SHARP-NOSE TOOL WITH 40° NOSE-ANGLE (RIGHT) GROOVES AND
POSITIVE FEATURES (RIDGES) POSSIBLE WITH 40° NOSE-ANGLE TOOL....................175
FIGURE 10-3. THEORETICAL CROSSFEED SURFACE FEATURE HEIGHT FROM DEAD-SHARP
TOOLS.........................................................................................................................175
FIGURE 10-4. ALTERNATE SHARP-NOSE TOOL GEOMETRIES TO PRODUCE BETTER SURFACE
ROUGHNESS THAN CAN BE CREATED WITH DEAD-SHARP TOOLS..............................176
FIGURE 10-5. RAKE FACE VIEW OF ROUND-NOSE AND SHARP-NOSE TOOLS..........................177
FIGURE 10-6 MAKING TETRAHEDRONS BY ROTATING SPINDLE 60° BETWEEN EACH SET OF
CUTS.............................................................................................................................179
FIGURE 10-7. SUCCESSIVE TOOL PASSES IN SAME GROOVE AT INCREASING DEPTH OF
CUT................................................................................................................................181
FIGURE 10-8. SEM IMAGES OF TETRAHEDRON ARRAY MACHINED IN HARD-PLATED COPPER.
SMALL INTERSTITIAL FEATURES WERE CAUSED BY TOOL CENTERING ERROR.............183
FIGURE 10-9. SEM IMAGES OF TETRAHEDRONS MACHINED IN OFHC COPPER (LEFT) BURR
RESULTED WHEN ONLY ROUGHING CUTS WERE USED. (RIGHT) BURR WAS REDUCED BY
USING FINISHING CUTS.................................................................................................183
FIGURE 10-10. SEM IMAGES OF TETRAHEDRONS MACHINED IN HARD-PLATED COPPER.
(LEFT) SLIGHT BURR ON FEATURE EDGES WHEN ONLY ROUGHING CUTS WERE USED.
(RIGHT) FINISH CUTS ELIMINATED BURR....................................................................184
FIGURE 10-11. BURR FORMATION IN TETRAHEDRON FEATURES MADE FROM DIFFERENT
MATERIALS......................................................................................................................185
FIGURE 10-12. BINARY THUNDERBIRD FEATURE MACHINED IN 17-4 PH STAINLESS STEEL
USING 1 MM RADIUS ROUND-NOSED TOOL. DETAIL OF WING FEATURE SHOWS
NEGLIGIBLE BURR FORMATION.................................................................................186
FIGURE 10-13. FESEM IMAGES SHOWING WEAR ON DEAD-SHARP DIAMOND TOOL AFTER
ACCUMULATING 2 M RASTER MACHINING DISTANCE IN 17-4 PH STAINLESS STEEL AND
1.6 M IN COPPER........................................................................................................188
FIGURE 10-14. WEAR OF ROUND-NOSED DIAMOND TOOL AFTER IT WAS USED TO CUT TOOL
STEEL WITH EVAM. SEM IMAGES TAKEN AFTER 20 M RASTER MACHINING
DISTANCE....................................................................................................................189
FIGURE 10-15. SEM IMAGES COMPARING WEAR OF DEAD-SHARP AND ROUND-NOSED
DIAMOND TOOLS AFTER APPROXIMATELY 2 M RASTER MACHINING DISTANCE IN
17-4 PH STAINLESS STEEL..........................................................................................189
FIGURE 10-16. SHARP-NOSE TOOL WITH 40 DEGREE INCLUDED NOSE ANGLE AND ROUNDED
TIP. SEM IMAGE........................................................................................................191
FIGURE 10-17. DETAILS OF GROOVES MADE IN HARD-PLATED COPPER USING THE 40º NOSE-
ANGLE TOOL SHOWN IN FIGURE 10-16. NOTE ABSENCE OF BURR. SEM IMAGES........191
FIGURE 10-18. DESIGN SKETCH FOR MICROMIXER MADE WITH SHARP-NOSE TOOL...........192
FIGURE 10-19. SEM IMAGES SHOWING RESULTS OF MICROMIXER FABRICATION EXPERIMENT.
(LEFT) CHANNEL JUNCTION ERROR. (RIGHT) DETAIL OF JUNCTION SHOWING RIBBON
BURRS ON CHANNEL FLOOR......................................................................................192
FIGURE 10-20. CROSSFEED PROFILE SHOWING LINEAR FORM OF MICROMIXER RESERVOIR
WALL AND FLOOR, OBTAINED BY WHITE-LIGHT INTERFEROMETRY......................193
FIGURE 10-21. (LEFT) UPFEED SURFACE PROFILE OF SIDEWALL. (RIGHT) AUTOCOVARIANCE
PLOT FOR THE SIDEWALL PROFILE..........................................................................193
FIGURE A-1. TWO CONSECUTIVE TOOL PASSES MODELED BY MATLAB. THE LOWER HALF OF
EACH TOOL PASS IS SHOWN. THE LOWER HALF OF EACH TOOL PASS IS DEFINED OVER A
LENGTH 2*+0.5*FUP. STARTING POINT FOR THE SECOND PASS IS DISPLACED FUP IN THE
UPFEED DIRECTION.................................................................................................214
FIGURE A-2. SUCCESSIVE CALCULATED TOOL PASSES, WITH STARTING POINTS FUP APART.
NOTE HOW OVERLAPPING TOOL PASSES PRODUCE MULTIPLE POINTS (X, Z) FOR EACH
VALUE OF X..................................................................................................................214
FIGURE A-3. THEORETICAL SURFACE, CALCULATED BY REMOVING ALL BUT THE
BOTTOMMOST POINTS AT EACH VALUE OF X, FROM THE OVERLAPPING TOOL PASS
TRACES IN FIGURE A-2(B)..........................................................................................215
FIGURE C-1. PROFILES OF TEST GROOVE. SPIKE-LIKE FEATURES IN ENTRANCE (A) AND EXIT
REGIONS (B) ARE INTERFEROMETRY ARTIFACTS, AS SHOWN BY (C) SMOOTH PROFILE
OBTAINED WITH AN ATOMIC FORCE MICROSCOPE (AFM)........................................230

xix
FIGURE D-1. Y-AXIS CAMERA—CRITICAL DIMENSIONS FOR LAYOUT...........................................234
FIGURE D-2. SCREEN SHOT OF MONITOR SHOWING Y-AXIS CAMERA CROSSHAIRS
POSITIONED ON A FEATURE OF INTEREST.................................................................235
FIGURE D-3. SCHEME FOR USING THE Y-AXIS CAMERA TO POSITION THE TOOL TIP AT A
DESIRED POINT ON THE WORKPIECE........................................................................236
FIGURE D-4. EXAMPLE OF AVERAGING THE LOCATION OF FEATURE EDGES TO ESTIMATE
CAMERA OFFSETS ΔX AND ΔY. FEATURE EDGES IN SKETCH ARE SHOWN MORE
CLEARLY THAN ON REAL FEATURES AS THEY APPEAR IN CAMERA MONITOR..............237
FIGURE D-5. SKETCH SHOWING PLUNGE CUT FEATURES MADE BY DIFFERENT TOOLS.
22 μM X 4 μM VIBRATION ELLIPSE AND DEPTH OF CUT 1 μM........................................237
FIGURE D-6. (LEFT) TOUCHOFF CAMERA INSTALLATION WHEN WORKPIECE HAS BEEN
TURNT FLAT TO OPTICAL QUALITY FINISH. (RIGHT) MONITOR VIEW SHOWING TOOL
AND REFLECTION IN WORKPIECE................................................................................240
FIGURE D-7. TOUCHOFF CAMERA INSTALLED IN ALTERNATE POSITION PROVIDING ANGLLED
VIEW OF SURFACE........................................................................................................242
FIGURE E-1. THERMOCOUPLE INSTALLATION LOCATIONS ON ULTRAMILL..................246
FIGURE F-1. LABVIEW CONFIGURATION AND CONTROL FOR THERMAL MEASUREMENT
CARD...............................................................................................................................249
FIGURE F-2. TEMPERATURE RECORDING AND DISPLAY BLOCK........................................249
FIGURE F-3. TEMPERATURE RECORDING AND DISPLAY BLOCK........................................249
FIGURE F-4. TEMPERATURE RECORDING AND DISPLAY BLOCK........................................250
FIGURE G-1. ULTRAMILL ENERGY BALANCE....................................................................252
FIGURE G-2. ULTRAMILL TEMPERATURE DATA DURING ENERGY BALANCE TEST. HALF-
ROUND STACKS OPERATING AT 1000 HZ FREQUENCY AND APPLIED VOLTAGE OF
400 Vpp. THERMOCUBE SETPOINT 20°C. PLASTIC THERMAL BARRIER IN PLACE.
ULTRAMILL ENERGY BALANCE..................................................................................254
FIGURE G-3. ULTRAMILL ENERGY BALANCE FOR TEMPERATURE DATA SHOWN IN FIGURE G-2.
HALF-ROUND STACKS OPERATING AT 1000 HZ FREQUENCY AND APPLIED VOLTAGE OF
400 Vpp. THERMOCUBE SETPOINT = 20°C. PLASTIC THERMAL BARRIER IN
PLACE.............................................................................................................................255
FIGURE H-1. SPANZITFEL HYPOTHESIS APPLIED TO UPFEED FEATURE FORMATION IN
EVAM............................................................................................................................259

XX
FIGURE H-2. COMPARISON OF ACTUAL SURFACES AND SIMULATED SURFACE BASED ON THE MINIMUM CHIP THICKNESS EFFECT AT TWO UPFEED VELOCITIES. (LEFT) PROFILES OBTAINED FROM ACTUAL SURFACES. (RIGHT) SIMULATED PROFILES..................261
FIGURE H-3. COMPARISON OF SIMULATED SPANZITFEL ROUGHNESS AND ACTUAL RESULTS FOR PART 62 (OPEN-CIRCULATION COOLING AND FRESHLY LAPPED TOOL).................................................................262
FIGURE H-4. GENERATION OF SURFACE FROM TWO SUCCESSIVE TOOL PASSES..................264
FIGURE H-5. GENERATION OF SPANZITFEL.................................................................265
FIGURE H-6. SURFACE GENERATED WHEN THE THIRD TOOL PASS MISSES THE SPANZITFEL PREVIOUSLY GENERATED.................................................................266
FIGURE H-7. FEATURE HEIGHT IS REDUCED BY THE THIRD TOOL PASS WHEN UPFEED INCREMENT IS SMALL, RELATIVE TO SPANZITFEL HEIGHT..............................266
FIGURE M-1. LAYOUT OF SOURCE AND RECEIVING FIBERS FOR THE BEAM BENDER. THE FIBER CENTERLINES DEFINE AND XZ COORDINATE SYSTEM.................................306
FIGURE M-2. FINDING SOURCE BEAM INTERSECTION WITH ELLIPTICAL RELECTOR SURFACE, AND THE ANGLE OF REFLECTED BEAM ENTERING THE RECEIVING FIBER........310
FIGURE N-1. ROTATION OF ELLIPSOID REFLECTOR FROM “PART COORDINATE SYSTEM” (TOP) TO MACHINING COORDINATE SYSTEM (BOTTOM)........................................311
FIGURE P-1. TETRAHEDRON SLICED FROM A CUBE CORNER.................................328
FIGURE P-2. BASE TRIANGLE OF A TETRAHEDRON.................................................330
FIGURE P-3. INTERIOR TRIANGLE OF A TETRAHEDRON...........................................330
FIGURE Q-1 CUT FIRST MEASUREMENT GROOVE WITH SPINDLE AT INITIAL ROTATION ANGLE........................................................................................................333
FIGURE Q-2 ROTATE SPINDLE 180 DEGREES ..........................................................333
FIGURE Q-3 CUT SECOND GROOVE AFTER ROTATING SPINDLE 180 DEGREES............334
FIGURE Q-4 ROTATE SPINDLE BACK TO ORIGINAL POSITION AND MEASURE GROOVE SEPARATION ........................................................................................................335
FIGURE Q-5 MEASUREMENT OF CENTERING FEATURES CUT INTO WORKPIECE. THE CENTERLINE DISTANCES OF THE X AND Y FEATURES ARE CALCULATED USING MEASUREMENTS OF THE Y CENTERING GROOE EDGES (Y₁ AND Y₂) AND ENDPOINTS OF THE X CENTERING FEATURE (X₁ AND X₂)..................................................336
1 INTRODUCTION

1.1 OVERVIEW

Microstructures can be defined as manufactured 2-D or 3-D features with critical dimensions ranging from several millimeters to smaller than 1 μm [15]. They can be functional small-scale devices in themselves, or features of a larger component. Accurate, economical microstructure generation is essential to the success of miniature products such as micro-electromechanical systems (MEMS), micro-optics, microfluidic “lab-on-a-chip” packages, and replication masters for molding-type processes.

Mechanical machining processes such as micro-milling and diamond turning can create microstructures with complex geometry [14, 15, 17]. When used with single-crystal diamond tools, such methods can achieve optical-quality surface finishes in ductile metals. However spindle-type cutting processes are plagued by issues such as runout, tool deflection and vibration, which limit the minimum feature size and tolerance. Moreover diamond tools wear at unacceptably high rates when machining ferrous metals and many brittle materials of interest.

Elliptical vibration-assisted machining (EVAM) eliminates problems caused by use of a high-speed spindle. In EVAM, the tool tip is driven in a tiny ellipse at high frequency, superimposing a periodic micrometer-range oscillation onto the main feed motion. The elliptical motion is produced by moving the tool cutting edge in a defined path, rather than by rotating the tool around a spindle axis. EVAM also reduces the tool force, improves surface finish, and extends diamond life when machining steel. To date, it has been mostly used to make binary and low-aspect ratio structures with vertical dimensions of only a few micrometers. The current work investigates EVAM’s ability to fabricate 3-D microstructures with sizes, aspect ratios, vertical dimensions, and surface finishes characteristic of micro-optic, micro-fluidic and MEMS applications.
1.2 MICROSTRUCTURES MADE BY MECHANICAL MICROMACHINING

1.2.1 Mechanical Micromachining

Microdevice fabrication methods have been researched for more than two decades. Lithography-based techniques adapted from the semiconductor industry are used to make many kinds of MEMS and microfluidic devices. These processes are typically limited to silicon-type materials and are usually economical only at large production volumes. Also lithography can only make 2-1/2 D and stepwise planar 3-D geometries. In silicon devices anisotropic etch behavior results in arbitrary side wall angles in the depth direction. Other processes exist that can create micro-structures with complex 3-D geometry, including focused ion beam milling (FIB), laser ablation, micro-electrodischarge machining (micro-EDM), micro-electrochemical machinining (micro-ECM), and LIGA. Each of these techniques has limitations: extremely low material removal rate (FIB), limited material range (LIGA, micro-EDM and micro-ECM), or large tolerances and/or low resolution (laser ablation methods, micro-EDM) [2, 6].

Mechanical micromachining is the adaptation of traditional chip-making processes, such as turning or milling, to the micro-scale. These processes overcome several disadvantages of other microfabrication techniques. They can produce features with complex sculpted 3-D geometry and optical-quality surface finish in polymers, ductile metals such as aluminum, copper, and electroless nickel, as well as semiconductor materials including silicon and germanium [14]. Material removal rates are $10^3$ to $10^5 \mu m^3/s$, comparable to laser methods and at least three orders of magnitude greater than for FIB milling. Dimensional tolerances

---

1 "LIGA" is a German acronym meaning "lithography, electroforming, and molding". It is a multi-process fabrication technology in which X-rays or UV beams are used to etch cavity structures in photoresist material. These features are then plated (with gold, nickel, or copper) to create a master for use with replication methods such as injection molding.
are typically in the 0.2 to 1.0 µm range, compared to 0.02 to 0.2 µm for FIB [2]. For applications where such precision is acceptable, mechanical micromachining offers a set of microstructure fabrication processes with large potential.

Two important types of mechanical micromachining are single-point diamond-turning (SPDT), and micro-milling based on high-speed spindles. SPDT uses diamond tools and lathe-configuration diamond turning machines (DTM's) to create optical-quality surfaces with nanometer-amplitude surface roughness. In general, feature geometries are limited to those which can be made by turning or facing operations, and the tool clearance angle limits the maximum slope of the cut surface. In SPDT the primary feed motion is achieved by rotating the workpiece past a "fixed" tool while surface form can be created by short-range tool moves in the depth of cut direction. The depth of cut variations may be made by the Z-axis (turning and slow slide servo) or by using a fast tool servo (FTS). The FTS is a high-bandwidth auxiliary axis which changes the depth of cut as a function of workpiece angular position and radius from rotational centerline. However short stroke length, frequency of operation, and machine dynamics of the tool servo make it difficult to create small high-aspect ratio features. For example, Panusittikorn [40] cut concave cylindrical and spherical surfaces that were over 100 µm deep, but which were longer than 5 mm in the tangential direction (an aspect ratio of 0.02). Long-range fast tool servos under development [7, 9] will eventually increase aspect ratio and vertical dimension range.

Micro-milling processes include vertical milling using end, ball, or pseudo-ball mills; horizontal milling; and micro-flycutting. Some recent examples from the literature illustrate these processes. Sweatt et al [58] used concave 0.660 mm diameter diamond end mills to plunge-cut convex aspherical lenslets in a polymer material. Spherical diamond end-mills were modified using focused ion beam (FIB) machining to make tools with the required aspheric profile. Freidrich and Vasile [20] used custom-made end-mills to create stepped trench features 8 µm wide in PMMA with an aspect ratio of 8. The milling tools were shaped
from cobalt-steel blanks using FIB. Takeuchi et al [45] and Kawai et al [24] used diamond pseudo-ball end-mills with multi-axis precision CNC machines to create 3-D figurines with submillimeter details in gold and copper. Machining time for these parts was as long as 36 hours due to the low material removal rate resulting from the small tool diameter. Microflycutting is illustrated by Takeuchi et al [15, 46] who made a variety of microstructured surfaces using spindle-driven single-crystal diamond tools. These microstructures, cut in brass, were 20 to 50 μm deep with aspect ratios exceeding one. Different feature cross-sections (e.g. rectangular, trapezoidal, triangular, etc.) were made by using diamond tools of several different cross-sections. To increase the variety of feature geometries, the cutting spindle axis was oriented at different angles to the workpiece which itself was fed along a single direction.

1.2.2 Mechanical Micromachining Issues

Micro-milling can make a wide range of 3-D geometries, complimenting those achievable by diamond turning. However accuracy and resolution of spindle-driven tools is reduced by several error sources. These include runout, chatter, and deflection of long tool shafts under machining loads. The minimum feature size is limited by the diameter of the tool. However as the tool diameter is reduced to improve resolution, tool deflection becomes more severe and tool strength is reduced. Smaller-diameter tools require higher spindle RPM to maintain the same cutting velocity. Preparation of milling-type tools is potentially a time-consuming, expensive operation as demonstrated in several of the above examples which used FIB machining to shape cutting edge profiles.

Single-crystal diamond tools are desirable for micromachining due to their great strength and because they can be manufactured with extremely sharp cutting edge radii in the tens of nanometers. Unfortunately diamond life is very poor when cutting ferrous metals, as well as other hard metals including nickel alloys and titanium. Diamond wear for these materials, while not understood in detail, is chemical in nature and is related to the propensity for
carbon to interact with the workpiece metal [41, 54]. Activation energy studies indicate the loss of carbon from the diamond occurs through formation of metal-carbon or metal-carbon-oxygen compounds in the workpiece, rather than by graphitization [29]. Brittle materials including glass also cause rapid diamond wear, although by mechanical abrasion instead of chemical mechanisms [27, 28]. Other tool materials such as tungsten carbide and cubic boron nitride (CBN) resist wear when machining ferrous metals, but are incapable of producing optical quality finishes. They can also produce severe edge burr when cutting non-ferrous metals such as brass [15, 47]. These material issues have helped limit mechanical micromachining's scope of application.

1.3 VIBRATION-ASSISTED MACHINING

Since the late 1980s, vibration-assisted machining (VAM) has been investigated as a way to overcome some of the limitations of mechanical micro-machining processes [4]. In VAM the tool oscillates at high frequency in a repeating path with an amplitude of roughly 1 to 20 μm. This vibratory motion is superimposed on the primary feed motion, so that the tool periodically breaks contact with the uncut workpiece material. As shown by Figure 1-1 the vibration can be reciprocating (1-D VAM) or elliptical (EVAM). Because the direction of vibration is normal to the tool rake face and in the same direction as the primary workpiece feed motion, VAM somewhat resembles flycutting and horizontal milling.

The absence of a high-speed spindle eliminates runout error. VAM possesses other significant advantages:

- Smaller tool forces – Cutting and thrust force in VAM are typically a fraction of the machining force observed in conventional diamond turning [8, 27, 28, 39, 49, 50, 51]. This improves surface finish and form error by supressing chatter [62] and by reducing undesired turning machine / toolholder structural vibration, relative to the workpiece.
Figure 1-1. 2 types of vibration-assisted machining. \textit{left} 1-D VAM \textit{right} Elliptical vibration-assisted machining (EVAM)

- Extended tool life when machining ferrous and brittle materials – When diamond tools are used to machine ductile metals like copper or aluminum, tool life can reach tens of kilometers' machining distance.\textsuperscript{2} But when diamond is used to cut steel, the surface finish (indicative of tool cutting edge condition) deteriorates drastically in only a few meters. In VAM acceptable surface finish in steel is maintained after more than 2000 meters [5, 28, 50, 56]. Dornfeld [15] asserts that vibration-assisted machining is presently the only practical, economic method for diamond-turning ferrous metals. VAM also extends tool life when cutting glass and other brittle materials [27, 37, 55].

- Near-elimination of burr on the edges of machined features [6, 8, 33, 39].

\textsuperscript{2} “Machining distance” is defined here as the raster distance covered by a fixed point on the EVAM tool, such as the center of the vibration ellipse. This distance differs from the total tool-workpiece contact distance, which is much greater owing to the many vibration cycles occurring in the time it takes the tool to cover a unit distance on the surface.
Ductile regime machining of brittle materials at increased depth -- Ductile regime cutting produces low surface roughness in glass and other brittle materials by eliminating brittle damage in the machined surface [21, 28, 44, 64]. Unfortunately for many important materials the ductile cutting depth is submicrometer for conventional diamond turning. With VAM, Moriwaki, Shamoto and Inoue [37] and Suzuki et al [55] were able to increase the depth of cut for ductile cutting of soda-lime glass by a factor of 15. Increased ductile machining depth has also been demonstrated for calcium fluoride [55], silicon carbide [39], tungsten carbide [32, 55], zirconia ceramic [55], fused silica [21], and brittle plastics [26]. This advantageous behavior is believed to derive from overlapping cutting passes erasing the brittle damage generated by previous passes [28, 39, 44]. Klocke and Rubenach [28, 44] argue that for glass, VAM parameters can be selected to optimize the strain rate, plastic flow and initial contact stress, to form continuous chips characteristic of true ductile machining.

**EVAM vs 1-D VAM**

EVAM offers additional advantages compared to 1-D VAM:

- Improved tool clearance when cutting tilted surfaces -- Figure 1-2 compares 1-D VAM to EVAM making a sloped surface. When cutting on a downward tilt it can be demonstrated that for 1-D VAM there are combinations of inclination angle, workpiece feed velocity and vibration amplitude that cause the tool to interfere with the newly cut surface during the backwards portion of the vibration cycle. In EVAM this interference can be avoided, because the backward motion occurs when the tool is also vertically disengaged from the workpiece.
**1-D VAM -- Tool can hit the workpiece during backfeed portion of cycle**

**EVAM - Tool clears the surface**

*Figure 1-2.* Comparison of 1-D VAM and EVAM when cutting a downward tilted surface

- Smaller instantaneous tool forces – VAM force measurements are usually made at ultrasonic vibration frequencies which exceed the sampling rate and bandwidth of most measurement systems. Therefore time-averaged values of machining forces are reported. These show reduced thrust and cutting forces for both 1-D VAM and EVAM compared to conventional diamond turning. To better understand machining force variation during the VAM cycle, Moriwaki and Shamoto [49] built a research unit capable of both 1-D VAM and EVAM which operated at just 6 Hz. As shown in the top panel of Figure 1-3,
their measurements reveal that in 1-D VAM the peak tool forces are approximately the same as in conventional machining. Smaller time-averaged forces in 1-D VAM are therefore mainly due to the tool cutting for only part of each vibration cycle. In contrast, EVAM modifies the cutting process so that peak, as well as average, tool forces are reduced (bottom panel of Figure 1-3). Cerniway [8] and Negishi [39] measured similar reductions in peak tool force at up to 1000 Hz, using the "Ultramill" EVAM system described in Section 1.4. The mechanism for force reduction may be related to the ellipse
geometry. When the ellipse major axis is horizontal (aligned to the principal feed direction), force reduction is due to very thin chips being cut during each vibration cycle. When the ellipse major axis is vertical, the thrust force becomes negative as the friction with the moving tool rake face helps “pull” the chip away from the workpiece.

- Improved diamond life in ferrous metals--Shamoto and Moriwaki [50] measured tool force and surface roughness as a function of cumulative machining distance while diamond-turning steel. They used an ultrasonic-frequency system capable of both 1-D VAM and EVAM. Figure 1-4 plots their results for time-averaged cutting and thrust forces, showing that these are smaller for EVAM compared to 1-D VAM. Furthermore, the tool forces increase more slowly with distance for EVAM. Figure 1-4 also compares surface roughness for the two processes. At any distance, the EVAM surface roughness is superior to that of 1-D VAM, and deteriorates at a slower rate. These behaviors indicate that EVAM provides a longer usable tool life, in terms of the distance which can be achieved for a cutoff surface roughness. The mechanism for this has not been clearly identified. Diamond wear rate, considered as the volumetric loss of tool material per time, is thermo-chemical.

![Figure 1-4](image.png)

**Figure 1-4.** Effect of diamond tool wear for 1-D and EVAM when diamond turning steel. (left) Average tool forces as a function of cumulative cutting distance. (right) Surface roughness as a function of cumulative cutting distance. [50]
in nature and follows the Arrhenius relationship, increasing exponentially with absolute temperature of the tool cutting edge [18, 29]. Until recently a dominant hypothesis proposed that EVAM extended diamond life because the intermittent cutting process allowed the tool to cool when not in contact with the workpiece, suppressing the driving force for the wear reactions. However finite element modeling by Lane [29] showed the tool cutting edge temperature in EVAM is actually higher than for conventional diamond turning at equivalent conditions. This is presumably because of the high surface velocity caused by superposition of high-frequency vibration onto the feed velocity. He also conducted machining experiments showing the rate of tool wear are the same magnitude for both EVAM and conventional diamond turning, although with proper parameter selection EVAM shows less wear for a given distance. Lane proposed that when cutting steel, the tool disengagement during EVAM prevented formation of a built-up edge and also reduced accumulation of chips or workpiece inclusions on the cutting edge which reduced roughness in the workpiece. However cumulative cutting distance was only a few meters in this work, and further research is necessary to explain EVAM’s relatively long tool life at distances in excess of a kilometer.

1.4 REVIEW OF MICROSTRUCTURE FABRICATION USING VAM

Despite the advantages described in Section 1.3, VAM has not seen wide use for microstructure fabrication. Most examples in the literature have been steel or glass executions of designs previously diamond-turned in non-ferrous ductile materials. Typically these parts are rotationally symmetric, or if non-rotationally symmetric, capable of being made on a lathe-type machine. They have low aspect ratios and/or small vertical dimensions. Secondary structures on these parts, such as grooves, ribs, or Fresnel lens features, are created by the tool cross section, rather than being sculpted by 3-D axis motions.
1.4.1 VAM Microstructure Examples

Klocke and Rubenach [28], Rubenach [44], and Dambon et al [13] used 1-D VAM to turn a variety of low-aspect ratio millimeter-scale optical elements and replication masters in glass and steel. Microstructures on these parts were made by using form tools with appropriate cross-sections.

Lee et al. [30] used EVAM with a round-nosed tool to machine 80 μm deep grooves in a glass substrate. These grooves were for alignment of 100 μm diameter optical fibers. Along the groove sides, where depth of cut below the original surface was shallow, ductile regime machining produced well-defined burr-free edges used to align the fibers. In the center portion of the groove, larger depths of cut were used to minimize machining time. This created brittle-fracture surface defects in the central region but this roughness was not critical to functionality. These parts had large depth but simple geometry.

Shamoto et al machined steel parts with an ultrasonic resonant-beam EVAM system. They made planar and rotationally-symmetric molds in hardened tool steel [48, 52, 53, 57]. Examples are presented in Figure 1-5. A 5-mm diameter Fresnel lens mold is shown in Figure 1-5(a). The Fresnel features were made by a round-nose tool with a 25 μm nose radius. Depth of the individual Fresnel features was 2 μm. The overall height of the part was 22 μm giving a total aspect ratio of about 0.001 (height divided by radius). The mold for a pick-up lens, shown in Figure 1-5(b), was turned in hardened steel using a round-nosed tool. The sag of this lens was too large for it to be made by precision grinding, but it was readily machined using EVAM diamond turning. Figure 1-5(c) shows a hardened steel flat plate with microribs for molding the front lens of an LED display. The ribs were 2 μm tall with a half-width of 2.7 μm. They were made by a diamond tool with a trapezoidal front cross section of 107˚ included angle.
(a) Fresnel lens mold
(b) Mold for pick-up lens
(c) Flat plate with microribs for molding LED lens

**Figure 1-5.** Molds machined in hard steel using EVAM [48]

**Figure 1-6** Micro-channels machined in plated copper using the Ultramill EVAM system and 50-μm nose radius round-nosed diamond tool. *(left)* 3 μm deep channels (center) Burr-free edge detail of 1.5 μm deep channel *(right)* Exit detail of channel. [3]
This part was made using a 5-axis precision CNC machine tool. The Shamoto EVAM system has been converted to be able to vibrate elliptically in all 3 dimensions [53] and has been used to cut tungsten molds for glass lenses [57].

1.4.2 EVAM Experience at the Precision Engineering Center

EVAM research at the Precision Engineering Center at North Carolina State University began in 2001 with emphasis on the "Ultramill" concept. The Ultramill uses parallel piezoelectric actuators driven by sinusoidal voltage signals, and a mechanical linkage to create elliptical tool tip motion. Cerniway [8] built a prototype tool while Negishi [39] developed an improved version capable of multi-kHz operation. This later version of the Ultramill is described in Chapter 3 and was used in the current work.

Cerniway and Negishi debugged the Ultramill concept and used it to explore fundamental properties of EVAM. They investigated surface roughness in grooves and flat surfaces cut in copper, aluminum, and PMMA. Both conducted tool force experiments, validating an EVAM tool force model developed by Cerniway. Cerniway performed a comparative tool wear study between EVAM and conventional diamond turning of hardened tool steel. He discovered that cutting edge wear takes place in EVAM when machining steel, but that it is smooth, regular, and non-detrimental to surface finish. Negishi showed the Ultramill could machine brittle materials by turning an optically flat surface in CVD silicon carbide.

Brocato [6] integrated the Ultramill with a 3-axis DTM providing precision XYZ motion. This enabled arbitrary binary features to be made by raster cutting. Parts made using this system included micro-channels 70 μm wide and up to 3 μm deep (Figure 1-6), the PEC's Angstrom symbol logo (Figure 1-7) and the Sandia thunderbird feature made by the author (Figure 1-8) [3, 6]. The Angstrom symbol had an overall size of 200 μm square and was cut in hard-plated copper using a round-nosed diamond tool with a 50 μm nose radius. These parts showed that the Ultramill can machine arbitrary binary microstructures with burr-free edges, at a feature size of...
resolution as small as 15 μm wide (the side bar on the top knot on the angstrom). The thunderbird was approximately 1.8 mm long x 1.2 mm wide, cut in low-carbon 17-4 PH stainless steel using a 1 mm nose-radius tool. In scanning electron microscope (SEM) images taken at 1000x, the diamond tool used to cut the thunderbird

Figure 1-7 SEM image and interferogram of 200 μm square Angstrom symbol [6]

Figure 1-8. Interferogram and SEM image of thunderbird made in 17-4 stainless steel [3]
showed no discernible cutting edge wear after 2 m accumulated raster distance. This absence of tool wear is consistent with results by others for diamond turning of steel by EVAM. However the surface roughness of all of these features was 15-25 nm RMS, considerably worse than Negishi’s results in copper and PMMA. The roughness was attributed to low stiffness in the depth of cut direction of the 3-axis DTM's air-bearing Y-axis, which caused machine vibration to be transferred into the machined surfaces.

Lane [29] used the Ultramill for detailed EVAM tool force and diamond wear investigations. These overturned then-existing assumption that reduced wear was caused by lower diamond temperatures, although alternative explanations were not conclusively proven.

1.5 Objectives of the Current Research

The current work seeks to show how the Ultramill EVAM tool can fabricate 3-D microstructures similar to those found in functional devices. Effort is required in three areas:

1. EVAM System Development

Performance limitations identified in earlier Ultramill research need to be addressed by:

- Increasing stiffness of the 3-axis Nanoform diamond turning machine's Y-axis, to reduce surface roughness caused by excessive vibration amplitude in the depth of cut direction

- Improving temperature stability of the Ultramill, to minimize form error caused by thermal expansion / contraction of its structural components.

- Improving machining system procedures, including touchoff on non-specular surfaces, precision X-Y positioning of the tool tip, tool centering, and tool lubrication.
2. Analytical Development

- Develop EVAM kinematic relationships and theoretical surface roughness prediction for motion required to make 3-D features.

- Develop a motion program planning method for raster machining of arbitrary, sculpted 3-D surfaces. This must address the need for 3-D cutter compensation including the elliptical tool vibration path.

3. Microstructure Fabrication Experiments

EVAM machining capability needs to be evaluated for two classes of 3-D features:

*Parts with complex sculpted 3-D geometry* - Made using round-nosed diamond tools, such features are characteristic of micro-optic elements. Specific objectives include making features with:

- Non-rotationally symmetric (NRS) geometry
- Vertical aspect ratio of more than 0.1 (compared to less than 0.01 for most published work).
- High sag (vertical relief) in excess of 50 μm. (Groove and binary features made to date have vertical dimensions of only a few micrometers.)
- Feature sizes ranging from ~50 μm to >1 mm.
- Optical quality figure error and surface finish

*High-aspect ratio features with straight, steep sidewalls* -- These are typical of structures such as microfluidic channels and micro-heat exchanger fins. Making them requires tools with a sharp-nose (triangular) cross section. EVAM machining with such tools has not been discussed significantly in the literature. Specific objectives include making parts with:
• Non-rotationally symmetric geometry
• Straight, steep sidewalls
• Vertical aspect ratio of 1 or greater.
• Vertical feature dimensions >10 μm.
• Minimum horizontal scale smaller than 10 μm.
2 3-D EVAM KINEMATICS AND THEORETICAL SURFACE ROUGHNESS

To make sculpted 3-D features requires an ability to cut surfaces that are inclined in the direction of primary feed motion. To generate these sloped surfaces, the workpiece feed motion must include a component in the vertical, or depth of cut, direction. The vertical feed also affects the theoretical surface roughness since in EVAM this is caused by the interaction of the elliptical vibration path with the overall feed motion of the workpiece.

Previously-developed EVAM kinematic and theoretical roughness relationships [8, 39, 49] are now extended to the case of feed motion possessing both vertical and horizontal components.

2.1 COORDINATE SYSTEM

Figure 2-1 shows the coordinate system used for the Ultramill. The X-axis is parallel to the direction of primary tool motion for cutting. This is the "upfeed" direction. The Y-axis lies along the "crossfeed" direction, perpendicular to the upfeed direction. The Z-axis is orthogonal to the X and Y axes, and parallel to the depth-of-cut direction. The negative Z direction is in the direction of increasing depth of cut.

![Figure 2-1. Coordinate system used for EVAM](image)
2.2 EQUATIONS OF MOTION

The top portion of Figure 2-2 shows the path of a diamond tool tip moving around a vibration ellipse of horizontal amplitude $A$ and vertical amplitude $B$. This cyclic tool motion is superimposed on the workpiece feed motion, which has instantaneous horizontal velocity $U(t)$ in the X-direction and instantaneous vertical velocity $V(t)$ in the Z-direction. Most diamond turning machines move the workpiece past a fixed tool to create the feed motion. However to maintain the sign convention of the elliptical vibration path, and to easily relate the tool motion to a fixed reference, hereafter the tool is assumed to be moving while the workpiece stays fixed. The net tool path for the combined elliptical and workpiece motion is shown schematically in the bottom portion of Figure 2-2, for an upward sloping surface.

The time to complete one circuit of the path is the period $T$. The vibration frequency $f$ is $1/T$. The instantaneous position of the tool relative to the ellipse center is given by

$$X(t) = A \cdot \cos(\omega \cdot t + \varepsilon)$$

$$Z(t) = B \cdot \sin(\omega \cdot t + \varepsilon)$$

where $X(t)$ and $Z(t)$ are the horizontal and vertical coordinates of the tool relative to the ellipse center, at time $t$. The angular frequency, $\omega$, is equal to $2\pi f$. The phase angle $\varepsilon$ is used to establish the initial location of the tool on the elliptical path for $t = nT$ ($n=0,1,2,\ldots$). This is useful when simulating tool motion, either analytically or numerically. Hereafter $\varepsilon$ is not shown but assumed to be present in the equations of motion.
Differentiating Equations 2-1 and 2-2 with respect to time gives the instantaneous horizontal and vertical velocity components $X'(t)$ and $Z'(t)$

$$X'(t) = -\omega \cdot A \cdot \sin(\omega \cdot t) \quad (2-3)$$
$$Z'(t) = \omega \cdot B \cdot \cos(\omega \cdot t) \quad (2-4)$$

where the variables in Equations 2-3 and 2-4 are as previously defined.
The upfeed and vertical velocity components of the overall tool motion are

\[ X'(t) = -\omega \cdot A \cdot \sin(\omega \cdot t) + U(t) \]  \hspace{1cm} (2-5)  
\[ Z'(t) = \omega \cdot B \cdot \cos(\omega \cdot t) + V(t) \]  \hspace{1cm} (2-6)

When the feed velocity components are constant \( U(t) = U \) and \( V(t) = V \). The tool motion becomes

\[ X'(t) = -\omega \cdot A \cdot \sin(\omega \cdot t) + U \]  \hspace{1cm} (2-7)  
\[ Z'(t) = \omega \cdot B \cdot \cos(\omega \cdot t) + V \]  \hspace{1cm} (2-8)

Integrating Equations 2-5 and 2-6 with respect to time yields the instantaneous position of the tool with respect to a fixed reference:

\[ X(t) = A \cdot \cos(\omega \cdot t) + \int_{0}^{t} U(\tau) d\tau \]  \hspace{1cm} (2-9)  
\[ Z(t) = B \cdot \sin(\omega \cdot t) + \int_{0}^{t} V(\tau) d\tau \]  \hspace{1cm} (2-10)

where \( \tau \) is a dummy variable for integration. The integral equations for tool tip position are required if the feed velocities are variable, for example when machining a surface with a profile curved in the X-Z plane. For constant feed velocities these equations become

\[ X(t) = A \cdot \cos(\omega \cdot t) + U \cdot t \]  \hspace{1cm} (2-11)  
\[ Z(t) = B \cdot \sin(\omega \cdot t) + V \cdot t \]  \hspace{1cm} (2-12)
In raster machining, the crossfeed velocity \( Y'(t) \) is zero, while for turning at small crossfeed rates it is small compared to the upfeed velocity. Hence \( Y(t) \) can be approximated as constant.

Equations 2-1 and 2-2 provide the instantaneous tool position relative to the ellipse center. Subtracting them from Equations 2-9 and 2-10 yields the location of the ellipse center \((X_{\text{CEN}}, \ Z_{\text{CEN}})\) relative to a fixed reference,

\[
X_{\text{CEN}}(t) = \int_{0}^{t} U(\tau) d\tau \\
Z_{\text{CEN}}(t) = \int_{0}^{t} V(\tau) d\tau
\] (2-13)

or for constant feed velocities,

\[
X_{\text{CEN}}(t) = U \cdot t \\
Z_{\text{CEN}}(t) = V \cdot t
\] (2-14)

The instantaneous direction of the feed motion \( \zeta(t) \) is

\[
\zeta(t) = \arctan \left( \frac{V(t)}{U(t)} \right)
\] (2-17)

while the instantaneous direction of the tool motion \( \kappa(t) \) is

\[
\kappa(t) = \arctan \left( \frac{Z'(t)}{X'(t)} \right) = \arctan \left( \frac{-\omega \cdot B \cdot \cos(\omega \cdot t) + V(t)}{\omega \cdot A \cdot \sin(\omega \cdot t) + U(t)} \right)
\] (2-18)

Angles \( \zeta(t) \) and \( \kappa(t) \) are measured from the X-axis as depicted in Figure 2-3(a).
(a) Direction of instantaneous velocity of ellipse center $\zeta(t)$ and tool tip $\kappa(t)$

(b) Tool instantaneous rake angle $\alpha(t)$ and clearance angle $\gamma(t)$.

**Figure 2-3.** Important angles for machining on sloped surfaces.
The instantaneous tool rake angle $\gamma(t)$ and instantaneous tool clearance angle $\alpha(t)$, relative to the X-axis are:

$$\gamma(t) = \gamma_0 + \kappa(t) \quad (2-19)$$
$$\alpha(t) = \alpha_0 - \kappa(t) \quad (2-20)$$

where $\gamma_0$ and $\alpha_0$ are the tool rake and clearance angles, respectively. See Figure 2-3(b). EVAM cycles should be designed so that the tool flank face does not contact the workpiece on the downward portion of the elliptical toolpath. This means $\gamma(t_1) \geq 0$ with $t_1$ the time in the vibration cycle when the tool first enters the workpiece during the downcutting portion of the cycle. By inspection of the tool path, it is seen this is the location of the steepest angle of tool motion $\kappa(t)$, for the segment of the vibration cycle where the tool is in the workpiece.

The depth of cut $d$ can be defined as the distance between the uncut work surface and the bottom of the groove made by the diamond tool. See Figure 2-4. When the feed motion has a vertical component, $d$ will vary with time and hence with the position in the X direction, creating a tilted surface. For EVAM, the instantaneous depth of cut $d(t)$ is given by

$$d(t) = abs(Z_{\text{CEN}}(t)) - B' \quad (2-21)$$

With the offset distance $B'$ given by

$$B' = \frac{\tan(\zeta(t)) + \sqrt{\tan^2(\zeta(t)) + 4B^2}}{2} \quad (2-22)$$
$B' = \frac{\tan(\zeta(t)) + \sqrt{\tan^2(\zeta(t)) + 4B^2}}{2}$

**Figure 2-4.** Depth of cut for EVAM on an inclined surface

$B'$ is derived by finding the tangent point for the instantaneous direction of motion $\zeta(t)$ to the vibration ellipse with semi-major and semi-minor axes $A \times B$. In practice, where the surface slope varies continuously, the offset distance is found by a motion program planning method such as described in Chapter 6, rather than explicitly by employing Equation 2-22.

In Equation 2-21 the origin for the depth of cut coordinate, $Z = 0$, is located on the uncut surface of the workpiece where it intersects the vibration ellipse, rather than at an absolute reference point.
2.3 EVAM Cycle Parameters

The "upfeed increment" $F_{UP}$ and "horizontal speed ratio" $HSR$ are defined [8] as:

$$ F_{UP} = \frac{U}{f} \quad (2-23) $$

$$ HSR = \frac{U}{\omega \cdot A} \quad (2-24) $$

$F_{UP}$ is the horizontal distance between equivalent points on the tool vibration path for two successive cycles, and is also the horizontal distance traveled by the ellipse center in one vibration cycle. $HSR$ is the ratio between upfeed velocity and the tool’s maximum horizontal vibration speed. Small values of $F_{UP}$ and $HSR$ mean that there is a large overlap between two successive elliptical cycles. As these parameters become larger the overlap grows smaller. Successive tool passes no longer overlap when $HSR > 0.318$.

Analogous parameters $F_{DEPTH}$ (depth feed index) and $VSR$ (vertical speed ratio) can be defined for vertical feed motion.

$$ F_{DEPTH} = \frac{V}{f} \quad (2-25) $$

$$ VSR = \frac{V}{B \cdot \omega} \quad (2-26) $$

$F_{UP}$, $F_{DEPTH}$, $HSR$, and $VSR$ are useful parameters for characterizing an EVAM cycle.
2.4 **Theoretical EVAM Surface Roughness**

Figure 2-5(a) depicts theoretical surface features generated by EVAM. The overlapping elliptical tool paths create a series of cusp-like features in the upfeed direction, as shown in Figure 2-5(c). The spacing of the cusps is $F_{UP}$. These features are superimposed on the crossfeed features created on successive raster passes by overlapping portions of the tool profile (Figure 2-5(b)), to create the pocked surface structure seen in Figure 2-5(a).

The theoretical surface roughness from crossfeed features, $PV_{CROSS}$, is controlled by the crossfeed spacing $\Delta Y$ and the tool nose radius $R_{tool}$, just as in conventional machining:

$$PV_{CROSS} \approx \frac{(\Delta Y)^2}{8 \cdot R_{tool}} \quad (2-27)$$

Cerniway adapted this equation to approximate the theoretical upfeed feature height, $PV_{UP}$. The surface feature height was assumed to be caused by the overlapping circular arcs corresponding to the maximum radius of curvature of the elliptical tool path, spaced at a distance of $F_{UP}$ (see Figure 2-5(c)). This resulted in Equation 2-28:

$$PV_{UP} \approx \frac{B \cdot U^2}{8 \left( A \cdot f \right)^2} = \frac{\pi^2 \cdot A}{2} \cdot (HSR)^2 \quad (2-28)$$

where the variables are the same as defined in Sections 2.2 and 2.3. Equation 2-28 is valid only for horizontal feed.
Negishi [39] predicted theoretical surface feature height, by numerical simulation of the toolpath, with horizontal feed velocity added to the elliptical vibration (Equation 2-11). $PV_{UP}$ was evaluated for a range of $HSR$ from 0.0004 to 0.04. These results showed that Equation 2-28 overestimates the surface feature height $PV_{UP}$. The discrepancy increases as $HSR$ gets larger.

Theoretical surface roughness estimates for sloped surfaces were investigated by numerical simulation of the tool tip motion, similar to Negishi’s approach. Appendix A describes a MATLAB model which is based on Equations 2-11 and 2-12. Figure 2-6 shows results for a
22 µm x 4 µm vibration ellipse \((A \times B = 11 \mu \text{m} \times 2 \mu \text{m})\), which was typically used in machining experiments later in this work. Theoretical surface roughness \(PV_{UP}\) is shown as a function of feed motion angle \(\zeta\) and upfeed increment \(F_{UP}\). The \(PV_{UP}\) surface roughness is defined here as the outward normal from the sloped upfeed surface.

This simulation shows two major results. Upward and downward slopes at the same feed motion angle produce the same theoretical \(PV_{UP}\) for small values of \(F_{UP}\). This is because the intersection between two successive cutting passes occurs at the same horizontal position, and the same vertical distance from the \(X\)-axis, whether the feed angle is \(+\zeta\) or \(-\zeta\).

Also, it can be seen that the theoretical roughness grows larger as the feed angle \(\zeta\), for any constant value of \(F_{UP}\). The roughness grows more severe at an accelerating rate relative to the rate of change of \(\zeta\). This result confirms the need to employ very small values of \(F_{UP}\) and \(HSR\) to provide a very smooth surface finish on sloping surfaces. For example, at constant \(F_{UP}\), the surface roughness for a 15º angle is more than 5 times that of a horizontal surface. To assure the theoretical roughness of this slope is smaller than an arbitrary value, the selected \(F_{UP}\) needs to be about 30% of what might would used for a horizontal surface. (These proportions apply only to the selected ellipse size and aspect ratio, and separate simulations must be run to examine other ellipse geometries.)

Vibration in the machine structure, and control system dynamic behavior, prevent these theoretical finishes from being achieved, and machined surfaces are usually several times rougher. Although the theoretical surface finish is the same for a slope of +/- \(\zeta\), the tool rake face angle will be more negative on an upslope and more positive for a downslope. This will result in much different machining forces, when cutting uphill or downhill, with consequent effects on tool life, surface finish and form error.
Figure 2-6. Theoretical PV surface roughness, for a vibration ellipse with semi-major axis $A = 11 \, \mu m$ and semi-minor axis $B = 2 \, \mu m$ as a function of upfeed increment $F_{UP}$ and feed motion angle $\zeta$. Bottom graph is detailed view for $F_{UP} < 2 \, \mu m$. 
3 EQUIPMENT DESCRIPTION AND IMPROVEMENTS

3.1 ULTRAMILL-NANOFORM SYSTEM

Brocato integrated the Ultramill EVAM tool with a Rank-Pneumo Nanoform 600 3-axis diamond turning machine [6]. This system was used to raster-cut binary microstructures such as the Angstrom symbol and "Thunderbird" described in Section 1.4.

Performance limitations emerged in these early parts which led to equipment modification for the current work. There were four significant issues:

- Ultramill temperature instability, which could increase form error. This was addressed by conversion to a closed-circulation cooling configuration for the piezoelectric actuators, from the original open-circulation arrangement.

- Millimeter-scale uncertainty in positioning the tool tip on the workpiece led to installation of a high-magnification video microscope system.

- Poor effective Y-axis encoder resolution of only 5 µm, due to electrical noise and lack of bidirectional repeatability.

- Low Z-direction stiffness of the original air-bearing Y-axis, causing significant surface roughness of 15-25 nm RMS.

The last two problems were corrected by replacing the air-bearing Y-axis with a hydrostatic oil-bearing axis with nanometer-resolution laser scale encoder.
These upgrades were executed incrementally, producing six distinct system configurations as shown in Table 3-1. Each configuration possessed different performance capabilities, and in some cases, different dynamics affecting machining accuracy. Configuration I-A was the initial Ultramill-Nanoform system used to make binary microstructures. Configuration IV is the final arrangement used to make the millimeter-scale optical surfaces described in Chapters 8 and 9.

Appendix B gives the basic operating sequence for cutting a part with the Ultramill-Nanoform system.

Table 3-1. Ultramill-Nanoform System Configurations

<table>
<thead>
<tr>
<th>Configuration</th>
<th>I-A</th>
<th>I-B</th>
<th>II</th>
<th>III-A</th>
<th>III-B</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Date</td>
<td>Aug 04</td>
<td>Jul 05</td>
<td>Sep 05</td>
<td>Apr 06</td>
<td>Jan 07</td>
<td>May 07</td>
</tr>
<tr>
<td>Y-Axis Bearing</td>
<td>Air</td>
<td>Air</td>
<td>Hydrostatic Oil</td>
<td>Hydrostatic Oil</td>
<td>Hydrostatic Oil</td>
<td>Hydrostatic Oil</td>
</tr>
<tr>
<td>Y-Axis Drive</td>
<td>Ballscrew</td>
<td>Ballscrew</td>
<td>Linear Motor</td>
<td>Linear Motor</td>
<td>Linear Motor</td>
<td>Linear Motor</td>
</tr>
<tr>
<td>Piezo Stack Cross Section</td>
<td>Triangular</td>
<td>Half-Round</td>
<td>Half-Round</td>
<td>Half-Round</td>
<td>Triangular</td>
<td>Triangular</td>
</tr>
<tr>
<td>Ultramill Cooling</td>
<td>Open Circulation</td>
<td>Open Circulation</td>
<td>Open Circulation</td>
<td>Open Circulation</td>
<td>Open Circulation</td>
<td>Closed Circulation</td>
</tr>
<tr>
<td>Touchoff Camera</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Y-axis Camera</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
3.2 **ULTRAMILL EVAM TOOL**

Cerniway [8] describes the Ultramill concept and Negishi [39] provides the detail design of the Ultramill EVAM tool used in the current work.

Figure 3-1 shows how the Ultramill's elliptical tool-tip motion is produced by a pair of piezoelectric actuators. Sinusoidal voltage signals, 90 degrees out of phase, are supplied to the parallel actuators. The toolholder functions as a mechanical linkage to convert the reciprocating motion of the actuators to an elliptical tool tip motion. The ellipse dimensions can be varied by changing the amplitude and phase difference of the voltage signals applied to the actuators. This system has operated at frequencies from 100 Hz to 4 kHz.\(^3\)

\[^3\] The first natural frequency of the piezoelectric stacks is approximately 12 kHz which would allow operation at 9.6 kHz with appropriate power electronics.
The flexibility to operate over a range of frequencies and ellipse geometries differentiates the Ultramill from "resonant" actuator designs which create elliptical motion by causing the toolholder structure to vibrate at a single discrete natural frequency [5, 31, 50, 52, 53, 57].

Figure 3-2 is a cutaway view showing the physical arrangement of the Ultramill. The two piezoelectric stack actuators are housed in a steel cooling chamber, through which dielectric fluid continuously circulates to remove waste heat generated by the actuators. Cooling system improvements are described in Chapter 4. Ceramic half-round pins, cemented to the top faces of the stacks, serve as pivot points for the hollow alumina toolholder. A titanium diaphragm provides preload force on the stacks and also provides a fluid seal around the toolholder.

Two actuator stack geometries were used in the Ultramill: the original triangular cross section, and half-round stacks with the same height and width. The half-round stacks were used because of an overheating failure in the first set of triangular stacks, but were later replaced when new triangular stacks came available. Figure 3-3 shows the two stack types.

![Figure 3-2](image-url)

Figure 3-2. Cutaway view of the Ultramill, showing its functional components.
while Table 3-2 compares their mechanical and physical characteristics. Triangular stacks are preferable because they have smaller mass (meaning a smaller preload requirement), smaller electrical capacitance (smaller electrical power requirement and waste heat generation), and larger ratio of surface area to mass (increased cooling effectiveness per unit power input).

**Figure 3-3.** Piezoelectric stacks used in the Ultramill. *(left)* Triangular cross section. *(right)* Half-round cross section. In both photos, the ceramic half-round pivot pins are not yet installed.

**Table 3-2.** Piezoelectric Stack Actuator Characteristics

<table>
<thead>
<tr>
<th>Stack Geometry</th>
<th>Triangular</th>
<th>Half-round</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass (per stack)</td>
<td>34.6 g</td>
<td>44.4 g</td>
</tr>
<tr>
<td>Natural Frequency</td>
<td>12.0 KHz</td>
<td>12.2 KHz</td>
</tr>
<tr>
<td>Required Preload (both stacks)</td>
<td>236 N</td>
<td>303 N</td>
</tr>
<tr>
<td>Capacitance (per stack)</td>
<td>0.25 μF</td>
<td>0.35 μF</td>
</tr>
<tr>
<td>Cross Section Area (per stack)</td>
<td>121 mm²</td>
<td>190 mm²</td>
</tr>
<tr>
<td>Convection Surface Area (per stack)</td>
<td>1,290 mm²</td>
<td>1,434 mm²</td>
</tr>
</tbody>
</table>

*A At 4000 Hz and 600 V_PP actuation voltage  *B Side faces + 1 end face
The piezoelectric stacks are driven by two parallel amplifier sets [39]. Each amplifier set drives one stack and consists of a power amplifier wired in parallel with a DC power supply. A two-channel signal generator supplies sinusoidal voltage signals with an operator-specified phase difference of 0° to 90°. The power amplifier boosts the sinusoidal voltage signal by a factor of 100, providing sufficient voltage to actuate the stacks. To prevent the actuators from being exposed to negative voltages which could cause the piezoelectric materials to depole, the DC power supply adds a positively-biased voltage to the sinusoidal voltage from the power amplifier. A routine check after installation of the second set of triangular stacks showed that the waveform produced by one of the amplifiers was badly distorted at frequencies above 2000 Hz. Signal generator waveforms were checked on an oscilloscope and found to be correct, indicating that the distortion was caused by a problem in the power amplifier. Following this discovery, Ultramill operations were restricted to a maximum frequency of 1500 Hz, with most parts machined at 1000 Hz. It is unclear whether this amplifier had failed at some point or if it had deficient throughout this work.

Appendix C describes how the tool ellipse was measured by cutting a plunge groove and then measuring the profiles of the tool entry and exit paths. This was investigated as an alternative to the optical method employed by Negisishi, which required taking the Ultramill from service to install a reflector target on the toolholder [39].

### 3.3 Diamond Tools and Toolholder

Round-nosed and sharp-nosed single-crystal diamond tools were used in this research. Most of the machining experiments were conducted with round-nosed tools such as the one shown in Figure 3-4(a). A sharp-nosed tool is shown in Figure 3-4(b). Such tools are needed to make features with steep, straight sidewalls. They differ from the round-nose tools in that the cutting edges are sharply inclined to the uncut work surface, and the nose radius ranges from “dead sharp” (less than 100 nanometers) to a few micrometers.
Figure 3-4. SEM images of single-crystal diamond tools. (a) Round-nose geometry. Tool shown has 1 mm nose radius. (b) Sharp nose tool. Tool shown is "dead sharp" with a nose angle of approximately 70˚ between the cutting edges.

Figure 3-5. Comparison of two toolholder types used with the Ultramill. (left) Original PEC design (right) Panasonic (PBL) design, with 1 mm nose radius diamond tool installed.

Figure 3-5 shows two versions of the alumina ceramic toolholder used in the Ultramill. The toolholders are hollow to minimize the oscillating mass. On the left is the original PEC design [39]. A less expensive design is shown on the right, which was obtained through Panasonic Boston Laboratory (PBL). The pivot-pin groove spacing is 11 mm for the PEC design and 10 mm for the Panasonic type. The shorter groove spacing of the PBL toolholder changes the length of the linkage base increasing the major axis of the toolpath ellipse by 10%. The PBL toolholder also has a flat, thinner base section causing high stresses near the pivot pin groove, making it more likely to fail during tool crashes or machining upsets.
3.4 **Nanoform Diamond Turning Machine**

3.4.1 Description

The Nanoform diamond turning machine (DTM) provides precision XYZ axis motion for machining micro-scale features. The X and Z axes have hydrostatic oil bearings with ballscrews and use laser interferometers to measure axis position. The original air-bearing Y-axis was also driven by a ballscrew but used a linear-scale encoder for position measurement (Configuration I-A/B). Control commands are provided to the axes by a Delta Tau UMAC controller. The spindle is locked when machining parts by EVAM, with the spindle vacuum chuck used to hold the workpiece. A videomicroscope camera mounted to the Z-axis provides a magnified view of the diamond tool’s rake face, to facilitate touchoff.

Figure 3-6 shows the Ultramill-Nanoform system after installation of a new hydrostatic oil-bearing Y-axis (Configuration II onward). This arrangement was used for the current work. This Y-axis has a linear electric motor drive and laser scale encoder with interpolation for position indication at a resolution of 1.1 nm. The right hand picture in Figure 3-6 also shows the Y-axis camera used to obtain a plan view of the workpiece for precise placement of the tool tip at desired XY coordinates (Configuration III-A/B and IV). In this arrangement the Ultramill is mounted to the axis carriage through the rear end of its base block via a carrier plate as shown in Figure 3-7.
**Figure 3-6.** Ultramill installed on Nanform DTM, with oil-bearing Y-axis. *(left)* Prior to installation of Y-axis camera (Configuration II). *(right)* Y-axis camera installed (Configuration III-A/B). Sideview touchoff camera removed in this view.

**Figure 3-7.** Ultramill bolted to Y-axis carrier plate.
3.4.2 Y-Axis Upgrade

The Angstrom symbol and Thunderbird logo (Section 1.4) were machined with the Nanoform’s original air-bearing Y-axis (Configurations I-A/B). It was theorized that vibration of this Y-axis, in combination with Z-axis vibration, caused the significant surface roughness measured in these parts. Brocato [6] investigated how axis vibration affected surface finish of parts made with the air-bearing Y-axis. Spatial frequencies were determined for upfeed surface profiles on test flats. The Z, X, and spindle amplitudes were found to be much less significant than those for the Y-axis, by almost an order of magnitude. The two most significant frequency components in the machined surface were very close to the resonant Z-direction frequencies, which were determined for the Y- and Z-axes by exciting the machine structure using a mechanical shaker.

The air-bearing Y-axis was replaced with an axis from Moore Nanotechnology Systems. The Moore axis uses a hydrostatic oil bearing for the slideway and has a linear motor drive instead of a ballscrew. With encoder interpolation carried out by the UMAC controller, the oil-bearing Y-axis can be programmed for step sizes of 1.1 nm, compared to 40 nm theoretical resolution for the air-bearing axis (noise problems with the encoder actually limited the air-bearing axis to micrometer-range resolution). The stiffness of the oil-bearing axis in the Z-direction is stated by the manufacturer to be approximately 2.5 MN/m. This compares with Brocato's experimentally-derived estimate for the air bearing axis stiffness of 500,000 N/m in the Z-direction. Table 3-3 shows the improvement in upfeed surface roughness obtained by switching to the stiffer oil-bearing axis.
Table 3-3. Upfeed surface finish comparison between air-bearing and hydrostatic oil-bearing Y-axis

<table>
<thead>
<tr>
<th>Y-Axis</th>
<th>Air-bearing</th>
<th>Hydrostatic Oil-bearing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration</td>
<td>I-A</td>
<td>II</td>
</tr>
<tr>
<td>Material</td>
<td>Hard-plated copper</td>
<td>Hard-plated copper</td>
</tr>
<tr>
<td>Part Description</td>
<td>small angstrom symbol [3]</td>
<td>test groove</td>
</tr>
<tr>
<td>Tool Nose Radius</td>
<td>.050 mm</td>
<td>1.0 mm</td>
</tr>
<tr>
<td>Machining Conditions</td>
<td>22 μm x 4 μm elliptical path, 1 kHz vibration frequency, 0.5 mm/s upfeed velocity</td>
<td></td>
</tr>
<tr>
<td>Upfeed Finish, PV</td>
<td>91 nm</td>
<td>48 nm</td>
</tr>
<tr>
<td>Upfeed Finish, RMS</td>
<td>19 nm</td>
<td>8 nm</td>
</tr>
</tbody>
</table>

Figure 3-6 shows how this Y-axis is mounted on a bolted steel plate riser assembly which in turn is mounted to the Z-axis carriage. The riser assembly positions the midpoint of Y-axis travel at the centerline of the spindle. Shearing stiffness of the bolted riser assembly was estimated by finite element analysis. For an eccentric load applied to the top plate (corresponding to a force transmitted by the Y-axis), the longitudinal stiffness (Z-direction) of the riser assembly is 6.3 MN/m and the transverse stiffness (X-direction) is 1.6 MN/m.

The dynamic response of the oil-bearing Y-axis was investigated using the Ultramill as an excitation source. Accelerometers were installed on the moving slide (Accelerometer 1) and on the axis frame (Accelerometer 2), oriented to sense the Z-direction acceleration. Figure 3-8 shows the accelerometer locations. The measurements were made with half-round piezo stacks in the Ultramill and no Y-axis camera (Configuration II). With triangular stacks the exciting force would be smaller, by the ratio of the triangular stack mass to that of the half-rounds. Acceleration data was collected by oscilloscope with the Ultramill operating at a frequency of 1000 Hz and with toolpath ellipse dimensions of 22 μm x 4 μm. Z-direction displacement of the Y-axis was not measured, because it was expected to be smaller than 50 nm which was the resolution of the available capacitance gauge system.
Figure 3-8  Accelerometer locations for Y-axis testing.

Figure 3-9. Oscilloscope response, Accelerometer 1, for oil-bearing Y-axis with Ultramill operating at 1000 Hz.
Figure 3-9 presents the oscilloscope signal for Accelerometer 1, showing Z-direction acceleration of the Y-axis slide. Acceleration peaks are visible every 0.001 s, corresponding to the 1000 Hz Ultramill frequency. The peak acceleration amplitude is seen to be 0.32 G. For a sinusoidal response, displacement can be obtained by twice integrating the acceleration:

\[
\begin{align*}
\ddot{z}(t) &= -\omega^2 \cdot A \cdot \sin(\omega t) \\
\dot{z}(t) &= -\omega \cdot A \cdot \cos(\omega t) \\
z(t) &= A \cdot \sin(\omega t)
\end{align*}
\] (3-1)

where \(A\) is the maximum displacement amplitude, \(f\) is the operating frequency of the Ultramill vibration in Hz, \(\omega = 2\pi f\) is the angular frequency, and \(z(t)\) and its first two time derivatives are displacement, velocity, and acceleration in the Z direction.

For the observed maximum acceleration in Figure 3-9, the peak displacement using Equation 3-1 is 81 nm. However it is obvious from the oscilloscope plot that multiple frequencies are interacting, that the acceleration deviates considerably from a pure sinusoid, and that it would be difficult to calculate the axis displacement response from the acceleration response. The displacement amplitude found using Equation 3-1 is therefore at best an indicator of the approximate displacement magnitude.

Accelerometer 2, attached to the Z-axis frame as shown on Figure 3-8, recorded accelerations of less than 0.010 G when the Ultramill was operating, or less than 3% of that recorded on the slide. The hydrostatic oil bearing is therefore demonstrated to be highly effective at absorbing vibration between the slide and axis frame.

The impulse response of the axis was checked by tapping the slide with a small hammer, while the Ultramill was turned off. From the accelerometer response (not shown here) the Z-direction natural frequency of the oil-bearing Y-axis was estimated to be 8 to 10 kHz.
3.4.3 Video Zoom Cameras

Two video-microscope cameras installed on the Ultramill-Nanoform are visible in Figure 3-6. A side view microscope camera provides a magnified view of the diamond tool to facilitate touchoff. A plan view camera is installed on the Y-axis to give a view of the workpiece surface. It can be used to position the tool tip at precise X-Y positions on the workpiece. Appendix D gives details on these cameras as well their use in touchoff and machining operations.
4 ULTRAMILL THERMAL STABILITY IMPROVEMENT

4.1 INTRODUCTION

4.1.1 Ultramill Temperature Variation

The Ultramill's piezoelectric actuators require active cooling by a circulating fluid to maintain a safe operating temperature below 100° C. Pressure to move the coolant through the system is produced by a positive-displacement diaphragm pump in the Thermocube temperature control unit. The diaphragm pump induces pressure pulsations in the circulating fluid, which can make the coolant supply hose vibrate. If the coolant supply hose is physically connected to the cooling chamber this vibration is transmitted to the Ultramill, possibly causing error in machined surfaces. The original design therefore used an open-circulation cooling scheme to isolate the Ultramill from the coolant supply hose vibration (System Configurations I-A/B, II, and III-A/B, as defined in Table 3-1).

Actuator temperatures for this cooling arrangement, obtained using a temperature sensor attached to the inside face of one of the stacks, showed non-stable behavior [39]. Frequent abrupt fluctuations in excess of 1°C were noted, as well as irregular upward or downward trends on a time scale of tens of minutes interspersed with periods of relatively constant stack temperature. Such temperature variations can cause form error by changing the tool tip position in the depth of cut direction, from thermal expansion or contraction of the Ultramill structure. The temperature instability is believed to be caused by low coolant flow rates and by formation and breaking of bubbles inside the Ultramill cooling chamber which disrupts the coolant flow pattern, reduces convection heat transfer from the stacks, and causes variations in the flow rate. The poor temperature stability of open-circulation cooling motivated investigation of a closed-circulation cooling arrangement. The goals of this work were to:
1. Reduce the temperature of the Ultramill by increasing the effectiveness of the heat transfer process.
2. Improve the temporal stability of the Ultramill temperature.
3. Minimize Ultramill vibration induced by the diaphragm pump.

4.1.2 Heat Generation by Piezoelectric Actuators

The Ultramill's piezoelectric stack actuators generate heat as a result of hysteresis during their expansion/contraction cycle. Equation 4-1 gives a relationship between input electrical parameters and heat production that is applicable to most stack designs [39, 42]:

\[
Q_{STACK} = \left( \frac{C \cdot f \cdot V_{pp}^2}{2} \right) \cdot K
\]

(4-1)

\(Q_{STACK}\) is the thermal power generated in a piezoelectric actuator stack, \(C\) is the stack capacitance, \(f\) is the actuation frequency, and \(V_{pp}\) is the peak-to-peak voltage magnitude applied to the actuator. \(K\) is the fraction of input power converted to heat for a given actuator stack. Negishi [39] experimentally established \(K = 0.24\) for the triangular actuator stacks. Table 4-1 uses Equation 4-1 to estimate \(Q_{STACK}\) for the Ultramill’s actuators, for a range of operating conditions.
Table 4-1. Estimated Piezo Stack Thermal Output Using Equation 4-1 and $K=0.24$
(Total for 2 Stacks)

<table>
<thead>
<tr>
<th>Triangular Stacks</th>
<th>Frequency</th>
<th>Voltage</th>
<th>Frequency</th>
<th>Voltage</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1000 Hz</td>
<td>1500 Hz</td>
<td>2000 Hz</td>
<td></td>
</tr>
<tr>
<td>Voltage</td>
<td></td>
<td>400</td>
<td>9.6 W</td>
<td>14.4 W</td>
<td>19.2 W</td>
</tr>
<tr>
<td>500</td>
<td>15.0 W</td>
<td>22.5 W</td>
<td>30.0 W</td>
<td></td>
<td></td>
</tr>
<tr>
<td>600</td>
<td>21.6 W</td>
<td>32.4 W</td>
<td>43.2 W</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Half-Round Stacks</th>
<th>Frequency</th>
<th>Voltage</th>
<th>Frequency</th>
<th>Voltage</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1000 Hz</td>
<td>1500 Hz</td>
<td>2000 Hz</td>
<td></td>
</tr>
<tr>
<td>Voltage</td>
<td></td>
<td>400</td>
<td>13.4 W</td>
<td>20.1 W</td>
<td>26.8 W</td>
</tr>
<tr>
<td>500</td>
<td>21.0 W</td>
<td>31.5 W</td>
<td>42.0 W</td>
<td></td>
<td></td>
</tr>
<tr>
<td>600</td>
<td>30.2 W</td>
<td>45.3 W</td>
<td>60.4 W</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2 OPEN- AND CLOSED-CIRCULATION COOLING

Figure 4-1 shows schematically the initial open-circulation Ultramill cooling system. The coolant is 3M Fluorinert FC-3283, a perfluorinated dielectric heat transfer liquid that can be used safely in direct contact with the actuators at the high voltages required by the Ultramill [65]. Coolant circulation is supplied by the Thermocube's diaphragm pump. The Thermocube is a self-contained unit designed to control the temperature of a circulating fluid to within ±0.1°C of a user-specified setpoint. Its thermoelectric heat exchanger adds or removes heat from the coolant at a rate up to 300 W. During steady-state operation, fluid leaving the Thermocube is cooled below the setpoint (return) temperature and increases in temperature as it passes over the actuator stacks. The volumetric flow rate is adjusted using a ball-float flow controller. A pressure reducing valve, downstream of the flow controller, improves the controller resolution by taking up most of the pressure difference between ambient and the pressure provided by the pump. The coolant discharges into a sight glass.

4 Manufactured by Solid State Cooling, Inc.
5 The Thermocube is available with a continuous-flow centrifugal pump, but this model is incompatible with the Fluorinert coolant used by the Ultramill.
mounted above the Ultramill cooling chamber. It then flows by gravity through the cooling chamber, around and between the piezoelectric actuator stacks, and into the coolant return line. From the Ultramill, the coolant drains through the return line back to the Thermocube. To facilitate gravity-driven flow, in this configuration the Thermocube is located on the floor, 1.2 m below the Ultramill center line.

Figure 4-2 shows the closed-circulation cooling configuration. In this arrangement the coolant supply line is physically connected to the bottom of the Ultramill cooling chamber. The volumetric flow rate of the coolant is again controlled by a ball-float flow controller with downstream pressure reducing valve. The coolant return line is attached to the top port of the cooling chamber. In the closed-circulation configuration, the Thermocube is located above the cooling chamber discharge to assist any air bubbles in the coolant circuit to flow away from the Ultramill.
Figure 4-1. Open-circulation cooling arrangement.

Figure 4-2. Closed-circulation cooling configuration.
4.3 **FORM ERROR IMPROVEMENT WITH CLOSED-CIRCULATION COOLING**

Figure 4-3 illustrates the improvement in form error obtained by using closed-circulation cooling. The figure shows crossfeed profiles obtained by white-light interferometry for the "potato chip" feature described in Chapter 7. This part is a typical feature with sculpted 3-D geometry. It took 55 minutes to machine, so the crossfeed profiles contain the effects over this time period of temperature-induced dimensional change of the Ultramill along the depth-of-cut direction. Open-circulation cooling (left panel) shows considerable deviation as great as 500 nm peak-to-valley (PV) from the desired part profile. This error is reduced to less than 80 nm PV for closed-circulation cooling (right panel).

**Figure 4-3.** Crossfeed profiles for "potato chip" part described in Chapter 7, obtained by white light interferometry. (*left*) Profile for open-circulation cooling shows form error because of Ultramill temperature variation. (*right*) Closed-circulation cooling nearly eliminates form error by improving temperature stability.
4.4 **EXPERIMENT SETUP**

A comprehensive set of experiments was conducted to obtain a quantitative understanding of the Ultramill cooling system performance. Appendix E gives a detailed presentation of the instrumentation setup used for these tests. A brief description is provided here to facilitate the discussion.

Temperature measurements were obtained at important locations shown in Figure 4-4. Thermocouples inserted into the fluid stream through tee-fittings measured the inlet and outlet coolant temperature (TC2, TC3). The base block core temperature (TC8) was measured by a

![Figure 4-4. Thermocouple installation locations on Ultramill.](image-url)
thermocouple inserted into a hole drilled to the Ultramill centerline. At the other locations, the thermocouples were attached to the component surface with polyimide-base tape. Most important of these was TC6, attached to the inside face of one of the piezo stacks.

Two methods were used to assess thermal expansion of the Ultramill along the depth of cut direction, which could cause drift in the tool tip position and consequent error in the machined surface:

- Tool-tip position was measured directly using a capacitance gauge. The diamond tool was replaced on the ceramic toolholder by an aluminum target with approximately the same mass. Figure 4-5(a) is a video-microscope view of the target and the capacitance gauge, which was fixed to the X-axis of the diamond-turning machine. The capacitance gauge was sampled at 20 kHz and measured the instantaneous distance from the target along the depth of cut direction as illustrated by Figure 4-5(b). This data included the effect of the toolholder's elliptical motion on the target position. The average target position was obtained by filtering out the elliptical motion with a 200-point rolling average of the instantaneous position data. Changes in the average target position showed the effect of Ultramill thermal expansion or contraction in the depth of cut direction.

- Raster-machined flats were cut to determine the impact of thermal transients on form error and surface roughness.

Temperature and capacitance gauge position measurements were recorded using LabVIEW Version 7 running on a dedicated PC. Appendix F gives the LabVIEW block diagram.
Figure 4-5. (left) Video-microscope picture of capacitance gauge and aluminum target installed on toolholder in place of diamond tool. (right) Target motion relative to capacitance gauge

4.5 ULTRAMILL COOLING SYSTEM EXPERIMENTS

4.5.1 Overview

Ultramill thermal behavior was compared for the open-circulation and closed-circulation cooling configurations. Key performance indicators were the temperature stability of the Ultramill structure (evaluated by the stack skin temperature measured by thermocouple TC6), and variation of tool tip position in the depth of cut direction. All experiments were executed at the following conditions:

- Thermocube setpoint of 20°C.\(^6\)
- triangular piezoelectric stacks in the Ultramill (Configurations III-B and IV in Table 3-1)

\(^6\) The Thermocube has its own PID controller which adjusts the thermoelectric heat exchanger's output to maintain a specified setpoint fluid temperature. Negishi [39] found that controlling the return coolant temperature led to more stable coolant temperatures in the Ultramill as well as a faster response to upsets or changes in setpoint.
• stack operating conditions of 1000 Hz and 400 $V_{PP}$, giving a total stack thermal generation of 9.6 W (calculated using Equation 4-1)

### 4.5.2 Coolant System Flow Rate

Coolant flow was measured by the in-line Dwyer Instruments ball-float flow controller. The flow controller was factory-calibrated for water. A calibration for the Fluorinert dielectric coolant was obtained by measuring the coolant pumped into a graduate cylinder during a specified period at a selected setting on the flow controller. The coolant rate thus calculated was compared to the flow-controller setpoint to obtain a calibration ratio. A 10-test average yielded an actual coolant flow of 0.42 L/min for 1 L/min indicated flow rate [43]. All coolant flow rates given hereafter are the actual rates estimated using this calibration.

In open-circulation cooling, the flow rate was limited by the available gravity head. During operation the flow was adjusted to maintain a constant liquid level in the sight glass. Open-circulation coolant flow rates varied between 0.30 and 0.45 L/min. The flow variation was caused by changes in the return circuit flow resistance, believed to be mainly due to bubble formation or dispersal in the Ultramill cooling chamber.

### 4.5.3 Energy Balance Test

Energy balance tests were performed to establish confidence in the Ultramill temperature measurements and to estimate the convective heat transfer coefficient between the Ultramill and the environment. The energy balance analysis, with sample results, is detailed in Appendix G. The balance was closed to within 10% of the estimated stack heat generation.

These results showed that convection from the external surfaces of the Ultramill can be ignored as it is very small compared to the stack heat generation and the heat removed by the coolant.
4.5.4 Temperature Tests

Figure 4-6 shows temperature data recorded during Ultramill operation with the gravity-driven open-circulation cooling configuration. The coolant rate varied between 0.30 and 0.40 L/min as adjustments were made to keep a constant liquid level in the sight glass. Figure 4-7 shows temperature data for the closed-circulation configuration with coolant flow maintained at 0.40 L/min for the first part of the run, then reduced to 0.16 L/min.

The top plot in each figure shows the temperature measured by TC6, the thermocouple installed on the inside face of the piezoelectric actuator. The polyimide tape attaching it to the stack may not fully insulate the sensor from the circulating coolant. This temperature is best considered to be "indicative" of the actuator thermal behavior, rather than a precise measurement of the stack surface temperature. The middle plots show temperatures for the solid steel base block at TC8, 9 mm behind the piezo stacks ("block core") and also TC10 at the end farthest away from the piezo stacks ("block end"). Coolant temperatures at the inlet (TC3) and outlet (TC4) of the Ultramill cooling chamber appear in the bottom plot.

With open-circulation cooling the indicated stack temperature fluctuated across a range of several degrees, with several abrupt changes. In contrast the stack temperature for closed-circulation cooling showed significantly smaller variation when the coolant flow was 0.40 L/min. The temperature itself was also lower, indicating better convection heat transfer between the stacks and the circulating coolant. At 0.40 L/min flow rate the indicated stack temperature fluctuated cyclically with a period of approximately 3.3 minutes and a 0.4°C peak-to-peak magnitude. The coolant temperatures also cycled with a slight phase difference from the stack temperature. This period apparently reflects the natural frequency of the cooling system arrangement (characterized by such parameters as length and volume of the coolant hoses, cooling chamber volume, and Thermocube control system dynamics). When the closed-circulation coolant rate was reduced to 0.16 L/min this cyclic behavior became more
Figure 4-6. Ultramill temperatures for open-circulation cooling configuration

Figure 4-7. Ultramill temperatures for typical machining test using closed-circulation cooling. Coolant flow reduced from 0.4 L/min to 0.16 L/min at 38 minutes’ elapsed time.
pronounced. The coolant temperature variation increased to 1.1 °C and the period grew to 7 minutes. Although the indicated stack temperature cycled with the same period as the coolant temperature, it also showed less regularity and fluctuated over a larger range. For this low flow case, the base block temperature varied cyclically with the same period as the stack and coolant temperatures, but with a noticeable phase lag. This indicates there is significant thermal interaction between the base block and the coolant and also possibly the actuator stacks. Temperature variation in the base block is undesirable since it is a source of thermal expansion/contraction in the depth of cut direction.

The coolant temperature oscillation is assumed to drive the cyclic temperature behavior in the stacks and base block. These oscillations are caused by lag in the Thermocube response when compensating for deviations from the return coolant temperature setpoint. When the Thermocube commands a change to the coolant supply temperature, the results are delayed and hence the control setpoint is continuously overshot. Small coolant flow rates exacerbate this behavior by increasing the lag time. The lag time is found by dividing the cooling system volume (hoses plus cooling chamber) by the coolant flow rate. Table 4-2 summarizes the lag

### Table 4-2. Effect of Coolant Flow Rate on Ultramill Temperature Oscillations

<table>
<thead>
<tr>
<th>Case</th>
<th>Open Circulation</th>
<th>Closed Circulation &quot;Low&quot; Flow</th>
<th>Closed Circulation &quot;Normal&quot; Flow</th>
<th>Closed Circulation &quot;High&quot; Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow Rate</td>
<td>L / min</td>
<td>0.30 - 0.45</td>
<td>0.16</td>
<td>0.4</td>
</tr>
<tr>
<td>Total Time Lag a</td>
<td>sec</td>
<td>35 - 52</td>
<td>87</td>
<td>35</td>
</tr>
<tr>
<td>Temperature Cycle</td>
<td>min</td>
<td>NA</td>
<td>6</td>
<td>4.5</td>
</tr>
<tr>
<td>Stack Temp Variation</td>
<td>°C RMS</td>
<td>10</td>
<td>2.1</td>
<td>0.3</td>
</tr>
<tr>
<td>(TC6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Block Core Temp</td>
<td>°C RMS</td>
<td>&lt;0.1</td>
<td>0.2</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>Variation (TC8)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coolant Temp</td>
<td>°C RMS</td>
<td>0.4</td>
<td>1.0</td>
<td>0.4</td>
</tr>
<tr>
<td>Variation (TC3, 4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*a Time lag for coolant to circulate from supply source at Thermocube, through Ultramill and return to Thermocube.*
times, period of the coolant temperature cycle, and magnitude of Ultramill temperature variations for open- and closed-circulation cooling. Table 4-2 and Figure 4-7 show that a significant coolant flow of 0.40 L/min is needed to maintain temperature stability in the Ultramill.

4.5.5 Long-Duration Temperature Stability

Figure 4-8 shows Ultramill temperature data for a 3.5-hour long machining run, executed with the closed-circulation configuration at a coolant rate of 0.40 L/min. Once the Ultramill reached thermal equilibrium after approximately 50 minutes, the temperature for all measurement locations showed minimal variation of less than 0.2 °C. This result demonstrates the long-term temperature stability which can be achieved by the closed-circulation cooling arrangement.

![Figure 4-8](image)

**Figure 4-8.** Ultramill temperatures recorded during 3-hr long machining run. After block temperatures TC8 and TC10 reach steady state at 80 minutes, they show variation of less than 0.2°C.
4.5.6 Capacitance Gauge Tests

Tool-tip drift caused by Ultramill thermal expansion was measured using a capacitance gauge as described in Section 4.4. Cap gauge testing has two advantages compared to cutting test surfaces: 1) the test duration can be set independent of the part size and 2) the capacitance gauge data is continuous in time rather than compromised by the raster scan pattern that may partially obliterate one machining pass with subsequent crossfeed passes.

Figure 4-9 shows results for an open-circulation test of approximately 50 minutes duration. The top panel shows Ultramill temperatures including the stack temperature while the bottom panel shows the cap gauge target position relative to its location when the Ultramill is powered down. Positive values for target position reflect motion toward the capacitance gauge sensor, equivalent to Ultramill thermal expansion in the direction of greater depth of cut. The cap gauge position reached steady-state at an elapsed time of approximately 12 minutes and then tracked with the indicated stack temperature, TC6. The total change in the cap gauge target position from starting the test was 15 µm, showing that it is essential for the Ultramill to reach thermal equilibrium before starting a cutting experiment.

Figure 4-10 presents results for closed-circulation cooling. The coolant rate was initially 0.40 L/min, but was later changed to 0.80 L/min and then reduced to 0.16 L/min. The temperature and cap gauge position plots in Figure 4-10 show how the Thermocube took several minutes to respond to a change in coolant rate, with adverse effect on both coolant and Ultramill temperatures. For example when coolant rate was increased to 0.8 L/min at approximately 46 minutes elapsed time, it took more than 10 minutes for the cap gauge position and Ultramill temperatures to return to steady-state. This shows the coolant flow rate should be kept constant when using closed-circulation cooling, since the changes in coolant rate cause an upset in tool tip (cap gauge target) position.
Figure 4-9. Open-circulation capacitance gauge test

Figure 4-10. Closed circulation capacitance gauge test.
Table 4-3 summarizes the variation in stack skin temperature and cap gauge target position for the tests shown in Figures 4-9 and 4-10. The peak-to-valley (PV) and RMS variations were obtained from the latter portion of each test segment, after steady-state was achieved following a change in conditions. In closed-circulation cooling, stack temperature and cap gauge position show improved stability with increasing coolant flow rate. At the normal (0.4 L/min) and high (0.8 L/min) flow rates the RMS cap gauge variation is smaller than for the open-circulation cooling configuration.

<table>
<thead>
<tr>
<th>Case</th>
<th>Coolant Rate</th>
<th>Data Duration</th>
<th>Stack Skin Temperature Variation</th>
<th>Cap Gauge Position Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L / min</td>
<td>min</td>
<td>°C PV</td>
<td>°C RMS</td>
</tr>
<tr>
<td>Open circulation</td>
<td>0.30 to 0.45</td>
<td>25</td>
<td>1.64</td>
<td>0.36</td>
</tr>
<tr>
<td>Closed circulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Low&quot; flow</td>
<td>0.16</td>
<td>10</td>
<td>2.31</td>
<td>0.48</td>
</tr>
<tr>
<td>Closed circulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Normal&quot; flow</td>
<td>0.40</td>
<td>25</td>
<td>.32</td>
<td>.05</td>
</tr>
<tr>
<td>Closed circulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;High&quot; flow</td>
<td>0.80</td>
<td>10</td>
<td>0.31</td>
<td>0.05</td>
</tr>
</tbody>
</table>

### 4.5.7 Machined Flats

Flats 1 mm long x 0.5 mm wide were raster-machined in hard-plated copper while collecting Ultramill temperature data. The tool nose radius was 1 mm. The crossfeed spacing was 5 µm. The upfeed velocity was 0.5 mm/s and the vibration frequency was 1000 Hz ($HSR=0.007$). These parts each took 8.2 minutes to cut. Three flats were cut using open-circulation cooling and 4 with closed-circulation cooling: two at a flow rate of 0.4 L/min, one at 0.8 L/min and one at 0.16 L/min.
Figure 4-11 plots the form error for the flats against the RMS variation in piezo stack skin temperature for the cutting period. The Zygo New View white-light interferometer was used to measure the form error of a rectangular area 0.5 mm long x 0.4 mm wide in the central region of the flat. The form error measurements covered approximately 6.5 minutes of machining time. The long sample period means the form error caused by Ultramill temperature instability are included as well as surface roughness, which is generated on a much shorter time scale and by mechanisms other than thermal instability.

Figure 4-11 shows that closed-circulation cooling did not improve the area-based form error in these flats. This differs from the cap gauge results in Table 4-3 which showed that variation in tool tip position gets progressively smaller with increasing coolant flow. At a flow rate of 0.40 L/min the form error was the same as for open-circulation cooling, although with closed-circulation cooling the temperature variation while making the part was much smaller. At a low flow rate of 0.16 L/min the form error was twice as large as at 0.4 L/min and also that of open-circulation cooling. This was because significant coolant an temperature variation caused dimensional variation of the piezo stacks and base block. The form error for high coolant flow (0.80 L/min) is also worse than at 0.40 L/min.

Figure 4-12 shows surface roughness for the flats, also obtained using white-light interferometry. Each data point in the figure is the arithmetic average of 3 roughness measurements taken at random locations. The sample areas were 0.013 mm long x 0.017 mm wide. In the crossfeed direction the sample areas included 4 adjacent cutting passes spanning 15 seconds of machining. In the upfeed direction the samples covered 0.013 sec. The short time covered by these surface roughness measurements mean they include the effects of the EVAM process and machine dynamics but not of long-period temperature variations. The surface roughness values are plotted against the same values of stack RMS temperature variation as in Figure 4-11.
Figure 4-11. Form error based on a 0.5 mm long x 0.4 mm wide area in the central region of the rastered flat.

Figure 4-12. Surface roughness for rastered flats, obtained from 0.017 mm x 0.013 mm measurement areas (each point is average of RMS surface roughness at three locations).
For closed-circulation cooling, roughness grew worse with increasing coolant flow. At coolant rates of 0.4 L/min and smaller, surface roughness was the same as for open-circulation cooling, 8 to 12 nm RMS. At 0.8 L/min, closed-circulation roughness was 20 nm RMS. The increase in surface roughness between 0.40 L/min and 0.80 L/min is believed to be caused by larger pump-induced pressure pulsations at the higher coolant rate (10 psi compared to 7 psi at the normal flow rate, obtained from pressure gauges on the coolant supply and return hose). The pressure pulsations affect surface roughness in two ways. They cause vibration of the Ultrastructure structure. Also, the pulsations are hypothesized to cyclically change the preload exerted by the Ultramill’s titanium diaphragm, which would cause the compressive strain and hence length of the piezoelectric actuators to also vary cyclically.

Table 4-4 shows the effect of subtracting the RMS roughness from the RMS form error results for the closed-circulation test flats. The resulting “corrected” form error is the magnitude of the form error caused by the Ultramill thermal instability. It can be seen that the corrected form error of the flats decreases monotonically with increasing coolant flow, a behavior similar to that of the cap gauge tests. This suggests that the form error improvements expected by

<table>
<thead>
<tr>
<th></th>
<th>Low Flow</th>
<th>Normal Flow</th>
<th>High Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coolant flow (L/min)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Form error a (nm RMS)</td>
<td>38</td>
<td>17</td>
<td>25</td>
</tr>
<tr>
<td>Surface roughness b (nm RMS)</td>
<td>8</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>Corrected form error c (nm RMS)</td>
<td>30</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>Cap gauge variation d (nm RMS)</td>
<td>122</td>
<td>48</td>
<td>21</td>
</tr>
</tbody>
</table>

a From Figure 4-12 b From Figure 4-13 c Roughness subtracted from form error d See Table 3-3
increasing coolant flow were offset in the test flats by the greater surface roughness caused by the positive-displacement diaphragm pump. In other words, as coolant flow increased the surface roughness grew worse faster than the form error improved, so that the white light measurement appeared to degrade at flow rates above 0.40 L/min.

4.6 SUMMARY OF ULTRAMILL COOLING SYSTEM EXPERIMENTS

Conversion of the Ultramill from open- to closed-circulation cooling produced these significant results:

- Closed-circulation cooling provided greater stack and coolant temperature stability when the coolant flow rate was maintained at 0.4 L/min or greater.

- Surface roughness is adversely affected by pressure pulsations from the Thermocube diaphragm pump. At a coolant flow of 0.4 L/min, surface roughness for both open- and closed circulation cooling was 8 to 11 nm RMS. Surface roughness in closed-circulation cooling became worse with increasing coolant flow, reaching 20 nm RMS at 0.8 L/min.

- With closed-circulation cooling, form error was equal to or better than that achieved with open-circulation cooling. Form error improved from 500 nm to 80 mm peak-to-valley, in 3-D parts which required an hour to machine. In rastered flats with an 8-minute machining time, form error of closed- and open-circulation was equal, but the higher surface roughness in closed-circulation cooling possibly obscured form error improvements.
The form error improvement gained by close-circuit improvement provides motivation for addressing surface finish degradation caused by the diaphragm pump. A continuous flow centrifugal pump eliminates the pressure pulsations caused by a positive displacement machine. The Thermocube is available with this type of pump, but its seal materials were incompatible with the Fluorinert coolant used by the Ultramill. After this work was finished, Lane [29] installed a pulsation dampening leg in the coolant tubing which provided sufficient surge volume and pressure head to suppress the pressure pulses from the diaphragm pump. While the pulsation-induced surface roughness was not eliminated, its amplitude was greatly reduced.
5 SINGLE GROOVE MACHINING EXPERIMENTS

Single-groove cutting experiments were performed to assess the surface finish achieved with the Nanoform's hydrostatic oil-bearing Y-axis, and to compare open- and closed-circulation cooling.

Table 3-1 defines in detail the system configurations referenced in this chapter.

5.1 PREVIOUS ULTRAMILL SURFACE ROUGHNESS RESULTS

Negishii made flat surfaces in PMMA using the Ultramill at EVAM frequencies from 1000 to 4000 Hz [39]. For this work the Ultramill was installed on a lathe-style diamond-turning machine, which possessed different vibration dynamics from the Nanoform. For HSR between 0.01 and 0.1, the measured height of surface features along the upfeed direction was within 10% of that expected from his numerical simulations.

Brocato [6] raster-machined flats in plated copper at $f = 1$ kHz and $HSR = 0.022$, using the Ultramill-Nanoform combination and the air-bearing Y-axis (Configuration I-A). The measured upfeed surface roughness was approximately 60 nm peak-to-valley (PV) as compared to a theoretical roughness of 6 nm PV predicted by Equation 2-27. Low-frequency Nanoform axis vibration was evident in the surface profiles and was a major contributor to roughness. Individual features in these profiles were approximately 12 nm PV, or twice as tall as the theoretical roughness.

In these initial experiments the surface was created by successive crossfeed passes. This limited understanding of upfeed feature formation and of the EVAM cutting process since each pass erased part of the surface made on the preceding pass. Phasing effects between adjacent passes possibly further modified the surface features created on earlier passes [8, 39].
5.2 SINGLE-GROOVE CUTTING EXPERIMENT

Single-groove test parts were cut after switching to the hydrostatic oil-bearing Y-axis (Configuration II), after the Y-axis camera was installed (Configuration III-B), and following conversion of the Ultramill to closed-circulation cooling (Configuration IV). By machining only single grooves, the features created by EVAM were isolated from crossfeed effects.

Figure 5-1 shows the single-groove test part. This layout enabled upfeed surface roughness to be assessed as a function of upfeed velocity $U$ and depth of cut $d$. For each trial, a square foreground was first cut by the Ultramill, giving a reference surface into which a series of grooves was then machined. Cutting features into a machined surface eliminated groove-to-groove depth variations arising from topography or tilt in the original workpiece surface. Upfeed velocity $U$ varied from 0.2 mm/s to 8 mm/s. The groove machining time ranged between 0.4 s ($U = 8$ mm/s) and 15 s ($U = 0.2$ mm/s), short enough for temperature changes in the Ultramill to be ignored. At each upfeed velocity, grooves were machined at depths of cut of 1, 2 and 2.5 μm. The tool nose radius for these tests was 1 mm, which created a relatively flat groove cross section.

7Table 3-1 describes the Ultramill-Nanoform system configurations used in this work.
5.3 SINGLE-GROOVE ROUGHNESS RESULTS

5.3.1 Open-Circulation Cooling

The initial set of single-groove parts were cut with the Ultramill cooled by the open-circulation method (Configuration II, as described in Table 3-1). With no pump-induced vibration or coolant pressure pulsations, surface features were caused only by the EVAM process and by Nanoform dynamics.

Three parts were made, two in hard-plated copper and one in 1100-alloy aluminum. Vibration frequency $f$ was 1000 Hz. Table 5-1 shows machining conditions and material properties for these parts. The first copper part (designated Part 60)$^8$ and the aluminum part (Part 61) were

---

$^8$ Approximately 180 parts were made in the course the dissertation research. These included setup grooves, touchoff points, and failed experiments, in addition to completed test parts.
machined, after which the tool was relapped due to concern that it might not be sharp. The final copper part (Part 62) was cut after the tool was relapped.

Surface profiles along the groove centerlines were obtained using a Zygo New View white light interferometer. The measurements were made at the interferometer’s maximum magnification which gave a 72 μm long profile. The surface roughness was calculated by the Zygo software, leveled to the best-fit plane. For each groove, profiles were taken at 4 random locations and the individual roughness measurements averaged. This minimized the effect of local particulate contamination or surface pitting on the calculated roughness, and also averaged out any low frequency surface variation with wavelength longer than the profile.

<table>
<thead>
<tr>
<th>Table 5-1. Machining Conditions and Material Properties for Groove Parts Machined using Open-Circulation Cooling</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Part ID</strong></td>
</tr>
<tr>
<td>Frequency, f (Hz)</td>
</tr>
<tr>
<td>Ellipse Dimensions, μm</td>
</tr>
<tr>
<td>Upfeed Velocity</td>
</tr>
<tr>
<td>$F_{UP}$</td>
</tr>
<tr>
<td>HSR</td>
</tr>
<tr>
<td>Tool Status</td>
</tr>
<tr>
<td>Material</td>
</tr>
<tr>
<td>Hardness</td>
</tr>
<tr>
<td>Elastic Modulus</td>
</tr>
<tr>
<td>Ductility</td>
</tr>
</tbody>
</table>
Figures 5-2 through 5-5 show RMS surface finish compared to upfeed velocity $U$ for several test cases.\textsuperscript{9} Also plotted is the theoretical RMS feature height derived from Equation 2-27.

Figure 5-2 shows RMS surface roughness results for hard-plated copper (Part 60) and 1100-alloy aluminum (Part 61) with the tool in its initial condition—this was assumed to be moderately worn. At any upfeed velocity, surface roughness was unexpectedly much greater than the theoretical value based on the EVAM parameters. For an upfeed velocity of 4 mm/s or smaller, the surface roughness was the same for both materials. However for $U > 4$ mm/s the surface roughness in the softer aluminum was slightly smaller than for the plated copper.

Figure 5-3 shows the effect on surface roughness of relapping the diamond tool. This was done to rule out excessive tool wear or cutting edge damage as a cause of large surface roughness. Grooves made in hard-plated copper with the lapped tool showed a 20 to 30 percent reduction in roughness for $U > 2$ mm/s but even after sharpening, roughness still remained much greater than the theoretical value.

\textsuperscript{9} Normally it is most useful to relate Ultramill results to characterization parameters such as upfeed index $F_{UP}$ or horizontal speed ratio $HSR$ (see Chapter 2). In these experiments all grooves were machined using the same size elliptical vibration path, and vibration frequency. The results are plotted against upfeed velocity $U$. to avoid creating an impression that they are valid for other combinations of machining parameters giving the same $HSR$.
Figure 5-2  Roughness Results - Parts 60 (Hard-Plated Copper) and 61 (1100 Aluminum)

Figure 5-3  Roughness Results before (Part 60) and after lapping the tool (Part 62).
The tool cutting edge radius was not measured due to the difficulty of obtaining an accurate value for this dimension, which on a sharp diamond is smaller than 50 nm. Therefore the tool's initial condition was unknown, as well as the sharpness improvement gained by relapping. The tool manufacturer inspected the cutting edge using high-power optical microscopy, before and after the relapping. This could not measure the edge radius, but allowed the manufacturer to make a qualitative evaluation that the tool was in good condition before it was refurbished. The small amount of improvement in surface finish therefore may indicate that the tool was relatively sharp from the start. If true, this means the excessive roughness must be considered to be characteristic of the EVAM process at large values of $U$, $F_{UP}$ or $HSR$.

Figure 5-4 shows the effect of depth of cut on RMS roughness. The material was hard-plated copper and the relapped tool was used (Part 62). For upfeed velocity smaller than 2 mm/s, surface roughness was unaffected by depth of cut. At greater values of $U$ the roughness was almost exactly the same for 1 and 2 µm cutting depth. However the surface roughness became worse when depth of cut was increased to 2.5 µm. Here the depth of cut $d$ was greater than the semi-minor axis $B$ of the toolpath ellipse ($B = 2$ µm, $d/B = 1.25$). For this condition, EVAM produces long continuous chips instead of small discrete chips [8, 39].

---

10 Drescher [16] describes a method to estimate the cutting edge radius by "painting" reference marks on it using the beam of a field-effect scanning electron microscope. It is tedious and requires the tool and toolholder to be removed from the Ultramill, which was not possible in these experiments due to time limitations.
5.3.2 Open-Circulation Cooling with Y-Axis Camera

Grooves were cut in hard-plated copper (Part 148) after the new plan-view camera was installed on the Y-axis. The camera and its support strut approximately doubled the Y-axis payload compared to the Ultramill alone. When this part was cut, the Ultramill had also been converted back to triangular stacks (Configuration III-B).

Figure 5-5 plots the RMS surface roughness for this case along with a part made before the Y-axis camera was installed (Part 60). The diamond tool was not pristine as it had been used for machining experiments immediately before the camera installation, so the figure compares roughness against a non-camera groove made before the tool was sharpened.
For $U \leq 1$ mm/s, the surface roughness was the approximately the same (10 to 12 nm RMS) regardless of the camera installation. However for upfeed velocity greater than 1 mm/s, the surface roughness was noticeably worse with the Y-axis camera. It is conjectured that the roughness deteriorated because adding the Y-axis camera adversely affected the Nanoform's vibration dynamics. This is partly due to the added mass of the camera assembly, which decreases the natural frequency of the oil-bearing Y-axis slideway for oscillation in the Z-direction. Also the camera is cantilevered at the end of its support strut, and vibrates in response to Z-direction accelerations. The reaction of the camera strut acts as an excitation to the Y-axis carriage further complicating its Z-direction motion.

![Figure 5-5](image_url)  
**Figure 5-5** Surface roughness – Y-axis camera installed (Part 148) vs. no camera (Part 60)
5.3.3 Closed-Circulation Cooling

In closed-circulation cooling (Configuration IV)\(^\text{11}\), pressure pulsations from the Thermocube's diaphragm pump are an additional source of surface roughness. Figure 5-6 compares the roughness of grooves cut using closed-circulation cooling (Part 158) at coolant flow rate of 0.4 L/min, to the results for open-circulation cooling (Part 148 and Part 62). For upfeed velocities less than 1 mm/s, closed-circulation and open-circulation cooling produced approximately the same surface roughness. This is consistent with the results presented in Figure 4-12 which showed that in rastered flats made at similar values for \(U\) and \(f\) (0.5 mm/s and 1000 Hz), the two cooling configurations gave about the same area-based roughness. However for \(U > 1\) mm/s the roughness obtained with closed-circulation cooling was considerably worse than for open-circulation cooling.

\(^{11}\) See Table 3-1
Figure 5-6. Surface roughness comparison between closed-circulation cooling with Y-axis camera (Part 158), open-circulation cooling with Y-axis camera (Part 148), and open-circulation cooling only (Part 62).

5.4 Profiles of EVAM Features

5.4.1 Open-Circulation Cooling

Figure 5-7 shows surface profiles for grooves cut in hard-plated copper using open-circulation cooling and the freshly-lapped tool (Part 62). These profiles were plotted using MATLAB, from data obtained by the Zygo New View white light interferometer. Figure 5-7 shows profiles for grooves made with upfeed velocities of 8, 4, 2, 1, and 0.2 mm/s. All five profiles are plotted at the same horizontal scale. The vertical scale of the last two profiles (1 and 0.2 mm/s) is exaggerated by a factor of two for clarity.
For upfeed velocity from 8 mm/s down to 2 mm/s (Figures 5-7(a)-(c)), the cusp-like EVAM surface features are clearly visible, and are spaced periodically at the predicted upfeed increment $F_{UP}$. Their height decreases as $F_{UP}$ grows smaller. The features have a repetitive appearance. As shown on Figure 5-3 the surface roughness of the EVAM features in these profiles is several times greater than the expected value based on Equation 2-27.

In the last two profiles, for $U$ of 1 and 0.2 mm/s, periodic EVAM cusps can no longer be distinguished (Figure 5-7(d)-(e)). For example, in Figure 5-7(d), over a 10 μm distance ten features should be visible at a regular spacing of 1 μm. But it is difficult to even identify the features created by EVAM. In Figure 5-7(e) at $U = 0.2$ mm/s the spacing of the expected EVAM features is approximately at the resolution limit of the interferometer and there is a random appearance to variations in the surface profile.

These profiles explain how the surface roughness varies with upfeed velocity in Figures 5-2 through 5-6. When $U$ is large, for example 8 mm/s, the features are tall, large and regular as in Figure 5-7(a). This results in a large value for surface roughness. As upfeed velocity is decreased (and $F_{UP}$ and HSR), the features become shorter and more closely spaced and the surface roughness grows smaller as well (Figure 5-7(b) and (c)). Below 2 mm/s, the features created by EVAM can no longer be detected and the surface profiles are irregular with small amplitudes (30 to 60 nm PV or 8-12 nm RMS) that do not appear to be affected by the upfeed velocity (Figures 5-7(d) and (e)).

As noted earlier the magnitude of EVAM feature heights is significantly larger than predicted by Equation 2-28 and this may be related to the process itself, rather than caused by factors such as material hardness or tool wear condition. Appendix H investigates the possibility that the “minimum chip thickness” effect [17] might be the source of these features. Although
Figure 5-7. Upfeed profiles at 8, 4, 2, 1, and 0.2 mm/s for part made in plated copper using freshly-lapped tool (Part 62). Open-circulation cooling.
cutting experiments were not performed, simulations suggest that, with certain assumptions about material behavior in the EVAM process, the minimum chip thickness effect remains a possible explanation for these features.

### 5.4.2 Closed-Circulation Cooling

Figure 5-8 shows surface profiles for grooves cut in hard-plated copper using closed circulation cooling (Part 158). They are given for upfeed velocities of 2, 1, and 0.5 mm/s. These cases correspond to the last three open circulation profiles shown in Figure 5-7, except that here the smallest velocity is 0.5 mm/s instead of 0.2 mm/s. These three closed-circulation profiles are plotted to the same scale as the open-circulation profiles in Figure 5-7(c)-(e), i.e. the last two are plotted with a vertical resolution twice that of the 2 mm/s profile.

The closed-circulation surface features generated at these upfeed velocities are qualitatively different from those made with open-circulation cooling. At $U = 2$ mm/s, the features in Figure 5-8(a) are taller and more rounded than the open-circulation features at the same velocity in Figure 5-7(c). For closed circulation at upfeed velocities smaller than 2 mm/s (Figures 5-8(b), (c)), distinct regular features at a spacing consistent with the Ultramill machining frequency are no longer discernible. Small features spaced at $F_UP = 1$ µm/cycle are detectable in Figure 5-8(b), but they no longer possess the regular shape of features seen at larger upfeeds. These features are dominated by long wavelength variations with larger amplitude. Both of the low-velocity upfeed profiles in Figure 5-8 differ from the equivalent profiles for open-circulation cooling, in that they are more angular and there are fewer small amplitude closely spaced variations visible.
Figure 5-8. Profiles of grooves made in hard-plated copper with closed-circulation cooling. Vertical range on panel (a) is 2 times that of the other two panels.

5.5 Autocovariance Analysis of EVAM Features

In microstructure fabrication, individual axis movement segments might be only tens of micrometers long, and the acceleration limits of the X-axis can make it non-meaningful to program velocities faster than 1 mm/s. At a vibration frequency of 1000 Hz, this means the upfeed increment $F_{UP}$ is $1 \mu m/cycle$ or smaller. Factors affecting surface finish in this low velocity range need to be well understood for creation of high-quality surfaces by EVAM.
When cutting PMMA using the Ultramill, Negishi [39] found that for low values of HSR the surface roughness was dominated by vibration of the turning machine axes. Similarly, Shamoto [56] reported that the best achievable surface finish for his resonant EVAM actuator was 10-12 nm RMS due to vibration of its 5-axis turning machine under excitement by the actuator. It seems likely that in the current work, axis vibration of the Ultramill-Nanoform system affects the surface roughness at low values of upfeed velocity. The groove profiles in Figure 5-7(d) and (e) made with open-circulation cooling have the appearance of high-frequency, small-amplitude signals superimposed onto lower-frequency, large-amplitude vibrations.

Autocovariance plots generated by the Zygo New View white-light interferometer are shown in Figure 5-9. These plots are for grooves cut in hard-plated copper at upfeed velocities of 4, 2, 1, and 0.2 mm/s, with \( f = 1000 \) Hz. The plots on the left side are from a part machined using open-circulation cooling (Part 62) The corresponding plots on the right side are from grooves made with closed-circulation cooling (Part 169). The spatial wavelength \( \lambda \) of periodic features in the autocovariance plot can be converted to a time-domain frequency \( f_T \) by Equation 5-1:

\[
 f_T = \frac{U}{\lambda}
\]  

(5-1)

At \( U = 4 \) mm/s the autocovariance plots for both open and closed-circulation cooling are dominated by the periodic features made at the EVAM vibration frequency of 1000 Hz. This is consistent with the surface profiles at this velocity in Figures 5-7 and 5-8, where the features were regular, and spaced at \( F_{up} \). In the closed-circulation case, the autocovariance also includes a long period variation at 30 Hz which is the frequency of the diaphragm cooling pump. The 1000 Hz EVAM features are superimposed on this longer-wavelength variation.
Figure 5-9. Autocovariance plots for single-groove parts made with open-circulation (Part 62) and closed-circulation cooling (Part 169). See Table 3-1, and Figures 4-1 and 4-2, for description of the open-anc closed-circulation cooling arrangements.
At $U = 2$ mm/s the 1000 Hz EVAM frequency is still visible but it is superimposed on lower frequencies which have large amplitude. In the open-circulation case a 70 Hz variation can be seen. For closed-circulation cooling the pump operation at 29-30 Hz is the main source of upfeed surface height variation.

When upfeed velocity is 1 mm/s or less, long period variations completely dominate the autocovariance plots and the 1000 Hz frequency can no longer be seen. For closed-circulation cooling the 30 Hz pump pulsations remain the primary source of surface roughness. In the open-circulation case, a 10 Hz periodic pattern appears, replacing the 70 Hz variation seen at 2 mm/s. From the available data, it is not possible to establish whether this is caused by a 10 Hz vibration, the superposition of multiple other frequency sources or a beat-type phenomena (for example, a 990 or 1010 Hz vibration interacting with the 1000 Hz Ultramill frequency). The reason for the 70 Hz frequency at $U = 2$ mm/s is also unknown. The source and actual time-based frequency of such vibrations cannot be identified solely from the spatial frequency of machined features. Axis vibration would need to be measured directly, for example by using a capacitance gauge to measure displacement and accelerometers to measure the frequency response to an excitation source.

For closed-circulation cooling the pressure pulsations caused by the diaphragm pump dominate axis vibration, and are the principal cause of surface roughness when $U$ is less than 2 mm/s. Major improvements in surface finish can be expected by suppressing these pressure pulses, either by developing a pulsation damper for use with the diaphragm pump, or by switching to a continuous-flow pump with no pulsations.\textsuperscript{12}

\textsuperscript{12} The Thermocube was available with a centrifugal pump but its seal material was incompatible with the Fluorinert coolant used by the Ultramill.
As noted in Section 5.3, for upfeed velocity less than 2 mm/s the surface roughness was 8-10 nm RMS regardless of the cooling configuration. The autocovariance results for open-circulation coolings suggest that the Ultramill excites a vibration with a magnitude of ~10 nm RMS. When upfeed velocity is less than 2 mm/s ($F_{UP} = 2 \mu m/cycle$), the vibration amplitude is larger than the height of the individual EVAM features, and they are dominated by it. Therefore no improvement in surface finish occurs when upfeed velocity varies within this regime. When upfeed velocity is greater than 2 mm/s, the EVAM cusp features are tall enough that they exceed the tool-workpiece vibration. Since this behavior is independent of pump-induced roughness, it will persist after the diaphragm-pump pulsations are addressed. Suppressing turning machine vibration therefore appears to be an important strategy to improve surface finish in parts machined at small feed velocity.

5.6 SUMMARY

The single-groove cutting experiments produced these significant results:

- The actual machined RMS roughness significantly exceeded the theoretical roughness predicted from EVAM parameters (using Equation 2-28).

- With open circulation cooling, the surface roughness fell into two regimes. When the upfeed was greater than 2 mm/s ($F_{UP} = 2 \mu m/cycle$ at 1000 Hz), the RMS roughness increased with larger values of upfeed velocity. Below this velocity, the roughness finish remained in the range of 8 nm to 12 nm RMS nm, independent of the upfeed velocity.

- For closed circulation cooling, the surface roughness at upfeed velocities of 1 mm/s or less was 8-12 nm RMS, the same as for open circulation cooling. At greater upfeed velocities, the surface roughness was worse than for open-circulation cooling.
• For upfeed velocity $U \leq 1$ mm/s, with open-circulation cooling the surface was dominated by features at 10 Hz. These are believed to result from Nanoform axis vibration.

• With closed-circulation cooling, the diaphragm pump pulsation created features at 30 Hz which was the main cause of surface roughness at small upfeed velocities. Use of a continuous-flow centrifugal pump, or a pulsation dampener as was employed by Lane [29] should eliminate or at least sharply suppress the pump-induced vibration features.
6 MOTION PROGRAM PLANNING

6.1 INTRODUCTION

The Nanoform’s controller issues commands to drive the machine axes along the path needed to create a desired surface. Usually this path has a different geometry from the surface because of the offset required between the contact point of the tool, and the tool center. Finding the required motion path is the goal of motion program planning. For 3-D features raster cut by EVAM, this is a non-trivial task because of the elliptical tool vibration path.

6.1.1 Raster Machining of 3-D Surfaces

Figure 6-1 shows the sequence of axis moves for a single crossfeed pass when rastering a 3-D feature. At the start of the pass, the diamond tool is at crossfeed position Y and out of contact with the workpiece surface. The tool moves downward to contact the workpiece (motion 1) and then along the upfeed (X) direction to execute a cutting pass (motion 2). During the cutting pass the depth of cut (Z) can vary to create a contoured surface. At the end of the cutting pass, the tool moves vertically upward to disengage from the workpiece (motion 3). It then retracts back along the upfeed pass to the original position (motion 4), and steps incrementally in the crossfeed (Y) direction to a new starting position (motion 5). The cycle repeats, with the depth of cut varied along the new upfeed pass as needed to make the specified surface.
6.1.2 Cutter Compensation

Figure 6-2 shows a single raster pass drawn on a 3-dimensional feature. To make the desired surface using EVAM, the programmed motion of the turning machine axes must be offset from the trace of the pass in two directions. In the crossfeed direction (lower right of figure) the tool's cutting edge profile determines the required offset between axis position and surface. In the upfeed direction the axis motion must be adjusted based on the elliptical vibration path\textsuperscript{13} of the tool (lower left). Commercial CNC machining software such as Surfcam is unable to compensate for the EVAM vibration path due to the complex manner in which the tangent vector to the elliptical path varies with upfeed position.

\textsuperscript{13} Technically, the path taken by the tool tip from the elliptical EVAM and the superimposed vertical and upfeed motion. For small values of $H SR$ the deviation from the purely elliptical vibration path is negligible.
The vibration ellipse center moves on a path defined by the work feed motion (see Equations 2-13 through 2-16). The ellipse center is at a constant position relative to the Ultramill structure and the Y-axis carriage. Hence, the ellipse center moves parallel to the X-Y-Z axis path, at a constant offset distance.

Figure 6-3 shows how a convex volume is generated by sweeping the tool cutting edge profile through the lower half of the vibration ellipse. This volume can be visualized as a solid and hereafter is called a "probe". The center point of the probe’s horizontal upper surface corresponds to the center of the vibration ellipse. The motion of the probe center thus
corresponds to the motion produced by the diamond turning machine axes. Finding the points through which the probe center must pass to cut a specified surface is equivalent to generating the motion program required to make that surface.

For a known probe geometry, it is straightforward to construct the surface produced by a motion program. The inverse problem—finding the motion program to cut a desired surface—is usually more complex. On the right side of Figure 6-3 the probe is shown in contact with an arbitrary surface. At all points of contact the probe and the surface must be mutually tangent, otherwise the probe penetrates the surface. When the probe contacts the surface while maintaining the mutual tangency requirement, the XYZ coordinates of its centroid correspond to the position of the Nanoform axes required to create that portion of the surface containing the tangent point(s).
The information needed to write a motion control program can be obtained by envisioning the probe positioned in contact and tangent with the surface, successively at all locations which define the surface. This operation creates a corresponding offset surface comprised of all the points through which the probe center must pass to create the machined surface.

The main problem is verifying simultaneous tangency at all points of contact. When analytic expressions for the surface and probe are available, tangency can be checked at each contact point by simultaneous solution of the equations describing the gradients for the surface and probe. However “this becomes computationally intensive when a large number of surface locations must be considered. Also problems can arise when evaluating gradients numerically, especially when the surface and/or probe have irregular shapes described by piecewise continuous functions.” [59]

6.2 MORPHOLOGY METHOD

6.2.1 “Dilation” and “Erosion”

As previously described, the volume formed by sweeping the tool cross section through the elliptical vibration path can be thought of as a solid probe. Probing the desired surface to establish the offset surface defining the motion program is similar to making an image of a specimen using a contact probe microscope such as an Atomic Force Microscope (AFM). Villarrubia [59] discusses how the morphological operations “dilation” and "erosion" can be applied to contact probe microscopy. These morphology procedures use set operations which, for the large number of surface points needed to image a specimen at high resolution, are computationally more efficient than attempting to develop gradient information. The surface and the probe volume are expressed as arrays of varying elevations. This allows evaluation of irregular surfaces and probe geometries, for which analytic expressions might not exist.
Dilation is proved to be equivalent to the operation of forming an image of a surface using a probe with known geometry. Since this image surface is analogous to the offset surface obtained by probing, morphology methods provide a way to generate the information needed to write axis motion programs. More details are provided in the following sections.

Similarly, erosion is shown to be equivalent to reconstructing the surface, for a given image and probe. This provides a way to predict the surface which will be cut using a given motion program.

**Dilation- Equivalent to Finding the Required Axis Motions**

This explanation of dilation, and of erosion in the following section, is paraphrased from Villarubiaa [59]. To clearly illustrate these concepts, the diagrams here are drawn in two dimensions but can be extrapolated to the 3-D case.

Figure 6-4 shows the conventional method of constructing of a toolpath surface for a specified surface $S$, using a probe $P$. The probe dimensions are based on the elliptical tool path dimensions but do not include the effects of feed velocity relative to the workpiece. The probe is positioned above $S$. It is then moved toward the surface until it makes contact at one or more points; contact is defined by mutual tangency between the surface and probe. When contact occurs, the centroid of the probe indicates the offset surface height for that point on the horizontal axis. This operation is repeated for every point on $S$. The completed offset surface $J$ is the union of all the centroid locations obtained by probing every point on the desired surface. All of the raster passes needed to make the surface $S$ therefore lie in $J$. The offset surface can be sectioned at specified crossfeed coordinates to obtain individual raster passes.
Figure 6-4. Creating toolpath offset surface $J$ by "probing" a surface $S$.

Figure 6-5. Comparison of probe and structuring element.
Note how the notch on the left side of the sample surface in Figure 6-4 is too small to be resolved by the probe, and that the portion of $J$ associated with this feature hardly deviates from the adjacent level surface. This means that the notch is too small a feature to be machined with the specified EVAM ellipse.

Morphology methods treat the part as a set of points. The desired final machined surface $S$ is defined by the upper bound of the set. Although the position of any individual point in the upper bound of the set can be described by a function $Z = Z(X, Y)$, set operations are applied to these points, rather than analytical or arithmetic operations.

The morphology equivalent to the probe of Figure 6-4 is called a “structuring element”. The structuring element is also represented by a set of points with an upper bound. The outward normals to the sample surface $S$ and to the upper bound of the structuring element $SE$ both have the same sense.

In the case here, structuring element $SE$ is obtained by reflecting probe $P$ from Figure 6-4 around the X-Z origin. In other words, $SE(X, Z) = -P(-X, -Z)$. Figure 6-5 compares the depictions of a probe $P$ derived from an elliptical tool vibration path, and the corresponding structuring element $SE$.

“Dilation” generates an image surface from a sample surface (in this case the machined surface) using the structuring element. The vertex of the structuring element is placed on a point of the sample surface. The upper bound of the structuring element is established for this surface point. To create the image, the vertex of the structuring element is swept through every point on the surface. The image is the intersection of all the upper bounds of the structuring element when it is translated through all the points on the surface. Figure 6-6 illustrates dilation of the surface $S$ by the structuring element $SE$ to generate image surface $I$.  

95
The key point of this discussion is now reached: the image \( I \) made by dilating surface \( S \) with structuring element \( SE \) is identical to the offset surface \( J \) created by probing in Figure 6-4. Villarrubbia [59] combines arguments from set theory and analytical geometry to prove this proposition. The proof is relatively long and sophisticated, and is omitted here for brevity.

\[ \text{Figure 6-6. Creating image } I \text{ by dilating } S \text{ with structuring element } SE \]

Dilation therefore is a way to generate the coordinate information needed for a motion program. The structuring element is derived from the tool cutting edge cross-section and the elliptical vibration path. The image resulting from dilation corresponds to all the points through which the center of the probe must pass to create the desired surface. Raster passes are extracted from the image by sectioning it at specified crossfeed Y-coordinates. Sets of X-Z points for programming the actual axis commands can then be obtained from the raster pass data.
Erosion—Equivalent to Constructing the Surface Made by a Motion Program

Given a structuring element $SE$, surface $S^*$ can be reconstructed from an image $I$. This operation is known as "erosion". Figure 6-7 illustrates erosion of the image found by dilation in Figure 6-6. In erosion the structuring element is inverted relative to its orientation for dilation. The reconstructed surface $S^*$ is obtained by taking the union of the lowermost points of $SE$ when its vertex is translated through every point on the image.

The reconstructed surface $S^*$ found by erosion of the image $I$ is identical to the surface obtained by sweeping the centroid of the ellipsoidal probe $P$ through the offset surface $J$. Villarrubia [59] again provides the necessary proof.

**Figure 6-7.** Reconstructing a surface $S^*$, by eroding the image $I$ formed in Figure 6-6 using structuring element $SE$. The notch in the original surface does not appear in the reconstruction due to it being smaller than can be resolved by $SE$. 
Constructing a surface by eroding an image has at least two applications to motion program planning:

1. The desired surface can be checked to verify that all features can be resolved by the selected vibration path ellipse. For example, the notch on the left-hand side of surface $S$ in Figures 6-4 and 6-6 is too small to be resolved with the selected probe (structuring element). This is confirmed by eroding the image created in Figure 6-6 to produce the reconstructed surface, which differs from the original surface by not including the notch.

2. Erosion can be used to simulate the effect of programming errors, uncertainty in the tool profile, deviations of the vibration ellipse, and systemic error in the diamond turning machine axes. Consider the case of a round-nose tool with a radius slightly different than the value used to program the axis motion. An image is constructed by dilating the desired surface using a structuring element which is derived from the assumed tool radius. The actual surface is then found by eroding the image using a second structuring element, this time based on the actual tool radius. The original surface can be subtracted from the actual surface to estimate the form error caused by the tool inaccuracy.

### 6.2.2 MATLAB Program for Motion Program Generation

Villarubbia [59] provides algorithms for structural element generation, dilation, and erosion. These were incorporated into a MATLAB program for EVAM motion program generation, but it took an extremely long time to run (13 hours for a typical 1000 x 1000 point depiction of a part).
The MATLAB "Image Processing Toolbox" [32] includes functions for dilation (`imdilate`), erosion (`imerode`), and structuring element definition (`strel` and `nhood`). These were used to write a MATLAB program called MORPH3D which generates the image surface, raster pass data, and G-code controller commands for a motion program. MORPH3D program listings are given for the parts cut with EVAM in Chapters 7, 8, and 9 (Appendices J, K, L, O). Execution times for these programs were more manageable, 1 to 2 hours.

MORPH3D performs the following actions:

1. *Create a matrix representation of the surface to be machined.*
   The desired surface is plotted on a square X-Y grid, typically at 1 μm resolution. The elevation of the surface Z at each grid point is obtained either from an analytic function or by representing the surface as an array of height values. Typical surfaces contain hundreds of thousands to approximately 2 million points.

2. *Create a matrix representation of the structuring element*
   The structuring element is defined on an X-Y grid at the same resolution used for the surface, then finding the Z value each grid point. This is accomplished by first creating a function Z(Y) that describes the cutting edge of the tool profile (that is, the crossfeed variation of the profile height). A height modifier function ΔZ(X) is generated for each point in the upfeed direction path corresponding to the tool vibration path. The structuring element is defined by

   \[ Z(X,Y) = Z(Y) + ΔZ(X) \]  

   (6-1)

   which is equivalent to "sweeping" the tool cross section through the elliptical tool path. Figure 6-8 shows a typical structuring element created using MORPH3D, for a round nose tool with nose radius R of 1000 μm, and toolpath amplitudes of 11 μm x 2 μm (A x B) or an overall ellipse of 22 μm x 4 μm. The equation describing this structuring element is
\[ Z(X,Y) = \sqrt{R^2 - Y^2} + \sqrt{B^2 - \frac{B^2}{A^2} \cdot X^2 - R} \]  

(6-2)

where

\[ Z(Y) = \sqrt{R^2 - Y^2} \]

\[ \Delta Z(X) = \sqrt{B^2 - \frac{B^2}{A^2} \cdot X^2 - R} \]

Figure 6-8. Structuring element generated by MORHP3D program. Element shown here is based on Equation 6-2, using a 1 mm tool radius and 22 μm x 4 μm vibration ellipse.
3. **Dilate the surface with the structuring element to form the image**

The function `imdilate` is used to create the image $I$, expressed as an array of heights $I(X,Y)$.

4. **Extract individual raster passes from the image**

For constant crossfeed spacing $\Delta Y$, the $n$th raster pass is located at $Y_n = (n-1) \cdot \Delta Y$, where $n$ is a raster pass number between 1 and the total number of passes in the part. The coordinates of the $n$th raster pass therefore consists of all the values of the image along $Y=Y_n$, or

$$Z_{\text{RASTER}}(X,Y) = I(X,Y_n) = I(X,(n-1) \cdot \Delta Y) \quad (6-3)$$

5. **Linearize the upfeed passes**

Each raster pass is converted to a set of line segments approximating the curvilinear path extracted from the image. This is because the Nanoform's controller cannot calculate continuous curved trajectories, only linear moves between designated points.

6. **Write the G-code motion program from the linearized upfeed passes**

The final output of the MORPH3D program is a text file made up of G-code commands which can be used by the Nanoform’s controller.
6.3 **Morphology Method Validation and Accuracy**

6.3.1 **Morphology Method Validation**

The MORPH3D morphology program was validated by running it on test cases for which the theoretical image surface can be calculated analytically.

Figure 6-9 depicts the three validation cases:

- hemispherical probe on a concave spherical surface
- hemispherical probe on a convex spherical surface
- sinusoidal sheet probed by a round-nosed tool swept through an elliptical vibration path

**Hemispherical Probe on Spherical Concave Surface**

Figure 6-9(a) shows a cross section sketch of this test case. The test surface is a portion of a 500 μm radius concave sphere subtending an angle of 60 degrees. The structuring element\(^\text{14}\) is a hemisphere with 50 μm radius. The theoretical image is a spherical surface concentric with the desired surface but with a radius of 450 μm.

Figure 6-10 shows the deviation between the theoretical image and the image generated by the MORPH3D program. The deviation is shown along profiles parallel to the upfeed (X axis) direction. A positive deviation means the program’s image has a larger radius than the theoretical image. Deviation is shown along the spherical surface centerline, and also for the chordal section at the quarter point in the Y-direction one-fourth of the radius in from the crossfeed edge of the surface.

\(^{14}\) Technically, the examples in Figure 6-9 show a probe contacting the surface. However from this point forward "probe" and "structuring element" will be used interchangeably.
Figure 6-9. Validation cases for testing the MATLAB morphology motion program.
The high-frequency component in the deviation plots is not fully understood but seems to be inherent in the dilation process. It appeared in the output from the initial (slow running) program based on Villarubbia’s algorithm (see page 92) as well as the program based on the MATLAB `imdilate` function. Its cause is conjectured to be truncation error and/or discretization error caused by how the MATLAB function represents the surface and structuring element as arrays of height values rather than continuous functions. Figure 6-10(b) shows the centerline and quarter-point deviation for a tighter grid resolution of 500 nm. The morphology program deviation at this resolution is smaller than 4 nm. This indicates that as the model's resolution improves to more closely approximate a continuous surface, the difference between theoretical and program image surfaces approaches zero.

The deviation between morphology and analytic models increases near the end points of the X-axis. These locations are regions of increasing slope in the hemispherical test surface. As result the tangent angle between the test surface and the hemispherical test surface in these regions is also increasing. With increasing slope, vertical changes are distributed across fewer of the (horizontal) points in the discretized X-Y array, leading to a loss of resolution in the image surface derived by the morphology method. This means the morphology image can deviate from the analytical surface by a greater amount.

Referring back to Figure 6-10(a), it can be seen that along the centerline the minimum value for the deviation is zero. However along the quarter-point profile the minimum value of the deviation is a small positive number (about 3 nm), shown as \( \Delta \). In the high-resolution case in Figure 6-10(b) and in the convex surface case (Figure 6-12) the quarter-point deviation profile has a minimum value of zero. In Figure 6-11 the minimum deviation value \( \Delta \) is plotted for multiple crossfeed positions for the concave part from the to near the edge. It can be seen that \( \Delta \) varies between zero and a few nanometers, depending on the crossfeed position. Thus the quarter-profile offset in Figure 6-10(a) is due to the fortune in selection of the location of the section rather than being an something unique to the particular case.
Figure 6-10. Accuracy of morphology method for concave spherical surface probed by hemispherical structuring element. “Deviation” is the difference in Z coordinate between the theoretical and program-generated image surface.
Hemispherical Probe on Spherical Convex Surface

This test case is depicted in Figure 6-9(b). The test surface is a section of a 500 μm radius convex sphere subtending an angle of 30 degrees either side of the centerline. The structuring element is again a hemisphere with 50 μm radius. The theoretical image (offset surface) is a spherical surface surface with a radius of 550 μm concentric with the desired surface.

Figure 6-12(a) shows the deviation between theoretical image and that generated by the MATLAB program for 1 μm resolution, while Figure 6-12(b) shows the deviation for a 500 nm grid. Once more the deviation is shown along the centerline and quarter section profiles.
Figure 6-12. Accuracy of morphology method for convex spherical surface probed by hemispherical structuring element. "Deviation" is the difference in Z coordinate between the theoretical and program-generated image surface.
The maximum deviation at any location is greater than for the concave surface. The results at
tighter resolution again show that deviation can be reduced by using a smaller grid size.
However even at 1 μm resolution the maximum deviation is less than 50 nm, or a figure error
of about 1/10th wave.

**Ellipsoidal Probe on Sinusoidal Sheet**

The final validation case is a sinusoidal sheet, shown in cross section in Figure 6-9(c). The
structuring element is derived for an elliptical tool vibration path with a round tool cross
section, as described by Equation 6-1. This is similar to the structuring element that would be
used for real machining projects. The theoretical image surface was found by matching the
gradients of the elliptical vibration path with those of the sinusoidal surface. Figure 6-13 shows
the deviation between the program's predicted image surface and the theoretical image surface,
with 1 μm resolution used for the surface and probe. The maximum deviation is approximately
15 nm, or smaller than 1/40 wave figure error. Maximum deviation occurs at the inflection
points of the sinusoidal surface, where the slope is greatest. This is consistent with behavior
of the spherical test cases in Figures 6-10 and 6-12, where the deviation increased with
increasing slope, but still remained within a small fraction of a wavelength of light.

**Summary of Validation Results**

The validation cases confirm that as grid size is reduced in the MORHP3D morphology
program, the calculated image surface approaches the theoretical image surface. A 1 μm grid
gave deviations from theoretical image surface of 15 nm to 50 nm, corresponding to a figure
error of 1/40 to 1/10 wave. The MORPH3D result should converge to the theoretical solution
at an infinitesimal grid spacing. However increasing the grid resolution causes a corresponding
large increase in program execution time.
6.3.2 Sensitivity to Upfeed Velocity and to Sloped Surfaces

The MORPH3D program generates the structural element by sweeping the tool profile along a circular or elliptical vibration path, producing an ellipsoid. This is a simplification compared to actual machining, where the tool path includes horizontal and vertical components motion in the upfeed (X-direction) motion. The feed motion results in a different structuring element shape from that produced using a pure elliptical vibration path.
Figure 6-14 shows how upfeed and vertical feed motion distort the vibration path from an ellipse. Since the structuring element is derived from the lower half of the vibration cycle (when the tool can be in contact with the workpiece, the elongation caused by upfeed motion is \( F_{UP}/2 \), and the horizontal length of the actual toolpath is \( 2A + F_{UP}/2 \). Similarly, with constant velocity in the depth of cut direction, the vertical position of the end of the vibration path is displaced from the starting point by half the depth feed increment, or \( F_{DEPTH}/2 \). Between these end points the vibration path shape must be calculated using Equations 2-11 and 2-12 for time \( t, \ 0 \leq t \leq T/2 \).

**Figure 6-14.** Distortion of elliptical vibration path caused by upfeed and vertical feed motion.
For most part geometries, the surface slopes will vary continuously, and so will the tool path shape. It is impractical to develop a continuously varying structuring element so the MORPH3D program takes the approach of using a constant, purely ellipsoidal element in a horizontal orientation. Error in the machined part shape is introduced by the difference between the actual tool path and the use of a constant structuring element.

Recalling Section 6.2.1, dilation creates the toolpath needed to produce a specified surface; erosion generates the surface resulting from a given toolpath. The magnitude of the error between true and idealized structuring elements was assessed by dilating a test surface using a structuring element based on a pure elliptical vibration path. The toolpath (image surface) was then eroded using alternate structuring elements derived from a vibration path which included feed motion. Figure 6-15 shows the test case. The desired surface was an up-and-down ramp with 10° inclination. The down ramp angle was based on the clearance angle for diamond tools used in subsequent fabrication experiments. The up ramp used the same angle for symmetry. The structuring element used to create the image assumed a 1 mm nose radius round-nosed tool driven in a 22 μm x 4 μm elliptical path ($A \times B = 11 \times 2$) at 1000 Hz. For the erosion operation, separate structuring elements were developed for the upward and downward directions.

Table 6-1 and Figure 6-16 shows the resulting vertical deviation between the desired and machined surfaces. A positive deviation means that the machined surface was cut deeper than desired. In Figure 6-16(a), the upfeed velocity was 4 mm/s ($HSR=0.058$). Excluding the endpoints, where the straight desired surface deviates from the curved transition due to the curved vibration path profile, the distortion is 178 nm on the uphill side or about 1/4 wave. Figure 6-16(b) shows deviation when upfeed velocity is reduced to 0.5 mm/s ($HSR=0.008$). Here the maximum deviation shrinks to 20 nm.
These results show that feed motion does cause form error by distorting the tool path from a pure ellipse. The amount of form error depends on $F_{UP}$ and $F_{DEPTH}$. For the 10° ramp case, when $F_{UP}$ is less than 0.5 μm / cycle, the form error caused by the actual toolpath is of the same magnitude as surface roughness and can be ignored for motion program planning.

![Diagram showing apparent feed motion and desired surface](image)

**Figure 6-15.** Test case for checking sensitivity to machining tilted surfaces

<table>
<thead>
<tr>
<th>Table 6-1. Error caused by structuring element distortion on sloped surface</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Upfeed Velocity, $U$</strong></td>
</tr>
<tr>
<td>HSR</td>
</tr>
<tr>
<td>$F_{UP}$ μm / cycle</td>
</tr>
<tr>
<td>$F_{Depth}$ μm / cycle</td>
</tr>
<tr>
<td>Form error, uphill nm</td>
</tr>
<tr>
<td>Form error, downhill nm</td>
</tr>
</tbody>
</table>
Figure 6-16. Deviation between desired and actual surface produced for tilted surface case shown in Figure 6-15. Feed motions distort the structuring element from ideal elliptical vibration path.
For programmed moves of short length (less than 100 micrometers), X-axis acceleration limits the peak velocity, regardless of the velocity commanded in the motion program. Individual movement segments for the parts machined in Chapters 7 to 10 were frequently even smaller, and feed velocities were typically programmed at 12 mm/min to 30 mm/min (200 to 500 μm/s). For 1000 Hz vibration frequency $F_{UP}$ at these speeds is only 0.2 to 0.5 μm/cycle with $HSR$ 0.003 to 0.007. Under such conditions the actual toolpath deviates only slightly from a pure ellipse producing form errors of only a few nanometers as illustrated by the sloped surface results given in Figure 6-16(b).
7 3-D MICROSTRUCTURES MADE USING ROUND-NOSED TOOLS

7.1 INTRODUCTION

A variety of 3-dimensional features were raster-machined by the Ultramill using a round-nosed tool. Three of these features are discussed in this chapter:

1. Upfeed "knife-edge" feature
2. "Flying wing" part (non-orthogonal planar features)
3. "Potato chip" part (sculpted 3-D geometry with sinusoidal surfaces in two directions)

Goals of this work were to:

- Demonstrate that EVAM can make 3-D features using axis motions to sculpt the geometry, rather than being created by the tool form.
- Fabricate positive (convex) and negative (concave) features with heights up to 20 μm. In comparison, the parts described in Chapter 1 typically had heights of 1 to 2 μm.
- Achieve aspect ratios greater than the parts depicted in Chapter 1, which typically had aspect ratios smaller than 0.01.
- Investigate form error and surface roughness that can be achieved by the Ultramill process.

These parts were machined in hard-plated copper. All were made using a 1 mm nose radius round-nosed diamond tool with 0° rake angle and 10° clearance angle. The elliptical vibration frequency was 1000 Hz with 400 Vp-p input voltage to the Ultramill. These conditions gave a nominal vibration ellipse of 22 μm x 4 μm or 24 μm x 4 μm overall, depending on whether a
PEC or PBL toolholder was used. All the parts were made with half-round piezo stacks and open-circulation cooling (System Configurations II or III-A), except for one repeat of the "potato chip" feature which demonstrated the form error improvement gained with closed-circulation cooling (Configuration IV). The system configurations are defined in Table 3-1 in Section 3.1.

Except for the "knife edge" feature, the motion programs were generated by the MORPH3D program described in Chapter 6. Structuring elements were based on the vibration ellipse dimensions, and a perfect circular cutting edge profile of 1.000 mm radius. Programmed upfeed velocity was 0.5 mm/s ($F_{UP} = 0.5 \mu m$ and $HSR = 0.007$), small enough to avoid significant distortion from an idealized ellipsoid structuring element.

Ideally, the beneficial effects of EVAM would be demonstrated by making identical comparison parts with a traditional micromachining process. Micro-milling would be the most appropriate process to make these sample part geometries; fast tool servo might have also been used. However, neither a milling spindle or FTS were available at the time the EVAM experiments were conducted. It was unfortunately not possible to make comparison parts.

7.2 **UPFEED KNIFE-EDGE**

7.2.1 **Part Description, Motion Program, and Machining**

This simple part was machined to see if the Ultramill could make a sharp edge in a non-binary feature, suitable for use as a fiducial. Figure 7-1 shows a cross section of this part. The leading edge face was at a 60° angle to the horizontal while the trailing face slope was 10°, corresponding to the tool clearance angle. The top of the knife edge was 3 \(\mu m\) above the background. The feature was 4 mm wide in the crossfeed direction. Crossfeed spacing between raster passes was 20 \(\mu m\). The toolpath motion was programmed based on the
vibration ellipse being tangent to the desired feature slope. The motion program for the knife edge continues the upward cut past the feature, then backfeeding in air before cutting the downslope (see inset on Figure 7-1). The program is given in Appendix I. Machining time was 23 minutes. The part was made using a single cutting depth.

**Figure 7-1.** Layout sketch showing cross-section of the "knife-edge" feature. Note backfeed step to elliptical vibration path create the feature, as well as the curved lower portion of the leading edge slope due to the
7.2.2 Metrology

Figure 7-2 is a Scanning Electron Microscope (SEM) image showing one end of the knife-edge feature. The viewpoint is looking straight down at the part, which creates foreshortening preventing details from being seen in the 60° slope. The top "knife edge" intersection between the inclined slopes is burr-free. The image shows the part cluttered with particles which the cleaning process could not remove.

The Zygo New View white light interferometer was used at maximum magnification to obtain the interferogram shown in Figure 7-3. This figure also shows a profile in the upfeed direction which provides a cross-section of the feature. The profile contains irregular, pit-like features which appear to be interferometry artifacts as they are not visible in the SEM image.

Figure 7-2. SEM image of "knife edge" feature
This profile allows estimation of knife-edge radius formed by the intersection of the two sloped surfaces. The enlarged detail in Figure 7-3 shows a circular arc of 300 nm radius superimposed on the knife edge. The resolution of the interferogram limits accuracy to which the knife edge radius can be estimated, but it appears to be sub-micrometer and possibly as small as a few hundred nanometers.

Another significant result seen in the interferometry is that the sloped surfaces appear to have slightly concave profiles. This was expected for the steep leading edge, where the large slope means the curved surface (formed by the elliptical tool vibration path) becomes tangent with the linear slope about 2/3 of the way up the part’s 3 µm height. However the appearance of

Figure 7-3. White light interferogram and profile of knife-edge part
The trailing slope is not understood. The concave profile is significant form error—a flat ramp was expected. The axes were programmed to make linear moves using the G-code “exact stop” command which forces them to halt at the end point of each programmed segment before executing the next movement command. They were supposed to move to the bottom of the 10° slope, stop, then move horizontally. This should have produced two intersecting linear surfaces with a very short curved region at the bottom of the slope. Instead the top of the curved region is steeper than the 10° programmed slope, meaning the actual cutting path was initially steeper than the programmed axis motion for a distance of approximately of 0.1 mm.

Three hypotheses are proposed to explain the concave form error although none is fully satisfactory.

- Different acceleration of the X and Z axes, perhaps aggravated by friction in the ballscrew nuts of the axis drives—In this case the X axis is assumed to take incrementally longer to reach programmed velocity compared to the Z axis, so that the tool motion is steeper than intended. Programming the first part of the downslope motion out of contact with the workpiece was supposed to eliminate this possibility, but the length of the “in-air” segment was perhaps too short.

- Vibration of the Ultramill structure and/or unfavorable control system dynamic response—The initial contact of the tool with the workpiece material creates an upset force. The control system transient response leads to unfavorable axis motion resulting in deeper (steeper) cuts near the top of the feature.

- Interference between the tool flank face and machined surface, since the slope and tool flank angle are approximately the same—This should lead to a rougher surface but does not seem to be a way to create concave form error. Increased tool forces might aggravate the vibration and axis response as proposed above.
7.3 FLYING WING

7.3.1 Part Description, Motion Program, and Machining

The “flying wing” part shown in Figure 7-4 was made to test EVAM’s ability to make planar surfaces which are not orthogonal to any of the machining axes. The part is 20 µm tall above the background which is 10 to 20 times greater vertical relief than the micrometer-range binary parts previously made with the Ultramill. The shallow “trailing edge” features have a 6.9º inclination and were machined with upward feed. The 9.7º sloped surfaces, which join at the feature’s “nose”, were cut downward. These slope angles are defined along the X-axis as shown in the profile in Figure 7-4. In the plane of the surfaces (compound X-Y-Z angle) the upward cut slope is 7.9º and the downcut slope is 12.0º. The upfeed moves were programmed using the G09 G-code command which forces the axes to stop at the end of a linear move segment. Crossfeed spacing of the individual raster passes was 10 µm. The part was machined with two roughing cuts of 10 µm and 9 µm depth respectively, followed by a finish cut 1 µm deep. Machining time was 49 minutes.

Figure 7-4. Design sketch for the "flying wing" feature
7.3.2 Metrology Results

Figure 7-5 is a wide-angle white light interferogram of the completed flying wing. The figure also shows upfeed profiles made at two cross section locations. It is evident that this part has form error as well as unexpected surface features. The "rear" surfaces cut with upward feed motion (6.9º slope) have a linear profile, at the desired slope angle. However the steeply sloped "front" surfaces have concave curvature rather than the desired planar surface. Also present in the steep surfaces are a series of linear features roughly perpendicular to the upfeed direction of the tool. The concave form error and linear features are confirmed by the SEM image in Figure 7-6.

The source of these errors is not apparent. The concave form error is suggested to be related to interference between the machined surface and the flank face of the round-nosed diamond tool. The 9.7º programmed slope of the toolpath is slightly less than the nominal 10º tool clearance angle. However this slope is only along the X-axis (upfeed) direction. The compound angle formed by the non-orthogonal planar surface and the tool flank face is acute, and the round-nosed tool turns out to not clear the intended surface. Concave form error may be cut into the machined surface because of the 1 mm tool radius, which results in a large penetration beyond the intended plane.
Figure 7-5. White light interferograms of the "flying wing" part, with profiles at two locations showing concave form error in the steeply sloped surfaces.

Figure 7-6. SEM image of “flying wing” feature. (left) Overall part  (right) Detail showing concave slope and linear feature errors.
The linear features in the 9.7° slopes (Figures 7-5 and 7-6) were imaged using the Zygo white light interferometer at maximum magnification. They were seen to make a descending stair-step profile in the direction of tool motion. The individual features were 20-60 nm high and regularly spaced at an upfeed interval of approximately 10 μm. For the programmed upfeed velocity of 0.5 mm/s, the feature spacing corresponds to a frequency of about 50 Hz. This was not one of the dominant frequencies found in the single-groove autocovariance analysis in Chapter 5 (see Figure 5-9). The origin of the linear features may also be from interference of the tool flank face with the cut sloped surface, but a mechanism has not been identified that explains the feature geometry or spacing.

Two other important metrology results for the flying wing are the flatness and roughness of the machined tilted planar surfaces. Figure 7-7 shows how flatness of the shallow-sloped surface was checked with the New View white-light interferometer. The part was tilted so the shallow-slope "rear wing" surface was orthogonal to the interferometer’s light beam. The measured area was delineated to exclude transition regions near the feature edges. “Flatness” was defined as deviation of the triangular measurement region’s surface from a best-fit plane. As seen in Figure 7-7, this was 294 nm RMS.

Figure 7-7 also shows the surface variation along span and chordal profiles. Deviation along the span was 310 nm RMS, slightly greater than flatness error for the entire part. The span measurement is roughly parallel to the crossfeed direction so this profile includes all form variations caused by long-term Ultramill temperature instability. Open-circulation cooling was used to machine this part. This cooling mode can cause variation in machined surfaces due to non-periodic thermal expansion or contraction of the Ultramill structure, as was demonstrated by experiments in Chapter 4—in particular see Figure 4-9 which shows variation of tool tip position measured during capacitance gauge experiments. The irregular spacing of peaks, and abrupt changes in the flying wing span profile are similar to the changes in cap
Figure 7.7. (top) White-light interferogram showing flatness of the shallow sloped surface forming one of the rear "wings". (bottom) Profiles along the span and chord of the flatness evaluation area.
gauge position during open-circulation cooling. Figure 7-8 shows there was significant piezo stack temperature variation while machining the final pass, with an overall 3 ºC range and a suggestion of smaller shorter term temperature variations throughout the cut. These results suggest that flatness of tilted planar surfaces can be improved by switching the Ultramill to closed circulation cooling to minimize temperature fluctuations.

The chordal profile is approximately parallel to the upfeed direction and, showed a deviation of 132 nm RMS. The variation took the form of step-change features approximately 400 nm PV, at an average upfeed spacing of 3 µm. This corresponds to a time domain frequency of about 170 Hz. Again, this frequency does not correlate with any discovered by the single-groove autocovariance analysis in Chapter 5.

Surface roughness was evaluated with white-light interferometry at three locations on the shallow-sloped surface of each "wing". The sample regions were 50 µm square and the surface roughness was relative to the best fit plane. Roughness was also measured for upfeed and crossfeed profiles in the sample regions. The results for each "wing" were averaged and are given in Table 7-1. Roughness of the sloped surface was significantly greater than for the flat

Figure 7-8. Piezo stack temperatures while machining finish pass of the flying wing.
background surrounding the feature. This is may be due to a change in the tool force when machining on an upslope. In EVAM, cutting occurs while the tool is sweeping upward from the bottom part of the vibration ellipse, and the rake angle grows increasingly negative (refer back to Figure 2-2). Compared to a flat surface, the rake angle is more negative at any point in the vibration cycle, by the angle of the slope. Cerniway [8] and Negishi [39] predict greater machining forces at negative rake angles; therefore cutting on an upslope should lead to larger average and instantaneous forces with consequent effect on tool vibration and surface roughness.

Table 7-1  Surface Roughness of Shallow Slopes on "Flying Wing" Part

<table>
<thead>
<tr>
<th>Location</th>
<th>RMS Roughness a</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Area Based b</td>
<td>Upfeed Profile c</td>
<td>Crossfeed Profile d</td>
</tr>
<tr>
<td>Left wing</td>
<td>41 nm</td>
<td>25 nm</td>
<td>25 nm</td>
</tr>
<tr>
<td>Right wing</td>
<td>41 nm</td>
<td>29 nm</td>
<td>33 nm</td>
</tr>
<tr>
<td>Theoretical roughness</td>
<td>-</td>
<td>0.7 nm ε</td>
<td>4 nm f</td>
</tr>
<tr>
<td>(wing)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flat background</td>
<td>19 nm</td>
<td>18 nm</td>
<td>16 nm</td>
</tr>
<tr>
<td>Theoretical roughness</td>
<td>--</td>
<td>0.2 nm</td>
<td>4.5 nm</td>
</tr>
<tr>
<td>(flat background)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

a Best-fit plane removed, then filtered by high-pass filter (10 mm⁻¹ threshold frequency)
b 50 μm x 50 μm region 
c d 50 μm long profile e From Figure 2-6 f From Equation 2-27
7.4 SCULPTED 3-D GEOMETRY ("POTATO CHIP" PART)

7.4.1 Part Description, Motion Program, and Machining

Figure 7-9 shows a surface consisting of parallel grooves with sculpted 3-D geometry. The grooves have a sinusoidal section in the crossfeed direction, which is superimposed on a sinusoidal profile in the upfeed direction. Equation 7-1 describes the surface for one groove:

\[
Z(X,Y) = \left( \sin \left( \frac{2\pi X}{\lambda_{UP}} - \frac{\pi}{2} \right) - 1 \right) \cdot \left( A_{UP} - \frac{A_{CROSS}}{2} \right) + \frac{A_{CROSS}}{2} \sin \left( \frac{2\pi Y}{\lambda_{CROSS}} - \frac{\pi}{2} \right) \]

(7-1)

where \( A_{UP} \) and \( \lambda_{UP} \) are the amplitude and wavelength in the upfeed direction, and \( A_{CROSS} \) and \( \lambda_{CROSS} \) are the crossfeed amplitude and wavelength. For the part shown, the width (\( \lambda_{CROSS} \)) of each groove is 320 \( \mu \)m and their maximum depth is 5 \( \mu \)m (\( A_{CROSS}=2.5 \) \( \mu \)m). The grooves have an aspect ratio of 0.031, defined as the peak-to-valley height divided by half the width. The length (\( \lambda_{UP} \)) of the part is 1280 \( \mu \)m and the amplitude \( A_{UP} \) of the upfeed profile is 4.5 \( \mu \)m. The overall peak-to-valley height of the part is 9 \( \mu \)m.

Appendix K is the MATLAB program listing for application of the morphology method to generate the motion program. The individual raster passes used 32 equal-length linear segments to approximate the sinusoidal upfeed profile. The theoretical maximum form error due to linearization was estimated at less than 50 nm. This was further reduced by specifying the axes moves to use the G01 command, which blends linear moves near the programmed endpoints. The crossfeed spacing was 5 \( \mu \)m, requiring 251 raster passes. The part was made in a single cut. Machining time was 51 minutes.
Figure 7-9 "Potato chip" part with 3-D sculpted geometry.

Figure 7-10. White-light interferogram of the "potato chip" part. Shown is the part made using closed-circulation cooling for the Ultramill.
7.4.2 Metrology Results

Two copies of the "potato chip" part were made, the first using open-circulation cooling of the Ultramill and the second after conversion to closed-circulation cooling.

White-light interferometry was used to measure these parts. The surface slopes were sufficiently shallow so that data could be obtained over the entire part. Figure 7-10 is a wide-angle interferogram of the part. Figure 7-11 shows upfeed and crossfeed profiles extracted from the interferometry data.\textsuperscript{15} In the upfeed direction, the machined profile closely matched the intended profile (Figure 7-11(a)). An individual raster pass took less than 12 seconds to execute, so there was scant opportunity for Ultramill temperature instability to cause form error. Crossfeed profiles were affected by temperature instabilities because they covered the full machining duration. The crossfeed profile for the part made with open-circulation cooling (Figure 7-11(b)) shows form error as large as 500 nm between the design and actual surfaces of ridge and groove features. When closed-circulation cooling was employed (Figure 7-11(c)) the crossfeed profile showed form error of less than 80 nm. This result demonstrates once again the importance of using closed-circulation cooling to minimize Ultramill temperature variations.

The surface roughness was not measured, because on this part the curvature of the surface at any location is relatively high so that there is significant departure from a flat plane. It was feared that this curvature would affect the accuracy of measurements rendering them of little value.

\textsuperscript{15} MATLAB tools developed by K. Garrard at the PEC (“Import_metropro” and “Slicer”) enabled 2-D profiles to be extracted from the Zygo data set, along user-defined section lines. The section lines for profiles could be positioned more precisely with these tools than by drawing them on the Zygo New View’s interactive display. Also they store the profile data in MATLAB arrays, facilitating analysis.
Figure 7.11. Upfeed and crossfeed profiles of "potato chip part" extracted from white-light interferometry data and plotted using MATLAB.
Micro-optics are an important application area for EVAM. Before attempting to make more complex parts, concave spherical reflectors were machined with the Ultramill to identify possible error sources. Form error on such parts can readily be measured using a laser interferometer with a spherical element, making them suitable for process development experiments.

8.1 Part Design

Figure 8-1 is a MATLAB representation of the intended spherical surface. The concave reflector has an aperture radius of 0.75 mm (1.5 mm diameter), a radius of curvature of 4.5 mm, and maximum sag of 63 μm. The part’s aspect ratio is 0.042 (sag divided by the aperture radius). The aperture diameter was chosen to be a convenient size for imaging with the Zygo GPI laser interferometer. Once the aperture was selected, the radius of curvature was calculated to assure the maximum downward angle of the surface was smaller than the tool’s clearance angle so that the tool flank face would not impact the machined surface. This is depicted in Figure 8-2, which shows the centerline profile of the surface. The relationship between aperture radius \( r \), maximum sag \( s \), radius of curvature \( \rho \), and maximum surface slope \( \theta \) is given by Equations 8-1 and 8-2:

\[
\rho = \sqrt{\left(\frac{r}{\tan \theta}\right)^2 + r^2} \quad (8-1)
\]

\[
s = -\rho + \sqrt{\rho^2 + r^2} \quad (8-2)
\]

where \( \theta \) is less than or equal to the tool clearance angle \( \alpha_0 \).
The reflector radius of curvature is much larger than the tool nose radius of 1 mm, so that interference did not occur between the tool flank face and the surface, at locations away from the centerline.

Figure 8-1. Design dimensions of spherical reflector.

Figure 8-2. Centerline cross section of concave spherical reflector.
8.2 PART FABRICATION

8.2.1 Motion Program and Machining Strategy

Table 8.1 gives the tool geometry and process parameters for machining the spherical reflector. The motion program was generated using the MORPH3D method described in Chapter 6. Appendix L gives the MATLAB code used to generate the motion program. The reflector surface and the structuring element were modeled at a 1 μm square grid resolution. The structuring element was based on the 1000 μm tool nose radius with no deviations from a circular arc. The maximum programmed upfeed velocity was only 0.2 mm/s, so distortion from the elliptical tool vibration path due to feed motion was assumed to be negligible.

<table>
<thead>
<tr>
<th>Tool Data</th>
<th>Process Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toolholder</td>
<td>Ellipse size (A x B)</td>
</tr>
<tr>
<td>Nose Radius</td>
<td>Frequency, ( f )</td>
</tr>
<tr>
<td>Rake Angle</td>
<td>Maximum upfeed velocity, ( U )</td>
</tr>
<tr>
<td>Clearance Angle</td>
<td>( F_{up} )</td>
</tr>
<tr>
<td></td>
<td>( HSR )</td>
</tr>
</tbody>
</table>

Crossfeed spacing was 10 μm, and 151 raster passes were needed to cover the width of the part. The desired curved profile in the upfeed direction was approximated by linear axis moves with a minimum segment length of 10 μm. Along the centerline there were 50 segments in the upfeed direction, reduced to 2 at the edges. Figure 8-3 shows the required toolpath for raster passes at two crossfeed locations, along the part centerline and approximately halfway between the center and the edge. Maximum vertical deviation between programmed linear
segments and the curved tool path extracted from the morphology program is less than 100 nm. This variation is of the same magnitude as the expected range of surface roughness achieved with closed-circulation cooling.

The required part depth of 63 μm was achieved by six successive roughing cuts each with a nominal depth of cut of 7 to 11 μm, followed by a finish cut 1 μm deep. To reduce machining time, only the raster passes immediately straddling the centerline were executed during the first cut. With each deeper cut, more raster passes were executed. Only the last roughing cut and the finish cut required the full number of raster passes. This scheme is depicted by Figure 8-4.

8.2.2 Machining Operations

The spherical reflector parts were cut using closed-circulation cooling with triangular stacks in the Ultramill.\textsuperscript{16} Coolant flow during machining was approximately 0.6 L/min. Four spherical reflectors were made, three in hard-plated copper and the fourth in a 2000-series aluminum alloy.

Total machining time per part was initially 68 minutes increasing to 74 minutes after modifying the machining program to correct an error caused by Z-axis friction, as described in Section 8.3. The finish cut by itself took 16 minutes. Average material removal rate was 12,800 μm\(^3\)/s overall, or 16,400 μm\(^3\)/s for the roughing passes alone. This MRR is comparable to laser ablation methods and micromilling [2,6].

\textsuperscript{16} System Configuration IV, as defined in Table 3-1 in Chapter 3.
Figure 8-3. Toolpaths for 2 raster passes on the spherical reflector.

Figure 8-4. Machining scheme for the concave spherical reflector
8.3 **METROLOGY RESULTS**

8.3.1 **Form Error**

Figure 8-5 shows part surface error measurements for two spherical reflectors made using the Zygo GPI laser interferometer. Figure 8-5(a) shows the first part to be machined, while Figure 8-5(b) is the last part made in copper, which included corrections for errors measured in the first part. The interferograms show the departure of the surface from the concave design radius of curvature (4.5 mm). The initial part had a figure error of 124 nm RMS while on the final part this was reduced to 62 nm RMS.

The first spherical surface machined contained two major errors assignable to the Nanoform diamond turning machine. The first is astigmatism, which was determined to be due to a squareness error between the Nanoform’s X and Y axes. The squareness error was estimated at 0.004 radians based on the interferometry results. Figure 8-6 shows how X-Y squareness error causes each crossfeed pass to be sheared relative to the preceding one. This distorts the shape of the surface. This problem was corrected in subsequent versions of the motion program by offsetting the programmed starting point of each raster (upfeed) pass from the preceding one, in the direction opposite to the squareness error. The actual starting point of the raster pass was thus brought to the proper location. The corrected part show in Figure 8-5 (b) still contains a small amount of astigmatism which is thought to be due to residual X-Y squareness error.
(a) First spherical reflector to be machined in copper, containing astigmatism and crossfeed "trench" errors. Form error is 124 nm RMS.

(b) Final spherical reflector made in copper, with form error reduced to 62 nm RMS.

**Figure 8-5.** GPI interferograms of spherical reflectors made with the Ultramill. Plots show deviation of the surface from the design radius of curvature (4.5 mm concave).
The second major error was a "trench" perpendicular to the upfeed tool motion, bisecting the part at the bottom of the spherical radius. The trench is visible in the interferogram in Figure 8-5(a). This feature was caused by friction in the Z-axis ballscrew nut [12] when the Z-axis reversed direction at the bottom of the concave tool path. Figure 8-7 plots X-axis versus Z-axis position for the centerline raster passes using data obtained from the Nanoform’s position-measuring interferometers\textsuperscript{17}. As shown in the top portion of Figure 8-7, the ballscrew friction causes a delay in the start of Z-axis upward motion. The X-axis motion at the bottom of the part is unaffected, so that the actual tool path deviates from the programmed path. To

\textsuperscript{17} Appendix A in [6] describes use of the Nanoform controller's built-in data capture software to obtaining axis position and performance data.
Figure 8-7. Preventing crossfeed trench at base of concave part. (Top) Friction in Z-axis ballscrew nut during reversal of direction causes it to lag behind X-axis at bottom of part (Bottom) Slowing down X-axis after Z-axis reversal keeps the two axes synchronized.
reduce this effect the programmed X-axis velocity was reduced for the movement segment immediately after the point of Z-axis reversal (bottom panel of Figure 8-7). This allows the Z-axis to "catch up" with the X-axis so that the combined axes’ motion conforms approximately to the desired toolpath. Figure 8-8 shows surface profiles in the upfeed direction at the bottom of the reflector, obtained using the New View white light interferometer at high power to give a narrow field of view. These profiles show that slowing the X-axis down during Z-axis reversal reduces the "trench" error from 350 nm PV to 120 nm PV. Note that the trench can no longer be seen in the interferogram of the corrected part, in Figure 8-5(b).

Appendices L.1 and L.2 give the revised MATLAB code to implement squareness correction and trench suppression.

Both surface profiles in Figure 8-8 show periodic features with a peak-to-valley height of 80 to 100 nm. The spatial frequency of these features corresponds to a time-based frequency of approximately 30 Hz, the same as the diaphragm pump used to circulate the dielectric coolant fluid through the Ultramill. These results confirm one of the important conclusions of the temperature stability study in Chapter 4, that closed-circulation cooling minimizes figure error caused by Ultramill temperature variations but that the installed diaphragm pump pressure-pulsation limits the quality of the machined surface (a continuous-flow centrifugal pump does not produce the pressure pulsations, but it could not be used as the dielectric coolant was incompatible with the seal materials of the available model). These parts were cut with a coolant flow of 0.6 L/min, greater than the flow rate subsequently determined to provide optimum surface roughness and form error.
8.3.2 Surface Roughness

Local surface finish measurements were made for the spherical reflector parts using the Zygo New View white light interferometer at 50x magnification (Mireau objective). Area-based surface roughness was measured at nine locations shown in Figure 8-9. The surface roughness was calculated by the interferometer software as the deviation from the best fit plane at each location for a 50 μm square region. This data was subsequently filtered by a high-pass filter with a cutoff spatial frequency of 10 mm\(^{-1}\), to remove the long-wavelength contribution from the concave figure of the part.

Roughness was measured for reflectors made in hard-plated copper and an aluminum alloy. The aluminum workpiece was scrap from the lab and its origin, alloy and temper were unknown. Its Vicker hardness was measured using a diamond indenter and found to be 0.7 GPa (\(H_V = 80\)), which is consistent with several 2000 series aluminum alloys. Before machining with the Ultramill, the aluminum workpiece was faced flat on the ASG lathe-type
diamond turning machine using a large-radius diamond tool. The preliminary facing operation produced a specular surface with a surface roughness of 6 nm RMS (based on 50 μm square region), as measured by the Zygo white light interferometer.

Tables 8-2 and 8-3 shows the surface roughness results, respectively, for the hard-plated copper and aluminum reflectors. There is significant difference between the two materials.

For hard-plated copper the roughness ranged between 20 and 30 nm RMS. The aluminum reflector’s roughness was larger—37 to 80 nm RMS. This part also had two distinct zones of significantly different surface roughness. For the shaded area shown on Figure 8-9 (corresponding to locations 4, 7, 8, and 9 in Table 8-3) the roughness measurements were more than 50% greater than on the rest of the part. Through an optical microscope the aluminum part showed a pebbly surface, and there was a noticeable difference in appearance between the two regions. This sharp roughness variation between regions of the surface did not appear in the plated copper parts.

Figure 8-10 compares high-magnification white-light interferograms for reflectors machined in the two materials. In the hard-plated copper part (Figure 8-10(a)) regular features in the upfeed direction can be seen. These correspond to the 30 Hz pulsation of the coolant pump. The crossfeed raster passes can also be identified in the surface. In the aluminum part (Figure 8-10(b)), the large surface roughness obscures the periodic features caused by pump pulsation and rastering, and the surface shows greater peak-to-valley irregularity.

There is no obvious reason for the different surface roughness in the two materials. Excessive tool wear is excluded as a cause because the same tool subsequently made other parts in plated copper and the 6061 aluminum ellipsoidal reflector described in Chapter 9. These later parts had surface roughness similar to the plated copper spherical reflectors. In the high roughness zone on the aluminum part, the Ultramill cut both downward and upward slopes; this seems to
eliminate some unique behavior of the specific alloy during EVAM. Not knowing the history of the aluminum raw material makes it impossible to know if the poor surface finish occurred because of damage, or some treatment particular to the specimen. The result does suggest that material selection include cutting tests, since the aluminum piece could be faced flat to a very smooth finish of only 6 nm RMS and the pebbly, rough surface was unexpected.
Figure 8-9. Location of surface roughness measurement zones on spheroid reflector. Shaded area indicates region of greater roughness on the aluminum part.

Table 8-2. Area-based Surface Roughness for a Hard-Plated Copper Spherical Reflector

<table>
<thead>
<tr>
<th>Material</th>
<th>Hard-Plated Copper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region</td>
<td>1</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 8-3. Area-based Surface Roughness for an Aluminum Spherical Reflector

<table>
<thead>
<tr>
<th>Material</th>
<th>Aluminum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region</td>
<td>1</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>73</td>
</tr>
</tbody>
</table>
Figure 8-10. White-light interferograms for spherical reflectors, showing difference in machined surface between (a) hard-plated copper and (b) aluminum.
9 OPTICAL SURFACE FABRICATION: FIBER OPTIC BEAM BENDER

9.1 INTRODUCTION

To investigate its ability to make functional optics, the Ultramill was used to fabricate a focusing reflector such as might be used in a low-loss fiber optic beam splitter. This was a considerably more complex project than the spherical reflector in Chapter 8, requiring fabrication of an aspheric, non-rotationally symmetric surface.

9.1.1 Beamsplitter Concepts

Figure 9-1 illustrates how an optical fiber propagates light by repeated total internal reflection (TIR). The fiber core and its surrounding cladding have different indices of refraction, which causes TIR when light rays are incident to the core-cladding interface at smaller than a critical angle. Incoming light rays must fall within a "cone of acceptance" to avoid exceeding the critical angle. Light emerges from the fiber as a diverging beam with an angle approximately the same as that of the entering rays.

Figure 9-1. Schematic of light propagating through an optical fiber by total internal reflection, and angle of acceptance and divergence.
Fiber optic applications frequently require splitting a source beam into multiple outputs. An inexpensive design is the "fused fiber splitter", in which two fibers are placed adjacent to one another and heated in controlled fashion until they become fused together. Part of the input light travels through the fiber wall into one of the branch fibers, while the remainder stays in the same fiber to form the other branch. Fused splitters have large losses, typically greater than 3 dB attenuation of the input power (50% loss) [21].

Attenuation smaller than 0.15 dB can be achieved with a "free-space" splitter such as that shown schematically in Figure 9-2. A planar reflector or prismatic splitter separates the source beam into multiple output beams. A refracting element is needed to collimate the diverging source beam ahead of the splitter element (shown as a ball lens in the figure). Similarly, beams entering the receiving fibers must be focused into a cone with an angle narrower than the fiber's angle of acceptance. This coupling function requires an additional lens for each branch.

Figure 9-3 shows a new beamsplitter concept in which the Ultramill is used to fabricate a focusing mirror that performs both division of the source beam and coupling into the receiving fibers. In a 1-to-2 splitter, the curved mirror eliminates the three collimating lenses, or 3 out of 4 optical components used in the standard design. The number of critical dimensional relationships between components is reduced from six to three. A final advantage of the concept is that EVAM could be used to precision machine a replication master in a material such as steel. Reflectors might then be mass-produced by polymer injection molding, with a mirror surface obtained by a using a process such as CVD to coat the part with gold, silver, or

---

18 The reverse case, of combining multiple input beams into a single output, is also required. Most splitter designs can be reversed to function as combiners.
19 The reflecting beam splitter project was suggested by T. G. Bifano of the Photonics Center at Boston University.
Figure 9-2. Free-space beam splitter with refracting elements to collimate light from the source fiber and to focus the branch beams into the receiving fibers.

Figure 9-3. Fiber optic beam splitter, using 3-D curved reflector.
aluminum. Figure 9-4 shows one possible physical implementation of the focusing reflector splitter. In this concept a monolithic base structure contains the reflecting surfaces. A push-on top structure holds the fiber ends in the proper position and orientation relative to the reflecting surfaces.

9.2 DEMONSTRATION REFLECTOR

9.2.1 Beam-Bender Concept and Test Setup

The Ultramill was used to fabricate the focusing reflector element for the beam-bender system shown schematically in Figure 9-5. This project is sufficient to evaluate EVAM's ability to make millimeter-scale, high-sag, non-rotationally symmetric optical surfaces needed for a beam splitter. Functional testing is simplified, as there is only one surface. As shown in the figure, a microscope objective couples the beam from a visible-light laser into the source fiber. The reflector element bends the divergent beam from the source fiber through a right angle.
while simultaneously coupling it into the receiving fiber. The output from the receiving fiber is measured by a silicon detector optical power meter. The fibers are held and aligned using standard fiber chucks and positioning stages (not shown in the figure).

**9.2.2 Optical Fiber Specification**

Two fiber characteristics affect the design of the focusing reflector's geometry: the core diameter, and the numerical aperture $NA$. The core diameter sets the size of the beam leaving the source fiber. $NA$ is related to the acceptance / divergence angle by

$$NA = \sin \alpha \quad (9-1)$$

where $\alpha$ is the half-angle of the cone of acceptance, and also the angle of the beam divergence. The total angle of acceptance / divergence is $2\alpha$. 

The test reflector was designed for Corning "InfiniCor®", a multi-mode silica fiber used in LAN applications. This fiber has the following properties [11]:

- Core diameter: 62.5 μm
- Numerical aperture NA: 0.275
- Acceptance / divergence angle: 16.0° (calculated from NA)

### 9.2.3 Optical Surface Design

As shown in Figure 9-6, an ellipse has the property that any ray originating at one focus will be reflected at an angle that causes it to pass through the other focus. This allows an ellipsoidal reflector to be designed without using optical design software or theory. The demonstration reflector is a portion of the surface an ellipsoid formed by revolving an ellipse about its major axis. This type of ellipsoid has an elliptical cross section in the X-Z plane, circular sections in the X-Y and Y-Z planes, and elliptical cross sections in any other intersecting plane.

**Figure 9-6.** Reflection property of an ellipse. Any ray originating at focus $F_1$ (blue rays) is reflected and passes through the other focus, $F_2$ (red rays).
Figure 9-7 shows the elliptical centerline profile of the demonstration reflector. In this figure
the surface is shown in the "part coordinate system", in which the X and Z axes correspond to
the centerlines of the receiving and source fibers. The source for the diverging light rays is at
focus $F_1$ which is on the Z axis. $F_1$ is located inside the source fiber, at a position obtained by
projecting the light rays leaving the fiber at the critical angle $\alpha$ backward until they intersect
the Z axis. The other focus, $F_2$, is at the center of the receiving fiber’s face. The source fiber
glass is located 3 mm from the origin in the Z-direction while the receiving fiber face is 8 mm
from the origin in the X-direction. These dimensions provide reasonable working distances
for the test setup. A beam splitter application would likely position the fibers closer to the
reflector to keep the package size small.

![Diagram of elliptical centerline profile](image)

**Figure 9-7.** Layout of elliptical centerline profile of reflector surface
for the beam bender
In Figure 9-7 the ellipse major axis is inclined at angle $\theta$ relative to the receiving fiber centerline. The elliptical cross section has semi-major axis $A = 5.388$ mm and semi-minor axis $B = 3.23$ mm. The reflector surface is obtained by revolving the elliptical section around its major axis, so that it has curvature into the plane of the figure. The required surface has a radius perpendicular to the major axis of 3.23 mm.

Appendix M provides a detailed description of the derivation of the reflector surface including relevant formulas from analytic geometry.

9.3 REFLECTOR FABRICATION

9.3.1 Machining Setup and Motion Program

The Ultramill was used to machine the beam bender reflector in 6061-T6 aluminum. Aluminum has a reflectance of 91 percent at the 632 nm wavelength provided by the laser used for the functional test [60].

Figure 9-8 is a sketch showing the machining setup. The reflector surface is tilted in a workholder so that the long axis of the part is parallel to the nominal upfeed direction of the Ultramill tool. This orientation keeps the maximum surface slope in the upfeed direction to less than 10 degrees, which is smaller than the clearance angle of the tool. These maximum slopes are approximately the same as for the spherical reflector described in Chapter 8.

---

20 It was hoped to cut a second reflector in stainless steel. Regrettably this could not be executed before the Nanoform was taken out of service to convert the spindle to a C axis.
Figure 9-8. (left) Workpiece setup for machining beam bender reflector (right) Design surface is rotated through angle $\psi$ to align it for machining.

The physical orientation of the part in Figure 9-8 required defining the elliptical surface in a "machining coordinate system". The surface was rotated through angle $\psi$ from the "part coordinate system" to bring the endpoints of the surface level with one another. Appendix N provides a mathematical description of this operation. For the beam bender, $\psi = 45.03^\circ$.

Overall size of the part in the machining orientation was 2.51 mm x 1.70 mm, with maximum sag of 131 μm. The part aspect ratio, defined as sag divided by half the width, was 0.154.

Table 9-1 summarizes the tool geometry, and EVAM machining parameters used to cut the reflector. The motion program was generated using the morphology method described in Chapter 6. The MATLAB code is given in Appendix O. The reflector surface and structuring element were modeled at a 1 μm square resolution. Maximum programmed feed velocity was only 0.2 mm/s ($HSR = 0.0026$) so there was negligible distortion from the elliptical tool vibration path due to the feed motion.
Table 9-1. Machining Conditions for Beam-Bender Reflector

<table>
<thead>
<tr>
<th>Tool Data</th>
<th>Process Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toolholder</td>
<td>PBL</td>
</tr>
<tr>
<td>Nose Radius</td>
<td>1 mm</td>
</tr>
<tr>
<td>Rake Angle</td>
<td>0º</td>
</tr>
<tr>
<td>Clearance Angle</td>
<td>10º</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The motion program included two corrective features based on the experience machining spherical reflectors. X-axis velocity was slowed down in the region where Z-axis direction reversal takes place (see Section 8.3.1) to prevent cutting of a lateral trench discontinuity. To prevent astigmatism caused by X-Y axis squareness error, the starting point of each crossfeed pass was offset in the opposite direction. The squareness error correction was 0.004 radians (825 arc-sec), which was the value used for the spherical surfaces.

The reflector surface required 171 raster passes at 10 µm crossfeed spacing. Nine roughing cuts accomplished bulk material removal. These were followed by a 1.4 µm deep finish cut. To reduce machining time, only the raster passes around the centerline were executed during the first few roughing cuts. To further reduce the machining time the crossfeed spacing was 50 µm for the first eight roughing cuts. Table 9-2 shows the depth of cut, number of raster passes, and crossfeed increment for each cut.

The raster passes included 5-µm deep linear ramps at each end. These were to assure that the full reflector surface was machined in case of workpiece misalignment relative to the Nanoform axes.
Table 9-2. Raster Passes, Crossfeed Spacing, and Depth of Cut for Reflector Surface

<table>
<thead>
<tr>
<th>Cut</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td># of Passes</td>
<td>3</td>
<td>7</td>
<td>11</td>
<td>15</td>
<td>19</td>
<td>23</td>
<td>27</td>
<td>31</td>
<td>171</td>
<td>171</td>
</tr>
<tr>
<td>Crossfeed Spacing, μm</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Depth of Cut, μm</td>
<td>10</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>1.4</td>
<td></td>
</tr>
<tr>
<td>Cumulative Depth, μm</td>
<td>10</td>
<td>25</td>
<td>40</td>
<td>55</td>
<td>70</td>
<td>85</td>
<td>100</td>
<td>115</td>
<td>130</td>
<td>131.4</td>
</tr>
</tbody>
</table>

9.3.2 Machining Operations

Figure 9-9 shows preparation of the workpiece as well as the machining setup on the Nanoform diamond turning machine. The raw material was a 1/2" diameter 6061-T6 aluminum rod. The rod was clamped in a workholder at an angle of 45°, which is approximately the required rotation angle $\psi = 45.03°$ between the part and machining coordinate systems. The figure shows how a chamfer was milled into the edge of the rod, to make a flat surface into which the reflector surface was cut. Figure 9-9(c) shows the workpiece and workholder installed on the Nanoform with the chamfered surface set normal to the Ultramill tool, and aligned with the plane formed by the X and Y machine axes. Unlike most parts made on the Nanoform which used the spindle vacuum chuck, the workholder for the reflector was mounted on a toolpost fixed to the X-axis. The chamfer was aligned in the Nanoform’s X-Y plane using a Federal Gauge.
Figure 9-9. Preparation and installation of the aluminum workpiece used to make the beam bender reflector. (a) Milling 45° chamfers. (b) Workpiece with chamfers in the workholder. (c) Workpiece and workholder installed on the Nanoform. A Federal gauge is used to align the chamfered work surface with the plane formed by the DTM's X and Y axes.

The milled aluminum surface finish was several micrometers RMS, too rough to give a specular tool tip reflection for touchoff. Initial touchoff was therefore confirmed by using the Y-axis camera to identify a small feature cut in the surface. A 3 mm x 2 mm flat region was next rastered using the Ultramill. This produced a smooth datum surface, into which the reflector was machined.

The Ultramill used triangular piezo stacks and closed-circulation cooling (System Configuration IV, defined in Table 3-1 in Section 3.1). The coolant flow rate was 0.4 L/min, the optimum rate determined from the cooling system experiments in Chapter 4.
Machining time to make the reflector was 121 minutes, not counting the rastered foreground. The finish cut took 30 minutes. The average material removal rate was 21,000 μm³/sec overall and 28,000 μm³/sec for only the roughing cuts. This is approximately twice the MRR when machining the spherical reflectors in the preceding chapter. The higher average MRR was because 50 μm crossfeed spacing was used for most of the roughing cuts, reducing the total number of required raster passes.

### 9.4 Performance Test

Figure 9-10 depicts the test setup schematically and is the same as Figure 9-5. Figure 9-11 is a photograph showing the optical fiber installation relative to the completed reflector. A 10x microscope objective coupled the laser beam (λ = 632 nm) into the source fiber. Incident light intensity was measured using a silicon detector optical power meter. Optical power readings were made with the silicon detector located 8 mm from the emission point.

![Test setup for the beam bender part.](image)

**Figure 9-10.** Test setup for the beam bender part.
Figure 9-11. Installation of aluminum workpiece containing beam bender reflector, source, and receiving fibers.

The reflecting efficiency $E_{REFL}$ of the reflector is a measure of the incident light that is reflected, without regard for focus or image quality:

$$E_{REFL} = \frac{P'_1}{P_1}$$  \hspace{1cm} (9-2)

where $P_1$ is the optical power leaving the source fiber and $P'_1$ is the optical power measured after the reflector element.

The coupling efficiency $E_{CPLG}$ measures the fraction of the power delivered by the receiving fiber. It is the principal measure of performance for the reflecting system and is given by

$$E_{CPLG} = \frac{P_2}{P_1}$$  \hspace{1cm} (9-3)

where $P_1$ is as in Equation 9-2 and $P_2$ is the power of the beam leaving the receiving fiber.
Table 9-3 gives the results of the performance test. Each reported power value is the average of three measurements. The source beam power, $P_1$, was measured by removing the reflector from test setup and projecting the beam from the source fiber directly onto the detector. The reflector was then placed back into the setup and the silicon detector used to measure the total reflected power ($P_1'$). For this, the silicon detector was positioned at the focus point of the mirror. Finally the receiving fiber was installed and the power delivered by the beam bender through the receiving fiber ($P_2$) was measured.

The reflector efficiency of 87% compares well with the theoretical reflectivity of aluminum at the laser wavelength (91%). However the reflector coupling performance was disappointing, with only 1% of the source fiber power passed through to the receiving fiber exit. This means the reflector did not focus properly and failed to couple the beam into the receiving fiber.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>$P_1'$</td>
<td>$P_2$</td>
<td>$E_{REFL}$</td>
<td>$E_{CPLG}$</td>
</tr>
<tr>
<td>Power leaving source fiber</td>
<td>Power leaving reflector</td>
<td>Power leaving receiving fiber</td>
<td>Reflector efficiency, $P_1'/P_1$</td>
<td>Coupling efficiency, $P_2/P_1$</td>
</tr>
<tr>
<td>140 µW</td>
<td>122 µW</td>
<td>1.5 µW</td>
<td>87.1 %</td>
<td>0.9%</td>
</tr>
</tbody>
</table>

Next the receiving fiber was removed and the beam from the reflector projected onto a white card. The card was moved through a range of positions inside and outside the design focal point. Figure 9-12 is a series of sketches that compares the expected and observed images formed by the beam bender. When the card was placed well outside the design focus point, the projected image was expected to be of significant size and elliptical. As the card was moved toward the reflector, the image should have shrunk and grown more intense, until at the focus point it became a bright point. After the card was inside the focus, the image then should have grown back to a discernible size. In very general terms the actual image behaved as expected: the image did grow smaller as the card was moved toward the reflector, reaching a minimum size at approximately the focus distance, then growing larger as the card moved still...
Figure 9-12. Sketches showing expected and actual images obtained by projecting the reflected beam onto a white card. The reflector has figure error, so the actual image is too large for the beam bender work at high efficiency.

closer to the reflector. However the smallest image was more than a millimeter across, significantly larger than that needed for minimum-loss coupling into the receiving fiber. The projected images were also badly distorted from an elliptical shape and were surrounded by large, irregular low-intensity penumbras.

The beam bender did, in fact, bring the reflected beam to a focus. But it was a very poor one compared to the performance requirement. This suggests the machined surface contained significant form error.
9.5 **METROLOGY RESULTS**

9.5.1 **Form Error**

The Zygo New View white-light interferometer can obtain data from slopes beyond 10° when imaging a narrow region of a part at high magnification. Figure 9-13 shows an interferogram of the entire reflector surface made by stitching 72 such narrow-angle views, using the New View interferometer's internal software. It contains approximately 10 million individual data points.

![Figure 9-13. Composite interferogram of reflector, obtained by using New View white-light interferometer to stitch 72 high-magnification data sets.](image)

To determine the reflector form error the metrology data must be aligned precisely to the design surface, so that the difference can calculated point-by-point. Between the two surfaces there are 6 degrees of freedom, three associated with rotating the metrology data about each of the three coordinate axes, and three with translating the data to the correct position relative to
the design surface. Appropriate fiducials need to be machined at the same time as the part, to enable the necessary angular and linear alignment between the coordinate axes of the two surfaces.

Unfortunately, the beam bender reflector was fabricated without fiducials. This deficiency made it unrealistic to attempt to analyze the full surface. Form error was evaluated along the reflector longitudinal centerline shown in Figure 9-14, to gain some understanding of the accuracy of the machined surface. The centerline profile was extracted from the Zygo output data set using MATLAB routines developed at the PEC. Without fiducials, this gave a more accurate profile than a direct measurement using the Talysurf stylus profilometer since it was difficult to align the Talysurf measurement path coincident with the reflector centerline.

The endpoints of the reflector profile were used to rotate and level the metrology data. In the design profile, the endpoints have a Z-direction offset of 5.4 µm; this is because when the design surface was rotated into the machining coordinate system, the rotation angle was 45° instead of the actual angle of 45.03°. As can be seen in Details I and II in Figure 9-15, the transition between the reflector profile and the linear entrance/exit ramps has a finite radius. Therefore identification of the reflector endpoints in the metrology data can have inaccuracy in the upfeed direction of tens of micrometers.

\[21\] MATLAB tools developed by K. Garrard at the PEC allowed 2-D profiles to be extracted from the Zygo data set, along user-defined section lines. These MATLAB tools allowed the section lines for profiles to be be positioned more precisely than can be achieved by drawing them on the Zygo New View’s interactive display.
Figure 9-14. Measured (white-light interferometry) and design profiles along the reflector centerline. Details I, II, and III are shown in Figure 9-16.
Figure 9.15. Reflector centerline profile details (called out on Figure 9-14).
Figure 9-16. Deviation between design and measured reflector surfaces along centerline, using leveling and alignment shown in Figures 9-14 and 9-15.

The point of maximum sag for the measured profile was aligned with that of the design profile. The RMS average deviation between the two profiles was calculated, with deviation defined as

\[ \text{deviation} = Z(X,Y)^{\text{DESIGN}} - Z(X,Y)^{\text{ACTUAL}} \]  

(9-6)

Finally, RMS deviation was minimized by shifting the two profiles horizontally relative to each other, while constraining them vertically.

Figure 9-16 shows the deviation profile after minimizing the RMS form error. This was 472 nm RMS. Positive deviation means the actual surface is at a deeper depth of cut than the design surface.
9.5.2 Surface Roughness

Area-based surface roughness of the reflector was evaluated using white-light interferometry at nine locations shown in Figure 9-17. The surface roughness was calculated by finding the deviation from the best-fit plane for a 50 μm square sample area. This data was then subsequently filtered by a high-pass filter with a cutoff spatial frequency of 10 mm\(^{-1}\), to remove the long-wavelength contribution from the concave form of the part.

Table 9-3 shows the surface roughness results for the beam bender reflector. In general the surface roughness was 25 to 32 nm RMS. The roughness is comparable to that of the spherical reflector machined in hard-plated copper (20 to 30 nm RMS) which had comparable slopes. It is significantly better than for the spherical reflector machined in 2000-series aluminum alloy, which had a roughness range of 40 to 80 nm RMS.

Figure 9-17. Measurement zones for surface roughness.
Table 9-4. Area-based Surface Roughness for Ellipsoidal Reflector Surface

<table>
<thead>
<tr>
<th>Material</th>
<th>6061-T6 Aluminum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region</td>
<td>1</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>30</td>
</tr>
<tr>
<td>Region</td>
<td>4</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>32</td>
</tr>
<tr>
<td>Region</td>
<td>7</td>
</tr>
<tr>
<td>Roughness, RMS (nm)</td>
<td>26</td>
</tr>
</tbody>
</table>

9.6 ERROR ANALYSIS OF BEAM BENDER REFLECTOR

Shortly after the beam bender reflector was machined, the Nanoform axis error was mapped for another project. X-Y and Y-Z squareness issues were discovered which potentially affected the reflector’s figure error as follows:

1. Measured Y-Z squareness error was 0.003 radians (62 arc-seconds). The effect of this error was to introduce error in the depth of cut, varying linearly with crossfeed position. One half of the reflector, on one side of the X (upfeed) axis was cut deeper than required while the other half was cut shallower.

2. Measured X-Y squareness error was 0.0018 radians (366 arc-seconds). However, based on the spherical reflector results from Chapter 8 the motion program corrected for a squareness error of 0.0040 radians (825 arc-seconds). Hence the program overcorrected by 0.0022 radians, introducing an X-Y squareness error of this magnitude in the opposite direction. The effect of this error was to displace the actual tool location along the upfeed direction from the desired location, "shearing" the surface when seen in the plan view and creating astigmatism.
Another error source was found in the morphology program used to generate the motion program. A 26 μm x 4 μm toolpath ellipse was used to define the structuring element but the correct dimensions were actually 24 μm x 4 μm. This mistake caused the actual surface to be cut at a slightly tighter local radius of curvature than cut than intended, varying along the upfeed direction with the greatest error at either end of the part.

The MORPH3D morphology program was used to simulate the above errors. Squareness errors were simulated by modifying the ideal image to include the distortion in actual axis position. Equation 9-4 gives the distorted image surface for Y-Z axis squareness error and Equation 9-5 addresses X-Y squareness error:

\[
I^*(X,Y) = I(X,Y) + \beta \cdot Y \quad (9-4)
\]

\[
I^*(X,Y) = I((X + \gamma \cdot Y),Y) \quad (9-5)
\]

I(X,Y) is the Z-coordinate of the intended image surface at given coordinates (X,Y), while \( I^*(X,Y) \) is the Z-coordinate of the distorted image at the same X and Y. The X-Y and Y-Z axis squareness errors are respectively \( \gamma \) and \( \beta \), expressed in radians. Eroding the distorted image \( I^* \) simulated the machined surface including the effect of the squareness errors.

The error caused by the incorrect ellipse dimensions was checked by finding the image based on the assumed (incorrect) vibration path, then eroding it with a structuring element with the correct vibration path dimensions (see Section 5.2.2)

The error between the generated and desired surfaces was simulated for each of the three sources individually, and also for the combined case. Table 9-5 shows the maximum deviation for each case. Note the individual errors act in different directions and reach their maximums at different points on the surface, so that the combined error is smaller than the arithmetic sum of the three components.
Figure 9-18 is a plot showing the deviation profiles for the combined error, at five different crossfeed coordinates spanning the reflector's width.

Table 9-5 and Figure 9-18 report the figure error as the vertical difference between the design surface and actual surface at position (X, Y):

\[
\text{deviation} = Z(X,Y)_{\text{DESIGN}} - Z(X,Y)_{\text{ACTUAL}}
\]  

(9-6)

A positive deviation means the design surface has smaller sag than the actual machined surface, that is the surface was cut deeper than intended.

<table>
<thead>
<tr>
<th>Error Source</th>
<th>Deviation between design and actual surface</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PV</td>
</tr>
<tr>
<td>XY squareness</td>
<td>416 nm</td>
</tr>
<tr>
<td>YZ squareness</td>
<td>553 nm</td>
</tr>
<tr>
<td>Vibration path</td>
<td>196 nm</td>
</tr>
<tr>
<td>Combined</td>
<td>756 nm</td>
</tr>
</tbody>
</table>

Table 9-5. Predicted deviation from design surface caused by Nanoform squareness and motion program errors

The simulated deviation profile along the centerline (Y=0) can be compared to the measured deviation in Figure 9-16. The simulated and measured deviation profiles have similar shapes, but the actual deviation has peak-to-valley and RMS variation about 8-10 times that of the simulation. This suggests there are yet undetected sources of form error, beyond axis squareness and the vibration path dimensions in the MATLAB morphology program. Compliance of the
Figure 9-18. Estimated deviation profiles between design and predicted reflector surface, for combined axis squareness and vibration path errors. Note apparent tool feed direction is drawb opposite of that for centerline profile in Figures 9-14 and 9-15.

Y-axis riser block (see Figure 3-6) is relatively large in the upfeed direction, 1.9 MN/m or about 0.5 N per micrometer. Since machining forces are likely to be different for cutting uphill compared to cutting downhill this compliance may be a contributing factor to the form error in the reflector.

Measurement accuracy is also a major concern. It should be the first issue addressed in any future work.
10 3-D MICROSTRUCTURES MADE USING SHARP-NOSE TOOLS

10.1 INTRODUCTION

10.1.1 Limitations of Round-Nosed Tools

Round-nosed tools are limited in the types of features they can make. This is illustrated by Figure 10-1, which depicts grooves and ridges made using tools with nose radii of 1 mm and 50 μm. The round tool cross-section creates shallow curved sidewalls. Increasing the depth of cut simultaneously increases the minimum spacing between adjacent positive features, in proportion to the square root of the ratio of the depth increase. Therefore round-nose tools can only make low-aspect ratio features. The machining experiments in Chapters 7 through 9 produced positive features up to 20 μm tall and concave parts 130 μm deep, but their aspect ratios were only 0.03 to 0.15.

Figure 10-1. Comparison of grooves of equal depth made by round-nosed tools with nose radii of 1 mm and 50 μm.
10.1.2 Sharp-Nose Tools

Use of sharp-nose tools with EVAM has not been explicitly discussed in the literature.

Tools with a sharp-nosed cross-section can overcome some of the geometry limitations of round-nosed tools. Sharp-nose tools are usually “dead-sharp”, meaning the nose radius comes close to being infinitesimal—usually the radius of such tools is smaller than 100 nm.

Figure 10-2(a) is an SEM image of a dead-sharp tool with a 40° included angle between cutting edges. Figure 10-2(b) shows the size and shape of ridge and groove features that this tool might be used to make. The angle \( \theta \) of the sidewall from the horizontal is

\[
\theta = 90^\circ - \frac{\psi}{2}
\]  

(10-1)

where \( \psi \) is the included angle between the tool's cutting edges. The angle of the sidewall from the vertical is simply \( \psi / 2 \). The aspect ratio of a single pass groove made with a sharp-nose tool is \( \tan \theta \), when aspect ratio is defined as the feature height divided by the feature half-width. The tool shown in Figure 10-2 can make straight sidewalls 20° from the vertical with an aspect ratio of 2.75.

Figure 10-3 illustrates the theoretical crossfeed surface features made by a dead-sharp tool. The crossfeed feature height \( PV_{CROSS} \) is related to the crossfeed spacing \( \Delta Y \) and the sidewall angle by Equation 10-2:

\[
PV_{CROSS} = \frac{\Delta Y \cdot \tan \theta}{2}
\]  

(10-2)
Figure 10-2. (left) Sharp-nose tool with 40° nose angle (right) Grooves and positive features (ridges) possible with 40° nose angle tool.

Figure 10-3. Theoretical crossfeed surface feature height from dead-sharp tools.

At a crossfeed spacing of 1 μm, the sharp-nose tool in Figure 10-2 will produce a theoretical feature height of 1.37 μm! This compares to theoretical feature heights of only a few nanometers when round-nosed tools are used. This characteristic poor surface roughness of sharp-nose tools might be partially overcome by appropriately shaping the tip region cross-section. On the left side of Figure 10-4 is a form tool for making microstructured groove-and-ridge features. The flat bottom can machine grooves with a corresponding minimum width at the floor, but limits this tool's ability to machine a surface with crossfeed slope or curvature. The tool on the right side of Figure 10-4 has a rounded tip with a radius of
several micrometers. This permits it to machine steep-sided grooves with relatively small bottom surface roughness, and also to make sloped or curved crossfeed surfaces when an appropriate crossfeed increment is used.

![Figure 10-4. Alternate sharp-nose tool geometries, to produce better surface finish than can be created with dead-sharp tools.](image)

**10.1.3 Sharp-Nose Tool Cutting Edge Velocity**

In EVAM the work material velocity, relative to the cutting edge, is different for sharp-nose and round-nosed tools. Figure 10-5 shows a front view of the rake face for both tool geometries. The tools are moving toward the reader with horizontal velocity $X'(t) + U$, where $U$ is the upfeed velocity and $X'(t)$ is the horizontal part of the elliptical vibration motion given by Equation 2-3. $Z'(t)$ is the vertical vibration velocity of the tool given in Equation 2-4.

At micrometer-range depths of cut, the engaged portion of the cutting edge is close to the centerline. For a round-nose tool this portion of the cutting edge is almost perpendicular to the vertical, so that the projection of $Z'(t)$ onto the cutting edge is zero. The motion of the workpiece material relative to the cutting edge is orthogonal, from “front-to-back” in the upfeed direction.
For a sharp-nose tool the cutting edge makes an acute angle with the vertical centerline, and the vertical velocity can be projected along the cutting edge:

\[ Z'_{\text{edge}}(t) = Z'(t) \cdot \cos \psi \]  

(10-3)

The edge velocity is non-zero and is “lateral” along the cutting edge. The relative instantaneous motion of the work material to the cutting edge is oblique, from front to rear in the upfeed direction and also is non-zero in a direction parallel to the edge. Because \( X'(t) \) and \( Z'(t) \) are functions of time, the speed and overall direction of this oblique motion changes constantly throughout the EVAM vibration cycle.
The ramifications of this difference in cutting edge motion were not investigated. However, as will be seen in the experiments described in Section 10.2 and Section 10.4, sharp-nosed tools showed different behavior from round-nosed tools in tool wear, burr creation, and surface roughness. It seems proper to mention this difference in the cutting process, as a possible factor to be considered in future research.

10.2 TETRAHEDRON ARRAYS

10.2.1 Part Design and Fabrication

A dead-sharp tool was used with the Ultramill to machine an array of regular tetrahedrons 80 μm tall on a 112 μm parallel pitch. The aspect ratio of these features, defined as the height over the half-width, is 1.4. This project was a feasibility study to see if EVAM can make replication masters for micro-corner cube arrays.

The tetrahedrons were made by ruling three sets of intersecting grooves using a tool with a nose angle $\psi$ of 70.53°. This specific angle is required to produce regular tetrahedrons, and Appendix P gives the derivation. The tool rake angle was 0° with a flank clearance angle of 10°. SEM inspection at 10,000x showed the nose radius was less than 50 nm, confirming that the tool was effectively “dead-sharp”.

Figure 10-6 shows how the tetrahedrons were made by rotating the Nanoform’s spindle after cutting each set of grooves. A first set of parallel grooves was cut, using a local coordinate system based on a datum point located a distance $R$ from the center of rotation (Cut I in Figure 10-6). The spindle was then rotated through a nominal 60° angle. The part was next translated along the X and Y axes through a distance $\Delta X = R \cos \theta$ and $\Delta Y = R \sin \theta$ to bring the tool back to the datum point for the groove coordinate system, where $\theta = 60^\circ$. A second set of

---

22 This project was sponsored by Oak Ridge National Laboratory (PO 4000048725).
Figure 10-6. Making tetrahedrons by rotating spindle 60° between each set of cuts.
grooves was then machined (Cut II). The part was again rotated through another 60° and the tool translated through $\Delta X$ and $\Delta Y$, before machining the final set of grooves (Cut III). The spindle motor had insufficient torque to hold a fixed position so the spindle was rotated by hand and locked into place. The spindle axis encoder was used to measure the actual angle of rotation which was found to have a worst case error of 0.043° for the intended 60° value. This resulted in a positioning error of 7 μm for the largest value of $R$ used in the cutting experiments, but corrections were made in the $\Delta X$ and $\Delta Y$ translation before machining the next set of grooves.

For all three groove sets to intersect at the same point, the tool tip needed to be aligned to the center of the spindle with micrometer-range precision. A customary method of centering a tool is to cut a center plug or a spherical feature, then measure the form error with a stylus profilometer or white-light interferometer. The direction and magnitude of the centering error can be derived from the shape and size of the form error. For two reasons this method was not used. First, the Ultramill needed to run continuously during the centering operation to maintain thermal stability, and it was judged that the running time would be excessive while measuring the setup parts, calculating centering corrections, and so forth (several parts would likely need to be cut, to zero in on the center). Second, with the dead sharp tool submicrometer crossfeed spacing would be required to make a test part where the crossfeed surface feature height would be small enough to permit measurement and for which the surface roughness would not saturate the form error. This implies a long machining operation, increasing the risk of damaging or prematurely wearing the tool. Therefore an alternate method was employed which used the Y-axis camera to measure the position shift of test grooves relative to an assumed center point when the spindle was rotated through half a revolution. This method is described in Appendix Q. The accuracy of this approach was limited by the camera resolution of 3 μm / pixel.
Since the tetrahedron geometry was created by the tool cross-section, the individual cutting passes were constant depth and the machining program was written as a series of groove cuts. The 80 μm deep grooves were made by a series of successively deeper cuts in the same crossfeed location, as illustrated by Figure 10-7. Different combinations of roughing and finish cuts were tried. These are summarized in Table 10-1.

![Figure 10-7. Successive tool passes in same groove at increasing depth of cut.](image)

**Table 10-1: Schemes for Groove Cuts for the Tetrahedron Arrays**

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Material</th>
<th>Total # of Cuts</th>
<th>Roughing Cuts</th>
<th>Finish Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copper – no finish cuts</td>
<td>OFHC Copper</td>
<td>8</td>
<td>8 at 10 μm</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>Hard Plated Copper</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Copper – finish cuts</td>
<td>OFHC Copper</td>
<td>11</td>
<td>7 at 10 μm</td>
<td>1 at 2 μm</td>
</tr>
<tr>
<td></td>
<td>Hard Plated Copper</td>
<td></td>
<td>1 at 4 μm</td>
<td>1 at 1 μm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 at 3 μm</td>
<td></td>
</tr>
<tr>
<td>SS – deep roughing cuts</td>
<td>17-4 PH SS</td>
<td>11</td>
<td>7 at 10 μm</td>
<td>1 at 2 μm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 at 4 μm</td>
<td>1 at 1 μm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 at 3 μm</td>
<td></td>
</tr>
<tr>
<td>SS – shallow roughing cuts</td>
<td>17-4 PH SS</td>
<td>15</td>
<td>12 at 6 μm</td>
<td>1 at 2 μm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 at 5 μm</td>
<td>1 at 1 μm</td>
</tr>
</tbody>
</table>
A 24 μm x 4 μm elliptical vibration path was assumed. Programmed upfeed velocity was 0.5 mm/s at $f = 1000$ Hz, or $HSR = 0.0058$. These parts were made with open-circulation cooling and half-round stacks installed in the Ultramill (Configuration III-A in Table 3-1).

### 10.2.2 Tetrahedron Results – SEM Imagery of Copper Parts

Tetrahedrons were made in C1100 oxygen-free high-conductivity (OFHC) copper, hard-plated copper (a material designed specifically for machining of microgrooves), and 17-4 PH stainless steel. Results for the OFHC and hard-plated copper parts are discussed here. The stainless steel results are discussed in Section 10.2.3.

Figure 10-8 shows SEM images of tetrahedron features made in hard-plated copper. The sequence used to make these features was 7 cuts each at 10 μm depth of cut, followed by finishing cuts of 4, 3, 2, and finally 1 μm. The edges of the 80 μm tall trihedrons were effectively burr-free. The small interstitial tetrahedrons resulted from centering error which was greater than desired due to the accuracy limitations of the camera-based centering method. These features are less than 8 μm tall. Even at 4000x (lower right detail in Figure 10-8) no burr was visible on the small tetrahedrons. Edges of these features were sharp, with a radius appearing to be significantly smaller than submicron size. The 80 μm tetrahedrons are the tallest positive surface feature made using the Ultramill.

Figures 10-9 and 10-10 compare parts made in OFHC copper and hard-plated copper respectively, with and without finishing cuts. The SEM images in Figure 10-9 show parts made in OFHC copper. In the left-hand pair of images the tetrahedrons were made using a set of 8 roughing cuts of 10 μm each, and no finish cut. These parts have noticeable burr at the feature edges. The right hand images, show the effect of using a sequence of finish passes in lieu of the final 10 μm rough cut. While a small burr remains, the parts were improved by using the finish cuts, which produce small machining forces.
Figure 10-8. SEM images of tetradron array machined in hard-plated copper. Small interstitial features were caused by tool centering error.

Figure 10-9. SEM images of tetrahedrons machined in OFHC copper. (left) Burr resulted when only roughing cuts were used. (right) Burr was reduced by using finishng cuts.
Figure 10-10 shows parts made in hard-plated copper. Using only roughing cuts produced a small edge burr. This disappeared when the finishing cuts were used, as previously noted.

### 10.2.3 Tetrahedron Results – Burr Formation

Figure 10-11 is a series of SEM images comparing tetrahedrons made in hard-plated copper, OFHC copper, and 17-4 PH stainless steel. These parts were all cut with the same machining conditions (ellipse size, vibration frequency, upfeed velocity, and roughing / finishing cut sequence). While some burr formation is seen in the soft OFHC copper, severe burring occurred when cutting the stainless steel material. Machining the stainless steel with other roughing / finish cut combinations did not reduce the amount of burr.

Table 10-2 compares properties of these materials which may affect burr formation—Vickers hardness, and elongation index (indicative of ductility). The values of these properties are also shown in Figure 10-11 under the corresponding SEM images. Burr severity appears to correlate with material ductility, but not with hardness.
Figure 10-11. Burr formation in tetrahedron features made from different materials.

Table 10-2. Properties of Materials Used for Tetrahedron Fabrication Experiments

<table>
<thead>
<tr>
<th>Material</th>
<th>Hard-Plated Copper</th>
<th>OFHC Copper (H04 temper)</th>
<th>17-4 Stainless Steel (annealed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temper</td>
<td>--</td>
<td>H4</td>
<td>Annealed</td>
</tr>
<tr>
<td>Hardness (Vickers)(A)</td>
<td>320</td>
<td>110</td>
<td>280</td>
</tr>
<tr>
<td>Ductility Index (% elongation)</td>
<td>&lt; 4 (B)</td>
<td>11 (C)</td>
<td>15 (D)</td>
</tr>
</tbody>
</table>

A Hardness tests by D. E. Brehl  B from R.O. Scattergood  C Ref [10]  D Ref [18]

The dead-sharp tool geometry itself may play a role in burr susceptibility. Figure 10-12 presents an SEM image of the edge of the "wing" feature on the Thunderbird part of Section 1.4, which was made using a 1 mm nose radius round-nosed tool. This part was cut in the same 17-4 PH stainless steel as the tetrahedrons. The depth of cut for making the raised Thunderbird feature was 2.5 μm compared to the 1 μm finish cut used for the tetrahedron features in Figure 10-11. The two parts were made with the same vibration frequency, ellipse size, and upfeed velocity. No burr is visible on the Thunderbird wing. The only significant difference between the two parts is the tool geometry.
Cutting edge wear was found on the dead-sharp tool after machining stainless steel (see Section 10.2.4). This does not appear to be a contributor to burring. The very first set of tetrahedrons machined in stainless steel showed severe burr. Also after multiple attempts to make stainless steel tetrahedrons, the worn tool was then used to machine the microchannel stream combiner described later in Section 10.3.2. This part was cut in hard-plated copper, and intersections between feature faces were visibly burr-free (see Figure 10-19).

In Section 10.2 it was noted that, unlike round-nosed tools, in EVAM dead sharp tools possess a component of workpiece motion parallel to the cutting edge. This cutting edge motion results in oblique cutting (while round-nosed tools cut orthogonally), broadly similar to milling. This may cause different cutting mechanics, with an effect on machining forces and chip formation that facilitates burr formation.

**Figure 10-12.** Binary thunderbird feature machined in 17-4 PH stainless steel using 1 mm radius round-nosed tool. Detail of wing feature shows negligible burr formation.
10.2.4 Tetrahedron Results – Tool Wear

The dead-sharp tool was inspected for wear after completing the tetrahedron machining tests. At this point the accumulated machining distance for the tool tip was 1.6 m in copper and 2.0 m in stainless steel. The actual sliding distance (due to the multiple passes in EVAM) was approximately 28 m for copper and 36 m in stainless steel.

The tool was examined using a Field Effect Scanning Electron Microscope (FESEM) which is capable of very high resolution images at magnifications up to 100,000x. Figure 10-13 shows FESEM images of the tool from two angles. The tool condition when new was compared to that following stainless steel machining. After machining, the cutting edge radius showed noticeable increase due to wear. The tip of the tool also wore back from the rake face, consistent with the tip region having the most cumulative contact distance with the material, due to the repeated cuts made at increasing depth in each raster pass. This sequence of progressively deeper cuts also appears responsible for the stepped or multi-layer wear pattern with increasing distance moving away from the tool tip.

Although tool wear occurred, it was relatively smooth without chipping or cracking of the cutting edge. In this regard, wear of the dead-sharp tool was broadly similar to Cerniway’s experiment cutting hardened tool steel with a round-nosed tool using EVAM. The SEM images in Figure 10-14 show Cerniway's tool after machining W2 tool steel for 20 m raster distance or ten times the distance cut by the dead-sharp tool in stainless steel. (The total sliding distance was 190 m, or five times that of the dead-sharp tool in steel). The wear along the cutting edge was smooth and non-detrimental to surface finish. This wear appears to be smoother, more regular, and of a greater dimensional extent than on the cutting edges of the dead-sharp tool. This may possibly be due to differences in wear mechanisms arising from the different tool geometries. However, the materials in the two cases were different and Cerniway's tool accumulated 10 times the machining distance. Finally, he cut flat grooves at
Figure 10-13. FESEM images showing wear on dead sharp diamond tool after accumulating 2 m raster machining distance in 17-4 PH stainless steel and 1.6 m in copper.
Figure 10-14. Wear of round-nosed diamond tool after it was used to cut tool steel with EVAM. SEM image taken after 20 m raster machining distance. [8]

Figure 10-15. SEM images comparing wear for dead-shap and round-nosed diamond tools after approximately 2 m raster machining distance in 17-4 PH stainless steel.
constant depth while the dead-sharp tool made repeated passes at progressively increasing depth. In general, both tool geometries avoided irregular, fractured cutting-edge damage characteristic of conventional machining of ferrous metals with diamond tools. Sharp-nose tools may be more susceptible to wear than round-nosed tools\(^ {23} \). Figure 10-15 compares SEM images of the round-nosed tool used to make the Thunderbird feature, and the dead-sharp tool after the stainless steel tetrahedron efforts. The tools were both imaged after each had cut for 2 m raster distance. No wear can be discerned on the round-nosed tool, in contrast to the evidence of wear visible on the cutting edge and rake face of the dead-sharp tool seen in Figures 10-13 and 10-15.

### 10.3 Additional Machining Experiments with Sharp-Nose Tools

#### 10.3.1 Grooves Made with 40° Nose-Angle Tool

Figure 10-16 is an SEM image showing a sharp-nose tool with \( \psi = 40^\circ \). The tip is rounded with a 2 \( \mu \)m radius to produce relatively flat surface roughness at reasonable crossfeed spacing. Figure 10-17 shows SEM images of single-pass grooves made this tool. The material was hard-plated copper. Even at 3000x magnification, the groove edges are free from burr. Unfortunately the tool tip broke after these grooves were made, preventing further experiments with this tool geometry.

---

\(^ {23} \) During preliminary review of this dissertation, T. A. Dow suggested that dead-sharp tools more wear near the tip because of the tool geometry. Diamond wear rate, expressed as material volume lost per meter of sliding distance, is assumed constant regardless of tool geometry. The dead sharp tip wears more rapidly than a round-nosed tool because the volume lost is concentrated in a small region.
Figure 10-16. Sharp nose tool with 40 degree included nose angle and rounded tip. SEM image

Figure 10-17. Details of grooves made in hard-plated copper using the 40° nose-angle tool shown in Figure 10-16. Note absence of burr. SEM images.

10.3.2 Microchannel Stream Combiner

Following the tretrahedron experiments, the 70° nose angle dead-sharp tool was used to make the microchannel stream-combiner shown in the sketch in Figure 10-18. This configuration is frequently used for combining streams in microfluidics experiments. The part was made by manually rotating the DTM spindle after machining each reservoir and branch channel. The features were 25 μm deep and were made using two 10 μm and one 5 μm cuts. The upfeed
Figure 10-18. Design sketch for microchannel stream-combiner made with sharp-nose tool.

Figure 10-19. SEM images showing results of microchannel stream-combiner fabrication experiment. (left) Channel junction error. (right) Detail of junction showing ribbon burrs on channel floor.
velocity was 0.33 mm/s ($HSR = 0.005$), the workpiece material was hard-plated copper, and
the Ultramill was cooled by the open-circulation arrangement (Configuration III-A defined in
Table 3-1).

The SEM images in Figure 10-19 show how tool centering error prevented the branch channels
from intersecting properly. Although the part failed to conform to the layout sketch in Figure
10-18 several important results were nonetheless obtained.

Although the tool cutting edges possessed visible wear after machining stainless steel, the SEM
images in Figure 10-19 show there is no burr at the corners where the channels meet. This
reinforces the hypothesis that the worn tool was not a primary cause for severe burring in the
stainless steel tetrahedrons, but rather that material properties such as ductility are important
when using sharp-nose tools with EVAM.

The high-magnification detail in Figure 10-19 shows filamentary "ribbon" burrs on the floor
of the channel. The crossfeed spacing for the floor was reduced to 250 nm in an effort to
create a relatively flat finish using a dead-sharp tool. At this spacing, theoretical feature height
was 280 nm PV, per Equation 10-2. The tight crossfeed increment created thin ridges of
material remaining between the individual cuts which were unable to withstand the high local
machining forces. The ridges then deformed or failed but irregularities in the machining
process allowed them to stay attached to the workpiece in some places, creating the burr.

White light interferometry was used to examine the form and surface roughness of the straight,
steep sidewalls created by the sharp-nose tools. The part was sectioned and tilted so that the
sidewall of the reservoir pocket was normal to the interferometer beam. Figure 10-20 shows
a cross section profile of the sidewall and floor obtained from interferometry. This profile
confirms that the sidewall produced by the sharp edge tool is straight (flat) with no discernible
form error.
Figure 10-20. Crossfeed profile showing linear form of micromixer reservoir wall and floor, obtained by white-light interferometry.

![Crossfeed profile showing linear form of micromixer reservoir wall and floor, obtained by white-light interferometry.](image)

Figure 10-21. *(left)* Upfeed surface profile of sidewall made with sharp-nose tool. *(right)* Autocovariance plot for the sidewall profile.

![Upfeed surface profile of sidewall made with sharp-nose tool. Autocovariance plot for the sidewall profile.](image)

Figure 10-21 is a profile in the upfeed direction for the sidewall. It shows that the sidewalls made with the sharp-edge tool and EVAM are not smooth but have a significant surface roughness of more than 120 nm RMS. This surface roughness is caused by regularly spaced features in the upfeed direction. The spacing interval is 6.7 μm, which is substantially greater than the upfeed increment $F_{UP}$ of 0.33 μm/cycle. From the associated autocovariance plot, the equivalent temporal frequency of these features is found to be approximately 50 Hz. A round-nosed tool cutting grooves in the same material produced a surface roughness of only 8-12 nm RMS when operated at the same vibration frequency and at comparable values of $F_{UP}$.
between 0.2 and 1 \( \mu \text{m}/\text{cycle} \) (see Figures 5-3 and 5-5). These round-nose tool surface features were at a spacing corresponding to a temporal frequency of 10 Hz per the autocovariance analysis in Figure 5-7.
11 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

The scope of this dissertation is broad, ranging from equipment development, motion control programming for EVAM, and functional microstructure machining using round-nosed and sharp-nosed diamond tools. Major conclusions for each area of effort are presented in Section 11.1. Recommendations for future work are given in Section 11.2.

11.1 CONCLUSIONS

Equipment Development

Hydrostatic oil-bearing Y-Axis
An oil-bearing Y-axis was installed on the Nanoform 3-axis diamond-turning machine, in lieu of the original air-bearing axis with ball-screw drive. This resulted in:

- Increased axis stiffness in the depth of cut direction from 0.5 to 2.5 MN/m.

- Surface roughness in flats improved to 8-12 nm RMS, compared to 15-25 nm RMS for the air-bearing axis.

- Crossfeed movement precision was improved to approximately 10 nm from >1 µm for the air-bearing axis

Ultramill Thermal Stability Improvement (Closed-Circulation Cooling)
The Ultramill was converted from open-circulation to closed-circulation cooling. This improved temperature stability of the Ultramill and improved form error machined in parts. Because the original diaphragm pump was retained, surface finish deteriorated because of pump-generated pressure pulsations.
In detail:

- Piezo stack temperature stability was improved to 0.4°C variation (peak to peak), compared to irregular multi-degree excursions seen with open circulation cooling.

- Closed-circulation form error was equal to or better than that with open-circulation. A typical 3-D feature, that took almost an hour to machine, had form error reduced from 500 nm to only 80 nm peak-to-valley.

- For an optimal coolant flow rate of 0.4 L/min, closed-circulation cooling produced a surface roughness of 8-12 nm RMS, the same as for open-circulation flow. At greater flow rates value pressure pulsations from the existing diaphragm pump caused significant deterioration of surface roughness and form error.

**Process Development**

**EVAM Kinematics and Theoretical Surface Roughness**

- Kinematic relationships determined by previous researchers for EVAM with horizontal feed motion were extended to include vertical feed motion (e.g. cutting an upward or downward slope).

- Theoretical surface feature height (equivalent to peak-to-valley surface roughness) was simulated for EVAM with vertical feed motion. Feature height grew worse with increasing slope, when all other machining parameters were kept constant.
**Motion Program Planning**

An innovative method was developed for finding the tool path required to raster machine an arbitrary specified surface. It finds cutter compensation for the tool edge profile and the elliptical EVAM vibration path, and generates a G-code motion program for use by the Nanoform diamond turning machine.

- The method adapts surface morphology operations used for image construction in scanning probe microscopy and for image processing. Standard MATLAB functions are available for these morphology operations.

- The morphology method was validated for multiple test cases where the required toolpath could be calculated analytically. Discretization of the desired surface and structuring element caused an estimated form error of 5 to 50 nm (peak-to-valley) when a convenient 1 μm square grid was used. Form error approached zero as grid resolution was made smaller.

- The morphology method was used to show that distortion of the elliptical vibration path, caused by workpiece feed motion at up to 10° tilt, could be ignored when the upfeed velocity was less than 0.5 mm/s at a 1000 Hz Ultramill frequency (equivalent to $F_{UP} < 0.5 \mu m / cycle$ or $HSR < 0.006$).
3-D Microfeatures Made Using Round-Nosed Tools

General Comments
Tilted planar and sculpted 3-D parts with convex and concave geometry were machined using EVAM with a round-nosed tool of 1 mm nose radius.

- Aspect ratios up to 0.15 (height to width) were achieved, compared to 0.01 for most parts described in the literature.

- Features were machined with heights to 20 µm and depths to 130 µm. Most parts previously made with the Ultramill were binary features with micrometer-range vertical dimensions.

- Slopes as great as 10° were typical. The maximum downward slope was limited by the tool clearance angle.

- Material removal rates on finish cuts was 13,000 to 23,000 µm³/s, a similar magnitude as processes such as laser ablation or ultraprecision micromilling.

Optical Components
A concave spherical reflector was machined in hard-plated copper and in a 2000-series aluminum alloy. An off-axis section of an ellipsoid was machined in 6061-T aluminum, to serve as the focusing reflector in a fiber optic beam-bender.

- Form error in the ellipsoidal beam-bender was estimated to be almost 500 nm RMS along the longitudinal centerline. This prevented the beam bender from achieving satisfactory coupling efficiency, due to poor focusing of the reflected light beam into the receiving fiber.
• Lack of fiducials on the beam-bender reflector inhibited assessment of the form error and determination of its causes.

• Although performance of the beam-bender was disappointing, error causes unrelated to the Ultramill were identified: large axis squareness error in the Nanoform turning machine, and a programming error in the motion program generation code.

No issues have been identified to prevent EVAM from machining optical surfaces to an acceptable level of accuracy, provided the Nanoform axes are aligned and that good precision engineering practice is followed throughout setup and motion programming.

• Form error for the concave spherical reflector was 62 nm RMS.

• Surface roughness on sloped reflector surfaces in hard-plated copper and 6061 aluminum was 20-30 nm RMS. Roughness for a different 2000-series aluminum alloy was much worse, 40-80 nm RMS and a pebbly surface was created.

Other Observations
• Planar surfaces showed concave rather than linear profiles, when cut on a downslope with a slope near the clearance angle of the tool. This may be due to interference between the workpiece and the tool.

• Surface roughness on sloped planar surfaces (cut going uphill) were worse (40 nm RMS) than for horizontal surfaces on the same part (~20 nm RMS).
3-D Microfeatures Made Using Sharp-Nose Tools

Sharp-nose tools were used to make tetrahedron structures, and cavity features (grooves, channels, and pockets) with straight sidewalls.

Micro-Structure Fabrication

Tetrahedrons as large as 80 µm tall (112 um pitch) and as small as ~5 um tall were machined in OFHC copper, hard-plated copper, and 17-4 PH stainless steel, using a dead sharp tool.

- Vertical aspect ratio was 1.4 (feature height divided by half-width).

- Feature geometry was produced by the cutting edge angle (cross section of the tool).

- Both the large and small features were burr-free with very sharp edges, for parts cut in hard-plated copper. Slight burr occurred on edges of parts made in OFHC copper.

- Severe burr, sufficient to render the part non-functional, occurred on parts made in stainless steel.

A simple microchannel stream combiner array was also machined with the dead-sharp tool, in hard-plated copper.

- 25 um deep cavity features (reservoirs and connecting channels) were produced.

- The part’s steeply sloped sidewalls appeared flat and straight in white-light interferometry, but they had a large surface roughness of 122 nm RMS. The spatial frequency of the roughness features was unexpected given the horizontal speed ratio ($HSR$), and differed from previously identified spatial frequencies for round-nosed tools.
**Burr Susceptibility**

- Burr severity correlated with increasing material ductility. Parts cut in hard-plated copper (elongation index = 4%) had no burr while parts made in 17-4 PH stainless steel (elongation index = 15%) were badly burred.

- Filamentery “ribbon burrs” were produced by the dead-sharp tool on flat channel floors, owing to the submicrometer crossfeed spacing needed to limit surface roughness. This produced micrometer-thick residual features between the cutting passes, which were prone to breaking under cutting and cleaning forces.

- Sharp-nose tools may be more likely to create burr than round-nosed tool, in the same material. Severe burr occurred on tetrahedrons machined in 17-4 PH stainless steel with a dead-sharp tool, but burr was not detected on edges of a binary feature made in the same material using a round tool.

- Sharp-nose tools in EVAM have a component of motion along the cutting edge, unlike round-nosed tools which maintain orthogonal cutting. This may change the mechanics of cutting and chip formation, affecting susceptibility to burr formation.

**Tool Wear vs. Tool Geometry**

- Both round-nose and sharp-nose tools develop cutting edge wear when used to machine ferrous metals, however the wear does not seem to be detrimental to surface finish or a source of burr generation.

- At a short cumulative raster distance (2 m) cutting in stainless steel, a dead sharp tool showed nose and cutting edge wear. A round-nosed tool did not appear to show wear, for the same raster distance cutting in the same material.
Other Observations

Flat single grooves were cut at constant 1000 Hz frequency over a range of upfeed velocities up to 8 mm/s, to investigate upfeed surface roughness caused by EVAM features.

- Regardless of upfeed velocity, the measured surface roughness was much larger than the theoretical value predicted by machining geometry.

- There were two regimes of surface roughness behavior, with the transition at an upfeed velocity between 1 and 2 mm/s ($F_{UP} = 1$ to $2 \, \mu m/\text{cycle}$ for 1000 Hz)

- At upfeed velocities below 1 mm/s, external influences such as axis vibration and/or coolant pump vibration dominated the features produced by EVAM. Surface roughness was 8-12 nm RMS, regardless of upfeed velocity.

- At upfeed velocities above 2 mm/s, the surface roughness increased with increasing feed velocity.

11.2 RECOMMENDATIONS FOR FUTURE WORK

Equipment Development

Improving surface finish on parts machined with EVAM is a major area for improvement. Two hardware actions should have a significant impact:

- Suppress coolant pump pressure pulsations by installing a pulsation dampener in the closed-circulation cooling circuit. If possible, replace the existing positive-displacement diaphragm pump with a continuous flow pump. This could mean changing the coolant fluid, as the present Fluorinert dielectric liquid is incompatible with the seal materials used in the centrifugal pump available for the Thermocube.
Investigate methods for damping the Ultramill excitation of the Nanoform Y-axis. Possible methods include making the base block out of a material with damping properties such as gray cast iron, or installing a shim made from a dissipative material between the Ultramill and the axis.

**Microstructure Creation**

- Include fiducials on machined parts, to facilitate metrology and to enable alignment of the measurements with design data, so that form error analysis can be accurately executed.

- Investigate and resolve appearance of concave form error on planar downslopes, when inclination is close to the tool clearance angle.

- Investigate surface roughness made by Ultramill on sloped surfaces (sculpted and planar geometry).

**Materials Investigations**

- Perform more experiments in making microstructures in ferrous materials with large vertical dimensions and 3-D geometry.

- Investigate use of EVAM to make 3-D sculpted geometry in brittle materials of interest (e.g. SiC, CaF, tungsten carbide).

- Investigate surface roughness and quality generated in different alloys at varying slope angles (this follows on the experience in Section 8.3.2 with the 2000-series aluminum which had a roughness of 6 nm RMS when faced by diamond turning, but which was 40-80 nm RMS on sloped surfaces made by EVAM).
Sharp-Nose Tools

Sharp-nose tools are valuable for making high-aspect ratio features with straight sidewalls. The actions suggested below follow directly from the findings of this research.

- Experiment with sharp-nose tools with rounded tips, to create flat channel floors with low surface roughness, and avoiding the “ribbon burrs” generated by dead-sharp tools.

- Continue investigation of propensity for burr formation when sharp-nose tools are used.

- Investigate tool wear for sharp-nose tools as compared to round-nosed tools.
12 REFERENCES


[34] Mathworks, Inc., on-line product guide for MATLAB Image Processing Toolbox, 
http://www.mathworks.com/products/image/


[38] Navitar Inc, "12X Zoom Lens Product Information Brochure", 
http://machinevision.navitar.com/pages/product_information/hi_mag_zoom_lenses/12x_overview.cfm


[42] Physik Instrumente Website, "Basic Introduction to Nanopositioning with Piezoelectric Technology", http://www.physikinstrum


13 APPENDICES
13.1 APPENDIX A: MODELING OF EVAM SURFACE FEATURES ON A SLOPED SURFACE

The theoretical surface features generated by EVAM when machining a sloped surface were simulated using MATLAB. This model determines the path taken by the diamond tool tip through several successive vibration cycles by calculating the instantaneous position $X(t)$ and $Z(t)$, using Equations 2-11 and 2-12:

$$X(t) = A \cdot \cos(\omega \cdot t) + U \cdot t \quad (2-11)$$

$$Z(t) = B \cdot \sin(\omega \cdot t) + V \cdot t \quad (2-12)$$

where $A$ and $B$ are respectively the vibration ellipse semi-major and semi-minor axes lengths, $U$ and $V$ are the horizontal and vertical feed velocity, $t$ is time, and $\omega = 2\pi f$ is the angular vibration frequency.

MATLAB calculates tool tip positions at discrete points, but Equations 2-11 and 2-12 are continuous functions. For computational purposes, points on successive passes must have the same value of $X$, permitting computationally-simple determination of which points lie on the machined surface and which are in air. Therefore Equation 2-11 needs to be solved for $t(X)$. Since the equation is transcendent, this solution is found numerically for each $X$ using a recursion method. With the resulting values of $t$, Equation 2-12 can then be solved for $Z(t)$, which is equivalent to finding $Z(X)$.

A plot of the tool tip position during the cutting pass is thus generated as a set of points $(X, Z)$. The theoretical machined surface can be found by examining all upfeed positions $X$ where the toolpath has been calculated. For each vibration cycle the model calculates the tool path over an interval $X = -A$ to $A + 0.5*F_UP$, where $F_UP$ is the upfeed interval defined in Equation 2-22. This interval is the length of the lower half of the toolpath. See Figure A-1.
Figure A-1: Two consecutive tool passes modeled by MATLAB. The lower half of each tool pass is shown. The lower half of each tool pass is defined over a length $2A + 0.5F_{UP}$. Starting point for the second pass is displaced $F_{UP}$ in the upfeed direction.

Figure A-2. Successive calculated tool passes, with starting points $F_{UP}$ apart. Note how overlapping tool passes produce multiple points $(X,Z)$ for each value of $X$. 
Successive tool pass have the same interval length but are displaced $F_{up}$ from the most recent preceding pass. See Figure A-2.

Typically, at each value of $X$ there will be more than one point $(X,Z)$, due to overlapping elliptical passes (Figure A-2). A comparison test is used to determine the point with the smallest value of $Z$. This point represents the theoretical machined surface at $X$, as shown in Figure A-3.

![Figure A-3](image)

**Figure A-3.** Theoretical surface, calculated by removing all but the bottommost points at each value of $X$, from the overlapping tool pass traces in Figure A-2 (b).

The theoretical surface roughness is defined as the outward normal peak-to-valley feature height. The points comprising the theoretical surface are leveled by rotating through the opposite angle as the nominal slope of the surface, $\chi$. This angle is the same magnitude as the angle of feed motion

$$\chi = \arctan \frac{V}{U}$$

(2-17)
After the surface has been rotated, the theoretical surface peak-to-valley height can be determined, by identifying the point with the largest value of $Z$ (in the rotated coordinate system).

### 13.1.1 Appendix A.1- MATLAB listing for simulation of EVAM features on a sloped surface

```matlab
dx=0.01;
X=-11:dx:11.1.;

time=zeros(1,length(X));

U=200;
A=11;
B=2;
V=54.9;
f=1000;
omega=2*pi*f;
phi=-pi;

X(1)=-11;
time(1)=0;

for index=2:length(X)
    clear xcheck
    time(index)=time(index-1);
    xcheck=A*cos(time(index)*omega+phi)+U*time(index);
    tol=X(index)-xcheck;

    while abs(tol)>.0001;
        dt=(X(index)-xcheck)/(U+omega*A);
        time(index)=time(index)+dt;
        xcheck=A*cos(time(index)*omega+phi)+U*time(index);
        tol=X(index)-xcheck;
    end
    index
    time(index)
end
```
\begin{verbatim}
figure(1)
plot(time, X)

figure(2)
stem(time)
Z=B*sin(time*omega+phi)+V*time;
figure(3)
plot(X,Z)

figure(4)
plot(time,X)

N=length(time);
figure(5)
stem(time)
passes=10;
delta=(U/f)/dx;
M=(passes-1)*delta+N;
clear index

X=X*1/dx;
X=round(X);
X=dx*X;

x=zeros(passes,M);
z=zeros(passes,M);

for index=1:N
    x(1,index)=X(index);
    z(1,index)=Z(index);
end
for index=N+1:M
    x(1,index)=x(1,index-1)+dx;
    z(1,index)=NaN;
end
figure(6)
plot(x(1,:),z)

length(x(1,index))
length(M)

zcheck=Z-(z(1,1:N))
figure(7)
stem(zcheck)
\end{verbatim}
\[ x(2,:) = x(1,:); \]
for index = 1:delta
    \[ z(2,index) = \text{NaN}; \]
end
for index = N+delta+1:M
    \[ z(2,index) = \text{NaN}; \]
end
clear index

for index = delta+1:N+delta
    \[ z(2,index) = z(1,index-delta)+V/f; \]
end
figure(8)
plot(x(1,:),z(1,:),x(2,:),z(2,:))

\[ x(3,:) = x(1,:); \]
for index = 1:2*delta
    \[ z(3,index) = \text{NaN}; \]
end
for index = N+1+2*delta:M
    \[ z(3,index) = \text{NaN}; \]
end
for index = 2*delta+1:N+2*delta
    \[ z(3,index) = z(1,index-2*delta)+2*V/f; \]
end
figure(9)
plot(x(1,:),z(1,:),x(2,:),z(2,:),x(3,:),z(3,:))

\[ x(4,:) = x(1,:); \]
clear index
for index = 1:3*delta
    \[ z(4,index) = \text{NaN}; \]
end
for index = N+1+3*delta:M
    \[ z(4,index) = \text{NaN}; \]
end
for index = 3*delta+1:N+3*delta
    \[ z(4,index) = z(1,index-3*delta)+3*V/f; \]
end
figure(10)
plot(x(1,:),z(1,:),x(2,:),z(2,:),x(3,:),z(3,:),x(4,:),z(4,:))

\[ x(5,:) = x(1,:); \]
clear index
for index=1:4*delta
    z(5,index)=NaN;
end
for index=N+1+4*delta:M
    z(5,index)=NaN;
end
for index=4*delta+1:N+4*delta
    z(5,index)=z(1,index-4*delta)+4*V/f;
end

x(6,:) = x(1,:);
clear index
for index=1:5*delta
    z(6,index)=NaN;
end
for index=N+1+5*delta:M
    z(6,index)=NaN;
end
for index=5*delta+1:N+5*delta
    z(6,index)=z(1,index-5*delta)+5*V/f;
end

x(7,:) = x(1,:);
clear index
for index=1:6*delta
    z(7,index)=NaN;
end
for index=N+1+6*delta:M
    z(7,index)=NaN;
end
for index=6*delta+1:N+6*delta
    z(7,index)=z(1,index-6*delta)+6*V/f;
end
figure(11)
plot(x(1,:),z(1,:),x(2,:),z(2,:),x(3,:),z(3,:),x(4,:),z(4,:),x(5,:),
z(5,:),x(6,:),z(6,:),x(7,:),z(7,:))

x(8,:) = x(1,:);
clear index
for index=1:7*delta
    z(8,index)=NaN;
end
for index=N+1+7*delta:M
    z(8,index)=NaN;
end
for index = 7*delta+1:N+7*delta
    z(8,index) = z(1,index-7*delta) + 7*V/f;
end

x(9,:) = x(1,:);
clear index
for index = 1:8*delta
    z(9,index) = NaN;
end
for index = N+1+8*delta:M
    z(9,index) = NaN;
end
for index = 8*delta+1:N+8*delta
    z(9,index) = z(1,index-8*delta) + 8*V/f;
end

x(10,:) = x(1,:);
clear index
for index = 1:9*delta
    z(10,index) = NaN;
end
for index = N+1+9*delta:M
    z(10,index) = NaN;
end
for index = 9*delta+1:N+9*delta
    z(10,index) = z(1,index-9*delta) + 9*V/f;
end
figure(12)
plot(x(1,:), z(1,:), x(2,:), z(2,:), x(3,:), z(3,:), x(4,:), z(4,:), x(5,:),
     z(5,:), x(6,:), z(6,:), x(7,:), z(7,:), x(8,:), z(8,:), x(9,:),
     z(9,:), x(10,:), z(10,:))

Xsurf = x(1,:);
Zsurf = zeros(1,M);
figure(13)
plot(Xsurf, Zsurf)

Zsurf(1:N) = z(1,1:N);
figure(14)
plot(Xsurf, Zsurf)
for index=delta+1:N
    if z(2,index)>z(1,index), Zsurf(index)=z(1,index);
    elseif z(2,index)<z(1,index), Zsurf(index)=z(2,index);
    end
end
for index=N+1:delta+N
    Zsurf(index)=z(2,index)
end
figure(15)
plot(Xsurf,Zsurf)

for index=2*delta+1:N+delta
    if z(3,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(3,index)<Zsurf(index), Zsurf(index)=z(3,index);
    end
end
for index=N+delta+1:2*delta+N
    Zsurf(index)=z(3,index)
end
figure(16)
plot(Xsurf,Zsurf)

for index=3*delta+1:N+2*delta
    if z(4,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(4,index)<Zsurf(index), Zsurf(index)=z(4,index);
    end
end
for index=N+2*delta+1:3*delta+N
    Zsurf(index)=z(4,index)
end
figure(17)
plot(Xsurf,Zsurf)

for index=4*delta+1:N+3*delta
    if z(5,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(5,index)<Zsurf(index), Zsurf(index)=z(5,index);
    end
end
for index=N+3*delta+1:4*delta+N
    Zsurf(index)=z(5,index)
end
figure(18)
plot(Xsurf,Zsurf)
for index=5*delta+1:N+4*delta
    if z(6,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(6,index)<Zsurf(index), Zsurf(index)=z(6,index);
    end
end

for index=N+4*delta+1:5*delta+N
    Zsurf(index)=z(6,index)
end
figure(19)
plot(Xsurf,Zsurf)

for index=6*delta+1:N+5*delta
    if z(7,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(7,index)<Zsurf(index), Zsurf(index)=z(7,index);
    end
end
for index=N+5*delta+1:6*delta+N
    Zsurf(index)=z(7,index)
end
figure(20)
plot(Xsurf,Zsurf)

for index=7*delta+1:N+6*delta
    if z(8,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(8,index)<Zsurf(index), Zsurf(index)=z(8,index);
    end
end
for index=N+6*delta+1:7*delta+N
    Zsurf(index)=z(8,index)
end
figure(21)
plot(Xsurf,Zsurf)

for index=8*delta+1:N+7*delta
    if z(9,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(9,index)<Zsurf(index), Zsurf(index)=z(9,index);
    end
end
for index=N+7*delta+1:8*delta+N
    Zsurf(index)=z(9,index)
end
figure(22)
plot(Xsurf,Zsurf)
for index=9*delta+1:N+8*delta
    if z(10,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(10,index)<Zsurf(index), Zsurf(index)=z(10,index);
    end
end

for index=N+8*delta+1:9*delta+N
    Zsurf(index)=z(10,index)
end
figure(23)
plot(Xsurf,Zsurf)

point1=floor(N/2);
point2=M-point1;

Zsurf(1:point1)=NaN;
Zsurf(point2:M)=NaN;
figure(24)
plot(Xsurf,Zsurf)

%roughness statistics
PV=max(Zsurf)-min(Zsurf)
PVnm=1000*PV

a=~isnan(Zsurf);
Xstat=Xsurf(a);
Zstat=Zsurf(a);

avg=mean(Zstat)
figure(25)
plot(Xstat,Zstat)
clear index
Ra=0;
RMS=0;
for index=1:length(Zstat)
    Ra=abs(Zstat(index)-avg)+Ra;
    RMS=(Zstat(index)-avg)^2+RMS;
end
Ra=Ra/(length(Zstat))
Ranm=Ra*1000
RMS=(RMS/length(Zstat))^0.5
RMSnm=1000*RMS
%%levelled surfaces

angle=atan(V/U);

Xrot=Xstat*cos(angle)+Zstat*sin(angle);
Zrot=-Xstat*sin(angle)+Zstat*cos(angle);

figure(26)
plot(Xrot,Zrot)

PVtilt=max(Zrot)-min(Zrot)
avgrot=mean(Zrot)
PVtiltnm=1000*PVtilt

clear index
Ratilt=0;
RMStilt=0;
for index=1:length(Zrot)
    Ratilt=abs(Zrot(index)-avgrot)+Ratilt;
    RMStilt=(Zrot(index)-avgrot)^2+RMStilt;
end
Ratilt=Ratilt/(length(Zrot))
Ratilt=1000
RMStilt=(RMStilt/length(Zrot))^0.5
RMSnmtilt=1000*RMStilt
13.2 Appendix B: Machining Procedure for Ultramill

The following initial state is assumed:

- Ultramill and cameras installed on the Nanoform.
- Coolant loop filled with Fluorinert dielectric coolant, and purged of air.
- Ultramill powered down.
- Thermocube turned off.
- Plan view camera offsets ΔX and ΔY determined.
- Touchoff camera in orientation for the material surface.
- Video cameras turned off.

1. Mount the work piece on the spindle vacuum chuck, or on a custom workholder attached to the X-axis tool post.

2. Power up and focus the touch off (side view) and Y-axis (plan view) zoom video cameras.

3. Start logging temperature data.

4. Turn on the Thermocube. Adjust the ball-float flow controller to give an indicated flow of 1.0 L/min (water calibration).

   This corresponds to the optimum actual Fluorinert flow rate of 0.4 L/min.

5. Set the Thermocube control setpoint temperature to 20.0°C. Press the "Start" button, so that the Thermocube begins controlling temperature.

6. Wait 10 minutes after starting Thermocube operation, before powering up the Ultramill.

   This allows coolant temperature transients caused by the initial Thermocube control response to damp out.

7. Power up the Ultramill:

   a. Set DC power supplies to 0 V (turn voltage potentiometer knobs counterclockwise until they reach the limit)
b. Set the signal generator frequency to 1000 Hz and amplitude to 1 V on both channels. Channel A (rake face) should have a non-zero phase angle, usually 90 degrees. Set Channel B (flank face) at zero phase angle.

c. Turn on the DC power supplies simultaneously. The voltmeters should read 0 VDC.

d. Slowly increase the DC voltage offset to 400 VDC by turning the potentiometer knobs clockwise. The DC voltage ramp-up should take at least 15 seconds.

e. Turn on the power amplifiers. Confirm that the peak voltage shown by the voltmeter on each amplifier is 100 times the signal generator amplitude voltage.

f. Step the Ultramill up to operating conditions. This is done by changing the frequency and/or voltage inputs on the signal generator. The maximum step changes for these inputs are

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Maximum Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>±500 Hz/minute</td>
</tr>
<tr>
<td>Voltage (amplitude)</td>
<td>±0.5 V/minute</td>
</tr>
</tbody>
</table>

Change only frequency OR voltage during each step. Change the inputs on both channels as soon after one another as possible. These small step changes prevent local overheating of the piezoelectric stacks.

g. Coolant density and viscosity are sensitive to temperature changes, which can affect the coolant volumetric flow rate. When making changes to the Ultramill power input, adjust the flow controller as required to keep a coolant rate of 0.4 L/min (actual) or 1 L/min (indicated).

8. After the Ultramill has been stepped up to operating conditions, wait for temperature equilibrium to be established in the base block.

If base block and/or end thermocouples have been installed, this can be determined by plotting the time history and noting when the peak-to-valley variation is 0.1 °C with a constant long-term temperature. Otherwise, wait at least 15 minutes after the Ultramill has reached operating voltage and frequency.
9. The Ultramill diamond tool should be cleaned while the piezo stacks are running.

   Use repeated flushing with methanol or isoproyl alcohol and near-zero-force wiping using a lint-free optical wipe. Shop-grade compressed air blow-off should be avoided after cleaning, because oil or water in the air may be deposited on the tool or workpiece.

   **WARNING:** Do not use acetone as this will break down the cement between the tool and toolholder.

10. Align the Y-axis camera crosshairs at the touchoff location, as viewed in the monitor. Then jog the X and Y axes through predetermined offsets $\Delta X$ and $\Delta Y$ to position the tool tip at the touchoff location.

11. Load the automated touchoff assistance program (if used) into the controller.

12. In "MDI" or manual mode, jog the Z axis until the tool is within a few micrometers of the workpiece surface. Set the coordinate system used by the touchoff assistance program to $Z=0$.

13. Execute touchoff, either by jogging the tool into contact with the workpiece, or by using a touchoff assistance program.

14. When touchoff is achieved, set $Z=0$ for the coordinate system used by the motion program.

15. Jog the Ultramill a safe distance back from the workpiece.

16. Load the motion program for the part to be machined into the controller.

17. Apply oil lubricant to the workpiece, using a syringe or swab. Also apply oil above the machining region, to serve as a replenishment reservoir.

18. Execute the motion program, to machine the part.

19. During machining, periodically add replacement coolant to the Thermocube reservoir. An interval of 30 minutes is suggested. Keeping the reservoir full will avoid a system shutdown caused by low coolant level alarm.
20. Monitor the level in the coolant leakage recovery cup. Drain it before it overflows.

Since removing and replacing the cup may create significant vibrations, "idle" periods can be programmed during which the Ultramill pulls away from the workpiece for removing and reinstalling the leakage recovery cup.

21. After the motion program finishes execution, jog the Z-axis to place the Ultramill a safe distance from the workpiece.

22. Power down the Ultramill.

   a. Step down the signal generator's inputs to frequency = 1000 Hz and amplitude = 1.0 V. Observe the following step rates:

<table>
<thead>
<tr>
<th>Frequency</th>
<th>maximum change of -500 Hz / min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage (amplitude)</td>
<td>maximum change of -0.5 V / min</td>
</tr>
</tbody>
</table>

Change only frequency OR voltage amplitude for each step.

   b. Turn off the rake face and flank face power amplifiers.

   c. Slowly reduce the offset DC voltage. This is done by turning the potentiometer knobs on the DC power supplies counterclockwise, until the voltmeters on the DC power supplies both read zero. It should take at least 15 seconds to ramp the DC voltage to zero. Turn the DC power supplies off.

23. Wait 5 minutes before turning off the Thermocube, to cool the piezo stacks.

24. Remove the workpiece for cleaning and metrology.
13.3 Appendix C: Ultramill Elliptical Tool Path Verification

Negish [39] measured the Ultramill's elliptical tool motion at operating frequencies by replacing the diamond tool on the toolholder with a reflecting plastic cube of the same mass. An optical displacement measurement system (the Dual Angstrom Resolver) used the reflection of a source beam to measure the horizontal and vertical position of the toolholder (cube) in the plane of vibration. This technique measured the vibration path to a precision of 10 nm, but required the Ultramill to be taken from service for an extended length of time to install the target cube in lieu of the tool.

An alternate method was investigated which checked the vibration ellipse dimensions by measuring the surface profiles of the exit and entrance regions of a raster-cut groove. In this approach the ellipse can checked without breaking down the machining setup. A test groove was made by slowly plunging the vibrating tool into the workpiece with zero upfeed velocity, to a nominal depth of cut equal to the ellipse semi-minor axis. A 3 mm long groove was cut and the feed motion stopped, after which the tool was extracted vertically. The profiles at the groove ends, where the tool plunged or extracted, were expected to indicate the path followed by the tool cutting edge.

Profiles of the entrance and exit regions along the groove centerline were obtained using white-light interferometry. Figure C-1(a) and C-1(b) show profiles for the entrance and exit of a groove made using a 1 mm nose radius tool mounted on the PEC-type toolholder. Figure C-1(c) is a profile of the region shown in Figure C-1(b), obtained using an Atomic Force Microscope (AFM) instead of an interferometer. The smooth AFM profile shows that the spiky features seen in the interferometry data are artifacts. This test groove was made with Ultramill input conditions of 1000 Hz frequency and with 400 V_{pp} applied voltage. These conditions should produce a toolpath ellipse of 22 μm x 4 μm overall dimension.
Figure C-1. Profiles of test groove. Spike-like features in entrance (a) and exit regions (b) are interferometry artifacts, as shown by (c) smooth profile obtain with an Atomic Force Microscope (AFM).
Figures C-1(a) and (b) show the theoretical tool path superimposed on the profiles of the groove entrance and exit. Also shown is a slightly different "estimated best fit" tool path ellipse of 21.8 µm x 4.4 µm (determined by trial and error), with its major axis rotated 2° above horizontal. These assumed tool paths were visually aligned with the profiles of the cut surfaces. The entrance and exit surface profiles match the assumed tool paths to within 100 nm for most of the horizontal length of the profile. It is therefore concluded that the generated Ultramill tool path is close to the assumed dimensions.

Cutting a test groove and measuring the entrance and exit profiles is thus demonstrated to be a quick method for estimating the toolpath. However, as suggested by fitting two slightly different toolpaths which approximate the cut surface, the accuracy of this method is limited compared with directly measuring the toolholder motion using a reflector cube. In an extended Ultramill machining campaign, the reflector method might be used to initially determine the tool vibration path very precisely. Thereafter test grooves could be cut periodically as a quality-assurance action to verify there had been no gross changes in the tool path.
13.4 APPENDIX D: ULTRAMILL VIDEO MICROSCOPE CAMERAS

13.4.1 Appendix D.1 Camera Description

From Configuration III-A onward, two video-microscope camera systems were installed on the Nanoform-Ultramill system: a side view camera to monitor the tool rake face during touchoff, and a plan view camera on the Y-axis.

Each camera system consists of three principal elements: zoom lens system, CCD camera, and video display monitor.

Magnification $M$ between the feature being imaged and the monitor is given by

$$M = P_{ZOOM} \cdot \frac{S_{MONITOR}}{S_{CCD}}$$  \hspace{1cm} (D-1)

where $P_{ZOOM}$ is the magnification of the lens system, $S_{MONITOR}$ is the diagonal size of the video monitor screen, and $S_{CCD}$ is the diagonal size of the detector chip in the CCD camera.

**Touchoff Camera**

The touchoff camera provides a side view of the rake face of the diamond tool. It was installed by Brocato [6] and used by every system configurations. The zoom lens element has a magnification range of 0.7x to 4.2x. With an 8.5” monitor, the magnification range is 17.8x to 107x. Appendix D.3 describes the use of this camera in several types of touchoff situations.
Y-Axis (Part Plan View) Camera

The Y-axis camera provides a magnified aerial view of the workpiece surface. It includes a crosshair reticle which is used to position the Ultramill's tool tip at desired X and Y coordinates on the work. Appendix D.2 describes use of the Y-axis camera in detail.

The Y-axis camera consists of five components: a color CCD video camera; an optical assembly including the zoom lens and reticle holder/projector; fold mirror; mounting bracket and support strut; and 15" monitor (not shown). The CCD camera has a 1/2" diagonal chip. The zoom lens has an unmodified magnification range of 0.58x to 7.0x. Adaptor tubes of various powers provide additional magnification to the base magnification. Front lenses in various powers can be used to increase or decrease the total magnification of the lens assembly but as magnification increases the working distance to the workpiece surface decreases. Table D-1 shows magnification, resolution, and working distance for various camera component combinations. For the Ultramill, a 0.5x front lens and 2x extension tube were used, for a magnification range of 17.4x–210x at the monitor. This power range gave sufficient working distance to permit installation of the Y-axis camera on the operator side of the Ultramill, without obstructing the touchoff camera's view of the tool rake face. The reticle holder contains a crosshairs reticle, the image of which is focused onto the camera CCD. The fold mirror allows vertical installation of the camera on the Y-axis (see Figure 3-6) in the available space. Figure D-1 shows how the working distance affects the camera layout.

The camera assembly is mounted on a 1" x 1" cross-section carbon steel strut which bolts to the Y-axis mounting plate. The camera mounting is sufficiently stiff that at 210x there is no blurring of the monitor image when the camera body or support strut is tapped lightly.
**Figure D-1** Y-Axis Camera – Critical Dimensions for Layout

**Table D-1. Y-axis Camera Performance**<sup>A</sup> [36]

<table>
<thead>
<tr>
<th>Front Lens</th>
<th>Working distance</th>
<th>Magnification at Monitor&lt;sup&gt;B&lt;/sup&gt;</th>
<th>Resolution</th>
<th>Field of View</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.58 x 7.0 x</td>
<td>0.58 x 7.0 x</td>
<td>0.58 x 7.0 x</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mm μm μm mm Mm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.25 x</td>
<td>341</td>
<td>8.7 x 105 x</td>
<td>33.33 6.67</td>
<td>27.58 2.28</td>
</tr>
<tr>
<td>0.50 x&lt;sup&gt;C&lt;/sup&gt;</td>
<td>165</td>
<td>17.4 x 210 x</td>
<td>18.52 3.33</td>
<td>13.79 1.14</td>
</tr>
<tr>
<td>0.75 x&lt;sup&gt;D&lt;/sup&gt;</td>
<td>108</td>
<td>26.1 x 315 x</td>
<td>11.90 2.22</td>
<td>9.19 0.76</td>
</tr>
<tr>
<td>1.00 x&lt;sup&gt;E&lt;/sup&gt;</td>
<td>86</td>
<td>34.8 x 420 x</td>
<td>9.26 1.67</td>
<td>6.90 0.57</td>
</tr>
<tr>
<td>1.50 x</td>
<td>50</td>
<td>52.2 x 630 x</td>
<td>6.17 1.12</td>
<td>4.60 0.38</td>
</tr>
<tr>
<td>2.00 x</td>
<td>32</td>
<td>69.6 x 840 x</td>
<td>4.50 0.83</td>
<td>3.45 0.29</td>
</tr>
</tbody>
</table>

Camera: Hitachi KP-D20BU color CCD camera, 1/2” format  
Zoom Lens System: Navitar "12-X" system: Series "12X Parfocal Zoom lens", 2x adaptor, reticle holder and 1x projector, thread-bayonet adaptor

<sup>A</sup> with 2x adaptor  
<sup>B</sup> \( S_{\text{MONITOR}} = 15" \), \( S_{\text{CCD}} = 1/2" \)  
<sup>C</sup> Installed during EVAM research  
<sup>D</sup> Can be installed without requiring physical modification of camera bracket & supports  
<sup>E</sup> No lens required for 1.00 x
13.4.2 Appendix D.2 Y-axis Camera Offset

The Y-axis videomicroscope camera provides a magnified aerial view of the workpiece surface as illustrated in Figure D-2. Figure D-3 shows how the Y-axis camera is used to position the tip of the tool at a precise location on the workpiece. The crosshair is aligned at the target location. The X and Y axes are then jogged through pre-determined offset distances $\Delta X$ and $\Delta Y$ which position the tool tip at the required location.

The offset between camera crosshairs and tool tip must be determined each time the Ultramill is installed on the Nanoform. The basic technique is to touch off and plunge-cut a small feature with the Ultramill running. The machine axis coordinates at touchoff are zeroed. Then the axes are jogged to line up the center of the feature with the crosshairs in the Y-axis monitor. The X and Y coordinates are noted from the DTM controller display and used as the offset values $\Delta X$ and $\Delta Y$ from the touchoff point.

![Figure D-2. Screen shot of monitor showing Y-axis camera crosshairs positioned on a feature of interest.](image-url)
In practice, finding the camera offset is more complicated. The features made by the Ultramill are not infinitesimal but have dimensions of at least several micrometers. This means that the location corresponding to the zeroed machine coordinates is not known exactly. As shown in Figure D-4 the horizontal offset $\Delta X$ normally requires finding the X coordinates of the left and right edges of the feature, and then averaging the measurements. The vertical offset $\Delta Y$ is determined in the same manner.

The feature size made by plunge-cutting depends on the tool geometry and depth of cut. At a depth of cut of 1 μm, the width of the feature made by a tool with 1 mm nose radius is 89 μm. A 50 μm nose radius tool makes a feature 20 μm wide. For a sharp-nose tool with 40° included nose angle the feature will be less than 1 μm wide, at depth of cut of 1 μm. Similarly the length of the feature in the upfeed direction is related to the geometry of the vibration ellipse. At 1 μm depth of cut, a 22 μm x 4 μm (overall) elliptical vibration path creates a
Figure D-4. Example of averaging the location of feature edges to estimate camera offsets $\Delta X$ and $\Delta Y$. Feature edges in sketch are shown more clearly than on real features as they appear in camera monitor.

$$\Delta X = 0.5(X_r + X_l)$$
$$\Delta Y = 0.5(Y_r + Y_l)$$

Figure D-5. Sketch showing plunge cut features made by different tools. 22 $\mu$m x 4 $\mu$m vibration ellipse and depth of cut 1 $\mu$m.
feature 18 μm long, while the feature length increases to 22 μm at $d = 2$ μm. Figure D-5 is a sketch illustrating the shapes and relative sizes of touchoff features made with these different tools.

Adequate camera magnification and good lighting are needed to identify the plunge-cut feature, and to discern the feature edges. Camera resolution limits the accuracy that can be obtained, in part because of the uncertainty it introduces in determining the location of the feature edge. Additional uncertainty is introduced by the non-infinitesimal thickness of the reticle crosshair as seen on the monitor. Feature edge locations are usually measured several times and the results averaged, which helps damp out the uncertainty in edge location caused by camera resolution as well as operator bias in placing the crosshair.

To improve visibility of features, they can be made with a short upfeed move instead of a simple plunge cut. When the feature crossfeed width is especially small, as occurs with a sharp-nose or small-radius tool, it may be more realistic to take a single Y-coordinate measurement along the estimated groove centerline, instead of averaging the positions of the top and bottom edges.

For the installed Y-axis camera at maximum magnification of 210x estimated camera offset error was 3 μm, for both X and Y directions. In other words, the uncertainty in tool tip location was a square approximately 6 μm on an edge. Increasing the camera magnification to improve its resolution should allow reduction of this error.
13.4.3 Appendix D.3  Touchoff Methods

"Touching off" is the act of bringing the tool tip safely into contact with the workpiece surface. Touchoff occurs when the operator observes the initial chip formation as the tool first contacts the workpiece surface and begins cutting. The coordinate system is then zeroed in the depth of cut direction. Touchoff thus is a means of finding the position of the workpiece surface, so that programmed depths of cut are within an allowable tolerance of the feature that is actually machined.

Compared to turning, where a continuous chip is produced, touch-off in raster machining with the Ultramill is difficult. The chip is tiny, and is visible for only a brief instant as the tool excavates a feature and removes all uncut material from the vicinity of the elliptical toolpath. The basic procedure is to move the diamond tool tip to within a few micrometers of the workpiece surface while observing it under high magnification using the side view touchoff camera. The running Ultramill is then jogged toward the surface in submicrometer steps until a chip is observed. For workpiece surfaces that do not produce specular reflections or that have high surface roughness, it may not be feasible to clearly discern the location of the tool edge relative to the workpiece surface. In these cases touchoff is confirmed by the appearance of a feature cut into the part surface, as seen through the Y-axis (plan view) camera.

Ultramill touch-off techniques were developed for three situations:

**Touching off into diamond-turned workpieces**

These surfaces have an optical quality surface finish and give specular reflections. This was characteristic of the aluminum and plated copper workpieces used for many of the machining experiments in the present work. Brocato [6] gives a detailed description of touching off in this case. For such situations the touchoff (side-view) camera is mounted so that its line of sight makes an angle of approximately 10° with the workpiece surface. A mirror-image of
Figure D-6. *left* Touchoff camera installation when workpiece has been turned flat to optical quality finish  *right* Monitor view showing tool and reflection in workpiece [6].

the tool appears in the workpiece which makes it easy to discern the location of the tool cutting edge relative to the material surface. Figure D-6 shows the installation of the side-view camera as well as a monitor screen shot of the tool and its reflection in the workpiece.

**Touching off into ground-surface workpieces**

Due to rapid tool wear, ferrous workpieces cannot be faced flat by conventional diamond-turning. An example of this situation is the stainless steel blank used as raw material for the thunderbird feature shown in Figure 1-8. This workpiece was surface ground, which produced a roughness of hundreds of nanometers RMS. The surface produced a diffuse reflection, with an indistinct image of the tool cutting edge.

In such cases touchoff can be achieved by using the side-view camera installed at an oblique angle of approximately 10° to the work surface as shown in Figure D-7. The diffuse reflection of the tool can be used to safely bring the tool to within ~10 μm of the workpiece. From here
touchoff is accomplished by jogging the tool toward the workpiece in sub-micrometer steps, using the Y-axis camera after each jog to see if a surface feature was made. When the tool tip is robust, such as a millimeter-radius round-nosed tool it be possible to touch off by noting creation of the initial chip in the side-view camera. For small radius and sharp-nose tools this procedure is unwise; it is possible to plunge the tool several micrometers into the workpiece without realizing, and the large depth of cut creates tool forces sufficiently large to break the tool. In this situation the only safe way to proceed is to use the Y-axis camera to confirm generation of a touchoff feature.

**Touching off into a "rough" surface**

Some workpieces cannot be diamond-turned or precision ground, and the surface presents no useable reflection of the tool edge. In such a case the sideview camera is installed in the alternate position shown in Figure D-7. In this setup the camera gives a more oblique (approximately 30°) view of the tool tip and the work surface. This allows the operator to estimate the closeness of the tool tip to the workpiece surface because surface features are generally easier to discern in the monitor than when the camera is at the extremely shallow angle used for specular surfaces. However it is difficult to judge the tool-workpiece separation to the sub-micrometer accuracy needed for touchoff. Touchoff therefore should be executed by using the Y-axis camera to confirm creation of a surface feature, as discussed in Appendix D.2.
13.4.4 Appendix D.4  Touchoff Verification Using the Y-Axis Camera

The Y-axis camera can be used to verify that a touchoff feature has been made in the part. The sequence of events is described here:

1. The tool tip is brought to within a few micrometers of the workpiece surface, using the side view touchoff camera.

2. The Z-axis is jogged in toward the surface in a short move (250 nm to 1 μm).

3. An upfeed move of 50 μm or less is made at very slow velocity, to create a linear elongated feature (optional).

Figure D-7. Touchoff camera installed in alternate position providing angled view of surface.
4. The tool is retracted a safe distance from the workpiece. The X and Y axes are then jogged by offset distances $\Delta X$ and $\Delta Y$, to bring the workpiece surface into the field of view of the Y-axis plan view camera.

5. The surface is inspected for a feature made by the touchoff. If a machined feature is visible in the monitor, then touchoff has occurred. The Z-coordinate for touchoff is that which resulted from the incremental jog in Step 2. If no feature is seen, touchoff is assumed to have not occurred. The axes are then jogged back through $\Delta X$ and $\Delta Y$ to position the tool over the workpiece. The Z axis is moved back to the coordinate achieved at the end of Step 2. The cycle is then repeated.

This procedure is readily automated by writing a motion program to be executed by the Nanoform’s controller.

Cutting a short touchoff feature, as suggested in Step 3 of the above procedure, has three advantages compared to simple plunge cutting:

- The feature is more easily discerned in the Y-axis camera's monitor.
- The width of the short linear feature is often easier to measure than with a plunge cut.
  The feature width can be used to calculate the depth of cut, which can be used in determining the actual surface position when knowing this is of critical importance.

Equation D-3 gives the relationship between depth of cut $d$ and feature width $w$ for round-nosed and sharp-nosed tools:
\[ d = R - \frac{\sqrt{16R^2 - w^2}}{4} \]  
\[ d = 2 \cdot w \cdot \cot \left( \frac{\alpha}{2} \right) \]

Equation D-3a is for a round-nosed tool with a nose radius of \( R \). Equation D-3b is used with a sharp-nose tool, where \( \alpha \) is the included angle of the tool rake face. The Z coordinate of the actual surface location is obtained by subtracting \( d \) from the touchoff Z value given on the DTM controller output.
13.5 APPENDIX E: TEMPERATURE MEASUREMENT FOR ULTRAMILL THERMAL EXPERIMENTS

Type "K" (chromal-aluminum) thermocouples were used to measure important Ultramill temperatures. Figure E-1 shows the thermocouple installation locations. The temperatures measured were:

- TC2 - ambient air temperature
- TC3 - coolant supply temperature to the Ultramill
- TC4 - coolant return temperature from the Ultramill
- TC5 - cooling chamber outside skin temperature
- TC6 - piezoelectric stack "indicated skin temperature"
- TC7 - base block outside skin temperature
- TC8 - base block core temperature
- TC10 - base block end temperature

The thermocouple designations correspond to channel assignments of the data acquisition system. TC9 was not used. TC1 was reserved for thermocouple cold-junction compensation.

Details of the thermocouple installation are:

- TC2 was located in ambient air approximately 50 mm above the toolholder, midway between the front face of the Ultramill cooling chamber and the workpiece on the diamond turning machine.
TC3 and TC4 were installed through holes drilled in 1/4" OD plastic plugs. The plugs were inserted into quick-connect tees in the coolant supply and return lines, placing the thermocouple bead into the coolant stream. Edges around the thermocouples were sealed with cyano-acrylate cement.

TC5 and TC7 were taped to the appropriate external surface (cooling chamber or base block) at the Ultramill's longitudinal centerline. The tape used was electrical insulation.

TC6 was taped to the inside surface of one of the piezoelectric stacks, approximately 12 mm from the toolholder end and about 4 mm from the centerline. The tape was an insulating polyimide material. Two layers were used, providing insulation from the circulating coolant. However, it is not known how well the thermocouple was protected from interaction with the coolant. Therefore, TC6 was considered to report "indicated"
stack skin temperature, rather than the internal temperature of the stack. Further, it is a local temperature which is assumed to be only an approximate indicator of the thermal state of the piezo stack, since the temperature distribution in the actuator is 3-dimensional and strongly influenced by variability in the convection heat transfer inside the cooling chamber.

- TC8 was installed in a hole drilled along the block centerline approximately 9 mm behind the interface with the piezo stacks. TC10 was inserted through a hole in the Y-axis mounting plate so that it contacted the rear face of the base block. These thermocouples were held in place by backfilling the holes with furniture glue injected by a hot glue gun.

LabVIEW Version 7 running on a dedicated PC was used to log the thermocouple and measurements. Appendix F shows the LabVIEW block diagram. The thermocouples were terminated at a specialized temperature data acquisition board (LabView PCI-4350) which performed cold-junction compensation, voltage amplification and scaling of the non-linear thermocouple signals to a linear calibration. Temperature data was sampled at 1 Hz, the capability of the temperature DAQ board. To filter data spikes the LabVIEW program reported a 3-point rolling average for each thermocouple.
13.6 Appendix F: LabVIEW Block Diagram for Ultramill Thermal Experiments

Figures F-1, F-2, and F-3 show the block diagram for the LabVIEW VI ("virtual instrument") used to log temperatures and capacitance gauge voltages for the Ultramill. The VI consists of a single block diagram, which is presented here in multiple panels for improved legibility.

Figure F-1 shows the configuration controls for the National Instruments PCI-4350 card used for processing the thermocouple signals. This card performs cold junction compensation (CJC) for the thermocouples, and scaling of non-linear thermocouple voltage signals into temperature units according to the specified thermocouple calibration (e.g. Type "K" as used in this study). This figure also shows the user-specified sample rate.

Figure F-2 shows the WHILE LOOP structure that controls the collection of thermocouple data. The loop utilizes input signals from Figure I-1, including those generated by the PCI-4350 card (dataflows A and B) and the sample interval (dataflow C). The loop commands capture of the temperature value from each channel at times corresponding to the sample interval. Temperature data is stored in an output data file and also displayed in a waveform chart that updates continuously like a strip-chart display.

Figure F-3 shows the WHILE LOOP structure to control the collection of capacitance gauge voltage data. The cap gauge and thermocouple data sampling are controlled by separate loops because temperature data is collected at approximately 1 Hz, while the cap gauge voltage signal is sampled at a frequency of several kHz. The sample rate for the cap gauge is configured by the user along with the number of samples to be exported in each iteration. These samples are averaged internally by the LabView software, to obtain a rolling average of the capacitance gauge voltage. The time and corresponding average voltages are logged in an output data file, and displayed in strip-chart format on the waveform chart.

248
Figure F-1. Labview configuration and control for thermal measurement card.

Figure F-2. Temperature recording and display block.
**Figure F-3.** Data capture and recording block for the capacitance gauge.
13.7 Appendix G: Ultramill Energy Balance

Energy balance tests for the Ultramill established confidence in temperature data obtained with the instrumentation setup described in Chapter 4 and Appendix E. Convective heat loss to the atmosphere from the Ultramill was demonstrated to negligible and can be ignored when evaluating thermal test results.

13.7.1 Appendix G.1 Control Volume Analysis

Figure G-1 shows the control volume for the Ultramill energy balance. The energy balance determines the rate of change of the stored energy $\dot{Q}_{CV}$ within the control volume and can be written

$$\dot{Q}_{CV} = \dot{Q}_{STACK} + \dot{Q}_{F1} - \dot{Q}_{F2} - \dot{Q}_{MP} - \dot{Q}_{CC} - \dot{Q}_{BB}$$

(G-1)

where

- $\dot{Q}_{STACK}$ is the heat produced by the piezoelectric actuator stacks. It is estimated using Equation 4-1 and assumed to be constant
- $\dot{Q}_{F1}$ is the sensible heat content of the coolant supply stream
- $\dot{Q}_{F2}$ is the sensible heat content of the coolant return stream
- $\dot{Q}_{MP}$ is the conduction loss to the Y-axis mounting plate
- $\dot{Q}_{CC}, \dot{Q}_{BB}$ are the convection loss to atmosphere from the cooling chamber and base block, respectively.
The sensible heat of the coolant streams is given by

\[ \dot{Q}_{Fi} = \dot{m} \cdot T_{Fi} \cdot C_P = \rho \cdot \dot{v} \cdot T_{Fi} \cdot C_P \]  \hspace{1cm} (G-2)

where \( i = 1 \) is the supply stream entering the Ultramill and \( i = 2 \) the return stream leaving. \( T_{Fi} \) is the temperature of the fluid stream, \( m \) is the mass flow rate, \( C_P \) is the coolant specific heat, \( \rho \) is its density, and \( v \) is the volumetric flow rate. \( C_P \) and \( \rho \) were obtained from vendor data for the Fluorinert 3823 coolant [65] and assumed to be constant.

Equation G-3 gives \( \Delta \dot{Q}_C \) the heat removal rate by the coolant:

\[ \Delta \dot{Q}_C = \dot{Q}_2 - \dot{Q}_1 = \dot{m} \cdot C_P \cdot (T_{F2} - T_{F1}) = \rho \cdot \dot{v} \cdot C_P \cdot (T_{F2} - T_{F1}) \]  \hspace{1cm} (G-3)
Convection heat transfer from the Ultramill to the surrounding air is given by

\[ \dot{Q}_{CC} = h_{CC} \cdot A_{CC} \cdot (T_{CC} - T_{AMB}) \quad (G-4a) \]
\[ \dot{Q}_{BB} = h_{BB} \cdot A_{BB} \cdot (T_{BB} - T_{AMB}) \quad (G-4b) \]

where the subscripts \( CC \) and \( BB \) denote the cooling chamber and base block respectively. \( \dot{Q}_{CC/BB} \) is the convection heat transfer rate between the component and its surroundings, \( h \) is the convective heat transfer coefficient, \( A \) is the surface area of the component, \( T_{AMB} \) is the ambient air temperature, and \( T \) is the surface temperature of the component. A still-air convection heat transfer coefficient of 10 W/m\(^2\)°C \([23]\) was used for both the base block and cooling chamber. The ambient air temperature measured by the thermocouple TC2 was used for \( T_{AMB} \) and assumed to be uniform for the space surrounding the Ultramill. The surface temperature for the cooling chamber, \( T_{CC} \), and base block, \( T_{BB} \), were assumed to be uniform for each component. The temperatures measured by thermocouples TC5 and TC7 were used for the surface temperatures of the cooling chamber and base block.

Heat transfer can also take place between the back face of the base block, and the mounting plate used to attach the Ultramill to the Y-axis carriage:

\[ \dot{Q}_{MP} = h_{GAP} \cdot A_{END} \cdot (T_{END} - T_{MP}) \quad (G-5) \]

\( \dot{Q}_{MP} \) is the heat transfer between the base block and the mounting plate, \( A_{END} \) is the area of the base block contacting the mounting plate, \( T_{END} \) is the temperature of the end face of the base block, and \( T_{MP} \) is the temperature of the mounting plate. The "gap conductance", \( h_{GAP} \) between two contacting surfaces was was estimated using the method described in \([62]\). A 2mm thick polystyrene barrier was installed between the end of the base block and the Y-axis mounting
plate to eliminate thermal interaction between the Ultramill and Y-axis. This decreased $h_{\text{GAP}}$ from 40,000 W/m²°C to 150 W/m²°C. The small heat transfer potential with the barrier allowed $\dot{Q}_{\text{MP}}$ in Equation G-1 to be treated as zero. The thermal barrier was removed for actual machining because it reduced the stiffness of the Ultramill in the depth of cut direction to 3 GPa, from 14 GPa without the barrier.

### 13.7.2 Appendix G.2 Energy Balance Test Results

The energy balance tests used temperature data obtained while running the Ultramill at constant piezo stack input conditions (1000 Hz frequency and 400 V$_{\text{PP}}$ applied voltage). The tests were conducted with half-round stacks, and the open-circulation cooling configuration.

![Figure G-2. Ultramill temperature data during energy balance test. Half-round stacks operating at 1000 Hz frequency and applied voltage of 400 V$_{\text{PP}}$. Thermocube setpoint = 20.0 °C. Plastic thermal barrier in place.](image-url)
Figure G-3. Ultramill energy balance for temperature data shown in Figure G-2. Half-round stacks operating at 1000 Hz frequency and applied voltage of 400 V\textsubscript{pp}. Thermocube setpoint = 20.0 °C. Plastic thermal barrier in place.

Maintaining a constant coolant flow rate was desirable but the open-circulation cooling required periodic adjustments to keep an adequate fluid level in the sight glass. Coolant flow ranged between 0.30 to 0.45 L / min.

Figure G-2 shows the temperature data collected during the energy balance test. The piezo stacks were powered up at elapsed time of 1000 seconds. The temperature spikes in the ambient air data at 500 and 800 seconds were caused by operator handling of thermocouple TC2. The coolant supply and return temperatures fluctuated initially but were steady state from about 1500 seconds onward. Thereafter the coolant temperature varied over a range of approximately 0.2 °C with an appearance of periodicity, but maintained a constant average temperature. The cooling chamber skin temperature stayed at roughly its initial temperature except for fluctuations of <0.5 °C. The base block temperature rose after the stacks were
powered on at 1000 seconds, and displayed steady-state behavior after 3000 seconds. This
temperature gain is consistent with a net heat flow into the base block from the piezo stacks,
and negates Negishi’s assumption that it can be treated as an isothermal sink.

Figure G-3 shows the net energy storage rate for the system, along with the component heat
transfer rates. These were determined by using the temperatures in Figure G-2 with the
appropriate equations in Section G.1. The stack energy output was zero until the piezos
were energized at 1000 seconds, thereafter it was assumed to be constant at 13.8 W. Equations
G-4(a) and G-4(b) were used to determine the convection loss from the base block and cooling
chamber. The convection heat loss was negligible due to the small temperature difference
between the Ultramill skin and the ambient air, along with the low natural convection heat
transfer coefficient. The heat removed by the coolant was determined using Equation G-3.
The waviness in the figure is because a constant fluid flow rate of 0.4 L/min was assumed but
the coolant flow rate actually needed frequent adjusting to maintain liquid level in the sight
glass. The swath-like appearance of the coolant heat removal is due to spikes in the
temperature data. Since the temperature difference between coolant supply and return streams
is only about 1 °C, these spikes have a large effect on the calculated coolant heat removal. The
slight disparity between stack input and the amount of heat removed by the coolant, evident
after about 2000 seconds, is because the flow rate was likely smaller than the rate of 0.4 L /
min used in the calculations.

The slow rise in the base block skin temperature seen in Figure G-2 indicates that heat is in
fact continuously flowing into the block during the test. However the average rate is only 0.5
W, determined from the temperature rise of approximately 1 °C over 2000 sec and the block’s
estimated thermal capacitance of 830 J/°C. This is less than 5% of the power supplied by the
actuator stack. A non-zero value for the net energy storage rate can also reflect uncertainty in
measurement or system parameters.
The output labeled "Net Energy Storage" in Figure G-3 is the same as $\dot{Q}_{CV}$ in Equation G-1. When it is zero, there is no change in the heat energy stored in the Ultramill control volume. Including the energy storage by the base block determined above, the energy balance is closed to within 10%.

A reasonable conclusion from these results that convection heat transfer losses from the Ultramill are not important to the system's overall thermal behavior and can thus be excluded from consideration. With $\dot{Q}_{CV}$ approximately zero, Equation G-1 reduces to

$$\dot{Q}_{STACK} = \dot{Q}_{F2} - \dot{Q}_{F1} \quad \text{(G-6)}$$

Therefore, the coolant supply and return temperature measurements provide an accurate indication of the heat removed from the stacks by the coolant.
13.8 APPENDIX H: THEORETICAL INVESTIGATION OF THE MINIMUM CHIP THICKNESS EFFECT IN EVAM

At small upfeed velocities, Nanoform axis vibration explains why actual upfeed feature heights and surface roughness are larger than those predicted by EVAM geometry. At larger velocities the EVAM surface features produced are at the predicted spacing \( (F_{up}) \) and look cusp-like as expected but they consistently have a greater height, by a factor of two or more.

The "minimum chip thickness" effect was proposed as an explanation for the actual surface feature heights. In precision turning of ductile materials it is observed that there is a minimum possible chip thickness, established by material hardness and elastic modulus, and by tool sharpness and cutting edge geometry.\([1, 17, 57]\) This minimum chip thickness is estimated to be \( 1/10 \) of the cutting edge radius for turning\([17]\). The phenomena results from elastic compression of the work material as it passes beneath the tool cutting edge and wear land on the flank face. The material then rebounds after the tool passes. The minimum chip thickness produces a corresponding residual feature in the work which protrudes above the theoretical surface expected from the tool profile. This protruding feature has been termed a spanzitfel.

Figure H-1 shows the hypothetical formation of spanzitfels by EVAM in the upfeed direction, and how they might explain the surface feature heights over a range of upfeed velocities. In Figure H-1(a), the tool is beginning to cut upward into the surface left by the preceding pass. By Figure H-1(b) the tool has completed its upward cut, leaving a new surface including a spanzitfel. When the upfeed increment \( F_{up} \) is short relative to the spanzitfel height, on the next downward pass the tool cuts off the top portion as shown in Figure H-1(c). This leaves a feature shorter than the initial spanzitfel, but taller than the theoretical feature based the toolpath geometry, Figure H-1(d). This behavior relies on the (as yet unproven) assumption.
(a) Tool cuts upward through workpiece  

(b) Spanzitfel is formed

(c) Top of spanzitfel is cut.  

(d) New spanzitfel is formed as tool completes upward pass through workpiece

(e) When upfeed is large enough, tool clears the spanzitfel created by the preceding pass.

**Figure H-1.** Spanzitfel hypothesis applied to upfeed feature formation in EVAM
that the spanzitfel material was work-hardened in such a manner that it can now be cut, whereas in the previous tool pass the material in its virgin state deformed elastically and rebounded after tool passage.

Figure H-1(e) shows what happens when $F_{UP}$ is so large that the tool clears the spanzitfel made on the previous pass. In this case, the final surface consists of a line of spanzitfels in the upfeed direction, and the surface roughness is created by their height.

Surface features resulting from the minimum chip thickness effect was modeled for the case of a 22 $\mu$m x 4 $\mu$m vibration ellipse ($A \times B = 11 \mu m \times 2 \mu m$) at a frequency of 1000 Hz. Minimum chip thickness was varied over a range of 25 nm to 400 nm and the resulting feature height simulated for upfeed velocities from $U = 2$ to 8 mm/s ($F_{UP} = 2$ to 8 $\mu$m/cycle).

The surface feature simulation is described in detail in Appendix H.1. The simulated surfaces were generated for several overlapping passes by plotting the toolpath as a function of time using Equations 2-11 and 2-12. The spanzitfel model depicted in Figure H-1 was applied to two successive tool passes, for the range of assumed minimum chip thicknesses.

Figure H-2 compares simulated features formed by the minimum chip thickness effect with those obtained from actual groove surfaces, for $U=8$ mm/s and $U=2$ mm/s. The spanzitfel was assumed to be 250 nm tall in both simulated cases. Results for the minimum chip thickness hypothesis agree qualitatively with the machined surface profiles. For $U = 8$ mm/s the upfeed spacing $F_{UP}$ is sufficiently far apart that the spanzitfel is not cut on a subsequent tool pass (the case depicted in Figure H-1(e)). In this case the simulated surface feature shape possesses the asymmetrical shape of the actual features, including the shallower slope of the trailing side.
At \( U = 2 \text{ mm/s} \) the spanzitfel is modified by subsequent cutting passes of the tool as in Figure H-1(a)-(d). The simulated profile again accurately reflects the general shape of the machined features, which in this case is symmetrical on its leading and trailing slopes.

\[ U = 8 \text{ mm/s} \]

\[ U = 2 \text{ mm/s} \]

**Figure H-2** Comparison of actual surfaces and simulated surfaces based on the minimum chip thickness effect at two upfeed velocities. *(left)* Profiles obtained from actual surfaces. *(right)* Simulated profiles.
Figure H-3 plots the height of simulated surface features generated by the minimum chip thickness hypothesis to actual surface feature heights, for a part machined in plated copper with a sharp tool (Part 62). The peak-to-valley surface feature heights for the actual part were obtained by averaging the heights of three randomly selected features obtained from profiles similar to those shown in Figure 5-7. This was done to make sure the machined feature height excluded the effect of Nanoform vibration in the depth of cut (Z) direction.

![Figure H-3](image-url)

**Figure H-3.** Comparison of simulated spanzitfel roughness and actual results for Part 62 (open-circulation cooling and freshly lapped tool)
For $U = 2$ to 4 mm/s ($F_{up} = 2$ to 4 µm/cycle) the feature height predicted by the simulation was within 10% of the measured height. This is the upfeed velocity range where the spacing between individual features allows a spanzitfel to be cut on subsequent passes of the tool. At $U = 6$ and $U = 8$ mm/s, the spanzitfels are too far apart to be cut on the next tool pass. At $U = 8$ mm/s an assumed 250 nm spanzitfel gave a predicted feature height within 2% of the measured feature, but for $U = 6$ mm/s the simulated feature deviated by more than 20% from the measured feature height. Conversely, a spanzitfel height of 210 nm was needed to make the simulated feature match the actual feature height at $U = 6$ mm/s. But at $U = 8$ mm/s this caused the simulated feature height to be 15% shorter than the measured height. These results mean the minimum chip thickness and spanzitfel cannot be constant with upfeed velocity. This does not invalidate the minimum chip thickness hypothesis for EVAM upfeed feature formation. However the mechanisms for spanzitfel generation would need to cause the feature height to vary as a function of changes in EVAM machining parameters.

13.8.1 Appendix H.1 Simulation of Spanzitfel Features Using MATLAB

The theoretical effect of spanzitfel height on surface roughness was evaluated using a modified version of the EVAM surface simulation program presented in Appendix A.

To simulate the effect of the spanzitfel, three successive tool passes were calculated for each feature. As shown in Figure H-1 the first two passes, displaced by an interval $F_{up}$, create the normal cusp-like EVAM feature. The following logic determines the generated surface:

$\begin{align*}
&IF \ z(\text{pass } 1, x) > z(\text{pass } 2, x), \ THEN \ Z_{\text{surface}}(x) = z(\text{pass } 2, x) \\
&ELSE \ Z_{\text{surface}}(x) = Z(\text{pass } 1, x)
\end{align*}$

where $z(\text{pass } n, x)$ is the tool tip vertical position at arbitrary upfeed position $x$, for the $n$th pass, and $Z_{\text{surface}}(x)$ is the generated machined surface at the same upfeed position $x$. 

263
The spanzitfel feature was then applied to the cusps as shown in Figure H-5. The theoretical surface was predicted by modifying the preceding logic:

\[
IF \ z(pass 1, x) > z(pass 2, x) AND [(z(pass 1, x) - z(pass 2, x)) < spanzitfel height] \quad (II)
THEN Zsurface(x) = z(pass 1, x)
ELSE Zsurface(x) = z(pass 2, x)
\]

The third cutting pass in the sequence was applied. The theoretical generated surface was determined using this logic:

\[
IF \ z(pass 3, x) < Zsurf(x) \quad THEN \quad Zsurf(x) = Z(pass 3, x) \quad (III)
ELSE \quad Zsurf(x) = Zsurf(x)
\]

In this case \(Zsurf(x)\) refers to the previously generated surface made by the first and second passes, and the spanzitfel, described in Case II above.

![Figure H-4 Generation of surface from two successive tool passes](image-url)
Figure H-5. Generation of spanzitfel

Figure H-6. Surface generated when the third tool pass misses the spanzitfel previously generated.
Feature height is reduced by the third tool pass when upfeed increment is small, relative to spanzitfel height.

Figure H-3 shows how there is no interaction with the third cutting pass and the previously created feature, when $F_{UP}$ is large, relative to the spanzitfel height. Figure H-4 shows how the spanzitfel feature is reduced in height by the third tool pass, when $F_{UP}$ is small compared to the spanzitfel.

Appendix H.2 gives the MATLAB code for the spanzitfel surface simulation program. It calculates the theoretical surface, peak-to-valley feature height, and RMS roughness for ten successive features. The program was executed multiple times for a range of upfeed velocities and spanzitfel heights (minimum chip thickness). The vertical feed velocity $V$ was zero, and vibration frequency $f$ was 1000 Hz, since this was used by the Ultramill to machine grooves in the experiments in Chapter 5.
13.8.2 Appendix H.2  Spanzitfel Simulation Program

%% SPANZITFEL.M
%%
%%

dx=0.01;
X=-11:dx:12.;

time=zeros(1,length(X));

U=8000;
A=11;
B=2;
V=0;
f=1000;
omega=2*pi*f;
phi=-pi;
spanzitfel=0.250;

X(1)=-11;
time(1)=0;

for index=2:length(X)
    clear xcheck
    time(index)=time(index-1);
    xcheck=A*cos(time(index)*omega+phi)+U*time(index);
    tol=X(index)-xcheck;

    while abs(tol)>.0001;
        dt=(X(index)-xcheck)/(U+omega*A);
        time(index)=time(index)+dt;
        xcheck=A*cos(time(index)*omega+phi)+U*time(index);
        tol=X(index)-xcheck;
    end
    index
    time(index)

end

Z=B*sin(time*omega+phi)+V*time;
figure(3)
plot(X,Z)

N=length(time);

passes=10;
delta=(U/f)/dx;
M=(passes-1)*delta+N;
clear index

X=X*1/dx;
X=round(X);
X=dx*X;

x=zeros(passes,M);
z=zeros(passes,M);

for index=1:N
    x(1,index)=X(index);
    z(1,index)=Z(index);
end
for index=N+1:M
    x(1,index)=x(1,index-1)+dx;
    z(1,index)=NaN;
end
figure(6)
plot(x(1,:),z)

length(x(1,index))
length(M)

zcheck=Z-(z(1,1:N))

x(2,:)=x(1,:);
for index=1:delta
    z(2,index)=NaN;
end
for index=N+delta+1:M
    z(2,index)=NaN;
end
clear index

for index=delta+1:N+delta
    z(2,index)=z(1,index-delta)+V/f;
end
figure(8)
plot(x(1,:),z(1,:),x(2,:),z(2,:))

x(3,:) = x(1,:)
for index = 1:2*delta
  z(3,index) = NaN;
end
for index = N+1+2*delta:M
  z(3,index) = NaN;
end
for index = 2*delta+1:N+2*delta
  z(3,index) = z(1,index-2*delta)+2*V/f;
end

x(4,:) = x(1,:)
clear index
for index = 1:3*delta
  z(4,index) = NaN;
end
for index = N+1+3*delta:M
  z(4,index) = NaN;
end
for index = 3*delta+1:N+3*delta
  z(4,index) = z(1,index-3*delta)+3*V/f;
end

x(5,:) = x(1,:)
clear index
for index = 1:4*delta
  z(5,index) = NaN;
end
for index = N+1+4*delta:M
  z(5,index) = NaN;
end
for index = 4*delta+1:N+4*delta
  z(5,index) = z(1,index-4*delta)+4*V/f;
end

x(6,:) = x(1,:)
clear index
for index = 1:5*delta
  z(6,index) = NaN;
end
for index = N+1+5*delta:M
  z(6,index) = NaN;
end
for index = 5*delta+1:N+5*delta
  z(6,index) = z(1,index-5*delta)+5*V/f;
end

x(7,:) = x(1,:)
clear index
for index = 1:6*delta
  z(7,index) = NaN;
end
end
for index=N+1+6*delta:M
    z(7,index)=NaN;
end
for index=6*delta+1:N+6*delta
    z(7,index)=z(1,index-6*delta)+6*V/f;
end
figure(11)
plot(x(1,:),z(1,:),x(2,:),z(2,:),x(3,:),z(3,:),x(4,:),z(4,:),x(5,:),z(5,:),x(6,:),z(6,:),x(7,:),z(7,:))
x(8,:)=x(1,:);
clear index
for index=1:7*delta
    z(8,index)=NaN;
end
for index=N+1+7*delta:M
    z(8,index)=z(1,index-7*delta)+7*V/f;
end
x(9,:)=x(1,:);
clear index
for index=1:8*delta
    z(9,index)=NaN;
end
for index=N+1+8*delta:M
    z(9,index)=z(1,index-8*delta)+8*V/f;
end
x(10,:)=x(1,:);
clear index
for index=1:9*delta
    z(10,index)=NaN;
end
for index=N+1+9*delta:M
    z(10,index)=z(1,index-9*delta)+9*V/f;
end
figure(12)
plot(x(1,:),z(1,:),x(2,:),z(2,:),x(3,:),z(3,:),x(4,:),z(4,:),x(5,:),z(5,:),x(6,:),z(6,:),x(7,:),z(7,:),x(8,:),z(8,:),x(9,:),z(9,:),x(10,:),z(10,:))
Xsurf=x(1,:);
Zsurf=zeros(1,M);

figure(13)
plot(Xsurf,Zsurf)
Zsurf(1:N)=z(1,1:N);
figure(14)
plot(Xsurf,Zsurf)
for index=delta+1:N
    if z(2,index)>z(1,index), Zsurf(index)=z(1,index);
    elseif z(2,index)<z(1,index) && (z(1,index)-z(2,index))<spanzitfel,
        Zsurf(index)=z(1,index);
    elseif z(2,index)<z(1,index), Zsurf(index)=z(2,index);
    end
end
for index=N+1:delta+N
    Zsurf(index)=z(2,index)
end

figure(15)
plot(Xsurf,Zsurf)
for index=2*delta+1:N+delta
    if z(3,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(3,index)<z(2,index) && (z(2,index)-z(3,index))<spanzitfel,
        Zsurf(index)=z(2,index);
    elseif z(3,index)<Zsurf(index), Zsurf(index)=z(3,index);
    end
end
for index=N+delta+1:2*delta+N
    Zsurf(index)=z(3,index)
end
figure(16)
plot(Xsurf,Zsurf)
for index=3*delta+1:N+2*delta
    if z(4,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(4,index)<z(3,index) && (z(3,index)-z(4,index))<spanzitfel,
        Zsurf(index)=z(3,index);
    elseif z(4,index)<Zsurf(index), Zsurf(index)=z(4,index);
    end
end
for index=N+2*delta+1:3*delta+N
    Zsurf(index)=z(4,index)
end
End
figure(17)
plot(Xsurf,Zsurf)

for index=4*delta+1:N+3*delta
    if z(5,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
        elseif z(5,index)<z(4,index) && (z(4,index)-z(5,index))<spanzitfel,
            Zsurf(index)=z(4,index);
        elseif z(5,index)<Zsurf(index), Zsurf(index)=z(5,index);
    end
end
for index=N+3*delta+1:4*delta+N
    Zsurf(index)=z(5,index)
end
figure(18)
plot(Xsurf,Zsurf)

for index=5*delta+1:N+4*delta
    if z(6,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
        elseif z(6,index)<z(5,index) && (z(5,index)-z(6,index))<spanzitfel,
            Zsurf(index)=z(5,index);
        elseif z(6,index)<Zsurf(index), Zsurf(index)=z(6,index);
    end
end
for index=N+4*delta+1:5*delta+N
    Zsurf(index)=z(6,index)
end
figure(19)
plot(Xsurf,Zsurf)

for index=6*delta+1:N+5*delta
    if z(7,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
        elseif z(7,index)<z(6,index) && (z(6,index)-z(7,index))<spanzitfel,
            Zsurf(index)=z(6,index);
        elseif z(7,index)<Zsurf(index), Zsurf(index)=z(7,index);
    end
end
for index=N+5*delta+1:6*delta+N
    Zsurf(index)=z(7,index)
end
figure(20)
plot(Xsurf,Zsurf)

for index=7*delta+1:N+6*delta
    if z(8,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
        elseif z(8,index)<z(7,index) && (z(7,index)-z(8,index))<spanzitfel,
            Zsurf(index)=z(7,index);
        elseif z(8,index)<Zsurf(index), Zsurf(index)=z(8,index);
    end
end
for index=N+6*delta+1:7*delta+N
    Zsurf(index)=z(8,index)
end
figure(21)
plot(Xsurf,Zsurf)

for index=8*delta+1:N+7*delta
    if z(9,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(9,index)<z(8,index) && (z(8,index)-z(9,index))<spanzitfel,
        Zsurf(index)=z(8,index);
    elseif z(9,index)<Zsurf(index), Zsurf(index)=z(9,index);
    end
end
for index=N+7*delta+1:8*delta+N
    Zsurf(index)=z(9,index)
end
figure(22)
plot(Xsurf,Zsurf)

for index=9*delta+1:N+8*delta
    if z(10,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(10,index)<z(9,index) && (z(9,index)-z(10,index))<spanzitfel,
        Zsurf(index)=z(9,index);
    elseif z(10,index)<Zsurf(index), Zsurf(index)=z(10,index);
    end
end
for index=N+8*delta+1:9*delta+N
    Zsurf(index)=z(10,index)
end
figure(23)
plot(Xsurf,Zsurf)

T1=find(z(1,:)==min(z(1,:)))
T2=find(z(2,:)==min(z(2,:)))
T3=find(z(3,:)==min(z(3,:)))
T4=find(z(4,:)==min(z(4,:)))
T5=find(z(5,:)==min(z(5,:)))
T6=find(z(6,:)==min(z(6,:)))
T7=find(z(7,:)==min(z(7,:)))
T8=find(z(8,:)==min(z(8,:)))
T9=find(z(9,:)==min(z(9,:)))
T10=find(z(10,:)==min(z(10,:)))

for index=T1:T2
    if z(3,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(3,index)<Zsurf(index), Zsurf(index)=z(3,index);
    end
end
figure(28)
plot(Xsurf,Zsurf)
for index=T2:T3
    if z(4,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(4,index)<Zsurf(index), Zsurf(index)=z(4,index);
end
figure(29)
plot(Xsurf,Zsurf)

for index=T3:T4
    if z(5,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(5,index)<Zsurf(index), Zsurf(index)=z(5,index);
end
figure(30)
plot(Xsurf,Zsurf)

for index=T4:T5
    if z(6,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(6,index)<Zsurf(index), Zsurf(index)=z(6,index);
end
figure(31)
plot(Xsurf,Zsurf)

for index=T5:T6
    if z(7,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(7,index)<Zsurf(index), Zsurf(index)=z(7,index);
end
figure(32)
plot(Xsurf,Zsurf)

for index=T6:T7
    if z(8,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(8,index)<Zsurf(index), Zsurf(index)=z(8,index);
end
figure(33)
plot(Xsurf,Zsurf)

for index=T7:T8
    if z(9,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(9,index)<Zsurf(index), Zsurf(index)=z(9,index);
end
figure(34)
plot(Xsurf,Zsurf)
for index=T8:T9
    if z(10,index)>Zsurf(index), Zsurf(index)=Zsurf(index);
    elseif z(10,index)<Zsurf(index), Zsurf(index)=z(10,index);
end
end
figure(35)
plot(Xsurf,Zsurf)

point1=floor(N/2);
point2=M-point1;

Zsurf(1:point1)=NaN;
Zsurf(point2:M)=NaN;

figure(24)
plot(Xsurf,Zsurf)

%roughness statistics
PV=max(Zsurf)-min(Zsurf)
PVnm=1000*PV

a=~isnan(Zsurf);
Xstat=Xsurf(a);
Zstat=Zsurf(a);

avg=mean(Zstat)

figure(25)
plot(Xstat,Zstat)

clear index
Ra=0;
RMS=0;
for index=1:length(Zstat)
    Ra=abs(Zstat(index)-avg)+Ra;
    RMS=(Zstat(index)-avg)^2+RMS;
end
Ra=Ra/(length(Zstat))
Ranm=Ra*1000
RMS=(RMS/length(Zstat))^0.5
RMSnm=1000*RMS
13.9 APPENDIX I: PROGRAM LISTING FOR UPFEED KNIFE-EDGE PART

Program listing (G-code) for making the “upfeed knife edge part” described in Chapter 7.

Main program P1071.nc

%  
N02 P1071  
N05 G54 G90  
N07 G00 X-10 Y0.0  
N10 G90 G1 X0.657 Y0.0 Z0.5 F10  
N15 G90 G09 G01 Z0.010 F1  
N16 G90 G01 Z0.000 F0.1  
N17 M98 L100 P8121  
N19 M98 L200 p8122  
N24 M98 L100 P8121  
N52 G90 G01 Z2.0 F6  
N99 M30  
%

Subroutine O8121 – flat field

(o8121)  
G54  
G90 G09 G01 X0.600 Z-0.005 F1  
G09 G01 X0.057 F30.0  
G09 G01 X0.000 Z0.000 F1  
G01 Z0.005 F1  
G91 G09 G01 Y0.020 F1  
M99
Subroutine O82122 – knife edge region

(o8122)
G54
G90 G09 G01 X0.600 Z-0.005 F1
G09 G01 X0.357 F30.0
G09 G01 X0.354 Z0.001 F1
G09 G01 X0.360 F1
G09 G01 X0.326 Z0.005 F1
G09 G01 X0.057 F30
G01 G01 X0.000 Z0.000 F1
G01 Z0.005 F1
G91 G09 G01 Y0.020 F1
M99
13.10 APPENDIX J: PROGRAM LISTING FOR FLYING WING PART

Listed below is the MATLAB code for the morphology based program used to generate G-code for the “flying wing” part described in Chapter 7. The G-code is over 20,000 lines and is therefore not shown.

%% MORPHOLOGY PROGRAM TO GENERATE G-CODE PROGRAM FOR THE FLYING WING
%% FEATURE

%initialize X,Y,Z matrices for surface description
Xsurf=0:1:1400;
Ysurf=0:1:1600;
Zsurf=0*ones(size(Xsurf));
[r,c]=size(Zsurf);
%create the surface for the rear set of planes
%left rear plane
for index=501:781
    for count=301:801
        Zsurf(index,count)=-52.237+.121*Xsurf(index)-.0279*Ysurf(count);
        if Zsurf(index,count)<0,
            Zsurf(index,count)=0;
        end
        if Xsurf(index)>0.56*Ysurf(count)+332,
            Zsurf(index,count)=0;
        end
        if Xsurf(index)<0.23*Ysurf(count)+431,
            Zsurf(index,count)=0;
        end
    end
end

%right rear plane
for index=501:781
    for count=801:1301
        Zsurf(index,count)=-96.85+.121*Xsurf(index)+.0279*Ysurf(count);
    end
end
if Zsurf(index, count)<0,
   Zsurf(index, count)=0;
end
if Xsurf(index)>-0.56*Ysurf(count)+1228,
   Zsurf(index, count)=0;
end
if Xsurf(index)<-0.23*Ysurf(count)+799,
   Zsurf(index, count)=0;
end
end
end

max(max(Zsurf))

Z1=Zsurf;

% create front left plane
for index=501:901
   for count=301:801
      Zsurf(index, count)=
         -0.167*Xsurf(index)+.1333*Ysurf(count)+43.33;
      if Zsurf(index, count)<0,
         Zsurf(index, count)=0;
      end
      if Xsurf(index)<=0.56*Ysurf(count)+332,
         Zsurf(index, count)=Z1(index, count);
      end
      if Xsurf(index)>0.8*Ysurf(count)+260,
         Zsurf(index, count)=0;
      end
   end
end
%create front right plane
for index=501:901
  for count=801:1301
    Zsurf(index,count)=-.167*Xsurf(index)-.1333*Ysurf(count)+256.67;
    if Zsurf(index,count)<0,
      Zsurf(index,count)=0;
    end
    if Xsurf(index)<=-0.56*Ysurf(count)+1228,
      Zsurf(index,count)=Z1(index,count);
    end
    if Xsurf(index)>0.8*Ysurf(count)+260,
      Zsurf(index,count)=0;
    end
  end
end
clear Z1
figure(3)
mesh(Zsurf)
max(max(Zsurf))
for index=1:57
  for count=1:65
    ZS1(index,count)=Zsurf(25*(index-1)+1,25*(count-1)+1);
  end
end
figure(114)
mesh(ZS1)
Zsurf=10*Zsurf;
figure(4)
mesh(Zsurf)
%develop "structural element" (probe geometry) for use in dilation
xtool=-11:1:11;
ytool=-243:1:243;
for index=1:length(xtool)
  for count=1:length(ytool)
    ztool(index,count)=(1000000-ytool(count)^2)^0.5+(4-(4/121)*xtool(index)^2)^0.5-1000;
  end
end
ztoolmin=min(min(ztool))
ztool=ztool-ztoolmin;
figure(19)
mesh(ztool)
ztool=10*ztool;  %multiply so that it is made of 100 nm high pixels

%define the structural element

for index=1:length(xtool)
    for count=1:length(ytool)
        nhood(index,count)=1;
    end
end
probe=strel(nhood,ztool);

image=imdilate(Zsurf,probe);
%figure(21)
%mesh(image)

offset=image-Zsurf;
%figure(22)
%mesh(offset)

figure(23)
plot(Xsurf,Zsurf(:,800),Xsurf,image(:,800),Xsurf,offset(:,800))

figure(24)
plot(Xsurf, Zsurf(:,600),Xsurf,image(:,600),Xsurf,offset(:,600))
figure(25)
plot(Xsurf, Zsurf(:,1200),Xsurf,image(:,1200),Xsurf,offset(:,1200))

for index=1:(length(Xsurf))
    for count=1:(length(Ysurf))
        if Xsurf(index)<251, Zsurf(index, count)=NaN;
        end
        if Xsurf(index)>1150,Zsurf(index,count)=NaN;
        end
    end
end

figure(1)
surfc(Zsurf)
hidden on
shading interp
colormap(copper)
clear probe; clear nhood; clear xtool; clear ytool; clear ztool
clear Zsurf;

imagemax=max(max(image));
image=image-imagemax;
image=image/10-10;

figure(27)
plot(Ysurf,image(780,:))
figure(28)
plot(Xsurf,image(:,800))
figure(29)
plot(Ysurf,image(615,:))

imagecheck=image;

%extract toolpath data
lowest=min(min(image))
%find vector of maximum toolpath points
for count=1:1601
    vector=image(:,count);
    [a,b]=max(vector);
    x2(count)=b;
    x1(count)=NaN;
    x3(count)=NaN;
    matrix=(vector>lowest);
    for index=2:length(matrix)
        if (matrix(index)==1 && matrix(index-1)==0),
            x1(count)=index;
        end
        if (matrix(index)==0 && matrix(index-1)==1),
            x3(count)=index;
        end
    end
end
length(x1)
length(x2)
length(x3)
ymin=1:1:1601;
figure(98)
plot(ymin,x1,ymin,x3,ymin,x2)
Xtool=zeros(17,length(Ysurf));
Ztool=zeros(17,length(Ysurf));

for count=1:length(Ysurf)
    if ~isnan(x1(count))
        dxup=(x2(count)-x1(count))/8;
        dxdown=(x3(count)-x2(count))/8;
        Xtool(1,count)=Xsurf(x1(count));
        Ztool(1,count)=image(x1(count),count);
        Xtool(2,count)=Xsurf(x1(count)+round(dxup));
        Ztool(2,count)=image(x1(count)+round(dxup),count);
        Xtool(3,count)=Xsurf(x1(count)+round(2*dxup));
        Ztool(3,count)=image(x1(count)+round(2*dxup),count);
        Xtool(4,count)=Xsurf(x1(count)+round(3*dxup));
        Ztool(4,count)=image(x1(count)+round(3*dxup),count);
        Xtool(5,count)=Xsurf(x1(count)+round(4*dxup));
        Ztool(5,count)=image(x1(count)+round(4*dxup),count);
        Xtool(6,count)=Xsurf(x1(count)+round(5*dxup));
        Ztool(6,count)=image(x1(count)+round(5*dxup),count);
        Xtool(7,count)=Xsurf(x1(count)+round(6*dxup));
        Ztool(7,count)=image(x1(count)+round(6*dxup),count);
        Xtool(8,count)=Xsurf(x1(count)+round(7*dxup));
        Ztool(8,count)=image(x1(count)+round(7*dxup),count);
        Xtool(9,count)=Xsurf(x2(count));
        Ztool(9,count)=image(x2(count),count);
        Xtool(10,count)=Xsurf(x2(count)+round(dxdown));
        Ztool(10,count)=image(x2(count)+round(dxdown),count);
        Xtool(11,count)=Xsurf(x2(count)+round(2*dxdown));
        Ztool(11,count)=image(x2(count)+round(2*dxdown),count);
        Xtool(12,count)=Xsurf(x2(count)+round(3*dxdown));
        Ztool(12,count)=image(x2(count)+round(3*dxdown),count);
        Xtool(13,count)=Xsurf(x2(count)+round(4*dxdown));
        Ztool(13,count)=image(x2(count)+round(4*dxdown),count);
        Xtool(14,count)=Xsurf(x2(count)+round(5*dxdown));
        Ztool(14,count)=image(x2(count)+round(5*dxdown),count);
        Xtool(15,count)=Xsurf(x2(count)+round(6*dxdown));
        Ztool(15,count)=image(x2(count)+round(6*dxdown),count);
        Xtool(16,count)=Xsurf(x2(count)+round(7*dxdown));
        Ztool(16,count)=image(x2(count)+round(7*dxdown),count);
        Xtool(17,count)=Xsurf(x3(count));
        Ztool(17,count)=image(x3(count),count);
    else
        Xtool(:,count)=NaN;
        Ztool(:,count)=lowest;
    end
end
figure(75)
mesh(Xtool)
figure(76)
mesh(Ztool)

Ytool=Ysurf/1000;
Xtool=Xtool/1000;
Ztool=Ztool/1000-.002;

clear x1; clear x2; clear x3;

feed=30*ones(1,12)
toolpath=Ztool+.020;
upfeed=zeros(10,201);

%load the extracted tool center position data into the arrays used for
%actual toolpath programming
for index=1:17
    for count=1:161
        upfeed(index,count)=Xtool(18-index,1+10*(count-1));
toolpath(index,count)=toolpath(18-index,1+10*(count-1));
    end
end
%crossfeed vector
crossfeed=0:.010:1.600;

fid=fopen('output.txt','wt')
fprintf(fid,'(08113)
');
fprintf(fid,'G55
');
fprintf(fid,'G90

');
for count=1:length(crossfeed)
    fprintf(fid,'G01 G09 X1.452 Z-.012 F20
');
    if ~isnan(Xtool(2,1+10*(count-1))),
        for index=1:17
            fprintf(fid,'G01 X%5.3f Z%9.5f F30 \n',
upfeed(index,count),toolpath(index,count));
        end
    end
    fprintf(fid,'G01 G09 X0 Z-.012 F30 \n');
    fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
    fprintf(fid,'G00 X1.6 \n');
    fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(count));
else
    fprintf(fid,'G01 G09 X0.000 Z-.012 F30 \n');
    fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
    fprintf(fid,'G00 X1.6 \n');
    fprintf(fid,'G01 Y%5.3f F5 \n',crossfeed(count));
end

end

toolpath=toolpath-.010;

for count=1:length(crossfeed)
    fprintf(fid,'G01 G09 X1.452 Z-.022 F20 \n');
    if ~isnan(Xtool(2,1+10*(length(crossfeed)-count))),
        for index=1:17
            fprintf(fid,'G01 X%5.3f Z%9.5f F30 \n',
            upfeed(index,count),toolpath(index,count));
        end
        fprintf(fid,'G01 G09 X0 Z-.022 F30 \n');
        fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
        fprintf(fid,'G00 X1.7 \n');
        fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(length(crossfeed)+1-count));
    else
        fprintf(fid,'G01 G09 X0.000 Z-.022 F30 \n');
        fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
        fprintf(fid,'G00 X1.7 \n');
        fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(length(crossfeed)+1-count));
    end
end

toolpath=toolpath-.009;

for count=1:length(crossfeed)
    fprintf(fid,'G01 G09 X1.452 Z-.031 F20 \n');
    if ~isnan(Xtool(2,1+10*(count-1))),
        for index=1:17
            fprintf(fid,'G01 X%5.3f Z%9.5f F30 \n',
            upfeed(index,count),toolpath(index,count));
        end
    end

fprintf(fid,'G01 G09 X0 Z-.031 F30 \n');
fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
fprintf(fid,'G00 X1.7 \n');
fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(count));

else
    fprintf(fid,'G01 G09 X0.000 Z-.031 F30 \n');
    fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
    fprintf(fid,'G00 X1.7 \n');
    fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(count));
end

end

toolpath=toolpath-.001;

for count=1:length(crossfeed)
    fprintf(fid,'G01 G09 X1.452 Z-.032 F20 \n');

    if ~isnan(Xtool(2,1+10*(length(crossfeed)-count))),
        for index=1:17
            fprintf(fid,'G01 X%5.3f Z%9.5f F30 \n',
                    upfeed(index,count),toolpath(index,count));
        end
        fprintf(fid,'G01 G09 X0 Z-.032 F30 \n');
        fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
        fprintf(fid,'G00 X1.7 \n');
        fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(length(crossfeed)+1-count));
    else
        fprintf(fid,'G01 G09 X0.000 Z-.032 F30 \n');
        fprintf(fid,'G01 G09 X-.160 Z.005 F30 \n');
        fprintf(fid,'G00 X1.7 \n');
        fprintf(fid,'G01 Y%5.3f F5 \n\n',crossfeed(length(crossfeed)+1-count));
    end
end

fprintf(fid,'M99 \n');

fclose(fid)
13.11 APPENDIX K: PROGRAM LISTING FOR “POTATO CHIP PART” (DOUBLE-SINUSOIDAL GROOVES)

Listed below is the MATLAB code for the morphology based program used to generate G-code for the “double sinusoid” part described in Chapter 7. The G-code itself is over 32,000 lines and is therefore not shown.

```
% define the sinusoidal surface
X=0:1:1280; % coordinates in terms of 1 micron pixels
Y=0:1:320; % coordinates in terms of 1 micron pixels

lambda1=1280; % wavelengths of sinusoidal functions in pixels
lambda2=320;  % wavelengths of sinusoidal functions in pixels

% find surface Z(X,Y) each gradation is 100 nm
for index=1:length(Y)
    F(index)=32.5+12.5*sin(2*pi*Y(index)/lambda2-pi/2);
end
for count=1:length(X)
    Z(count,index)=F(index)*sin(2*pi*X(count)/lambda1-pi/2)-F(index);
end

Zact=upfeed_1.z;
Xact=upfeed_1.d;
zmin=min(Z) % minimum value of surface
zmin=min(zmin)%
Zsurf=Z-zmin; % creates surface with min value of 0;

figure(1)
plot(Xact,Zact,X,Zsurf)

Z=1000*z;
Z=Z-min(min(Z));

cl=find(Y==0);
Zmax=max(Z(:,cl));
zmax=max(z(:,cl));
```

287
for index=1:length(X)
    for count=1:length(Y)
        if Z(index,count)>Zmax,Z(index,count)=Zmax;
        end
        if z(index,count)>zmax,z(index,count)=zmax;
        end
    end
end

figure(2)
mesh(Z')

%% check the surface along several sections
xsect1=X;
cl=find(cl==80);
zsect1=Z(:,cl);
xsect2=X;
s2=find(s==y(160));
zsect2=Z(:,s2)

figure(3)
plot(xsect1,zsect1,xsect2,zsect2)
xmid=0.5*(xsect1(1)+xsect1(length(xsect1)));
X=X-min(X)+2;

%DEVELOP THE STRUCTURAL ELEMENT (probe) FOR USE IN DILATION
% USE RADIUS = 1000 micrometers   A=12 and B = 2 micrometers
xtool=-12:1:12;
ytool=-350:1:350;

for index=1:length(xtool)
    for count=1:length(ytool)
        ztool(index,count)=(1000^2-ytool(count)^2)^0.5+(4-(4/144)*xtool(index)^2).^5-1000;
    end
end
ztoolmin=min(min(ztool))
ztool=ztool-ztoolmin;
%figure(5)
%mesh(ztool)
%ztool=10*ztool;  %multiply so that it is made of 100 nm high pixels
%define the structural element

for index=1:length(xtool)
    for count=1:length(ytool)
        nhood(index,count)=1;
    end
end
probe=strel(nhood,ztool);

image=imdilate(Z,probe);

figure(6)
mesh(image)

diff=image-Z;
figure(7)
mesh(diff')

clear diff
pack

figure(8)
plot(X,Z(:,cl),X,image(:,cl))
aa=find(Y==100);
figure(9)
plot(X,Z(:,aa),X,image(:,aa))

Zsurf=imerode(image,probe);
figure(10)
mesh(Zsurf')

Zsurf=Zsurf-min(min(Zsurf));
checksurf=Z-Zsurf;
figure(11)
mesh(checksurf');
clear Zsurf
clear checksurf
pack

%create the toolpath image from the full image

toolpath=image-min(min(image));
toolpath=toolpath-max(max(toolpath));
toolpath=toolpath(:,1:10:end); %toolpath consists only of crossfeed spacing every 10 um
figure(12)
mesh(toolpath')
axis square

% check profiles from the toolpath

[rows, columns]=size(toolpath)
cl=floor(1+columns/2)
bb=floor(columns/4)

figure(13)
plot(X,toolpath(:,1),X,toolpath(:,bb), X, toolpath(:,cl))

clear count
clear index
pack

toolpath(find(abs(toolpath)<=0.001))=0;

% set up divisions and interval lengths in the upfeed passes

for count=1:columns
    clear Lpass
    Lpass=find(toolpath(:,count)<0);
    segment(count,1)=Lpass(1)-1;
    segment(count,2)=Lpass(length(Lpass))+1;
    distance(count)=segment(count,2)-segment(count,1);
    div(count)=floor(distance(count)/10)+1;
    if div(count)<2, div(count)=2;
    end
    if div(count)>33, div(count)=33;
    end
    interval(count)=distance(count)/div(count);  % length of each
    upfeed cutting segment in um
end

n=1:1:columns;
figure(14)
stem(n,distance)
figure(15)
stem(n,div)
figure(16)
stem(n,interval)
figure(17)
plot(n,segment(:,1),n,segment(:,2))
%% set up matrix of actual tool points
%% 10 um crossfeed spacing and 33 upfeed points in a pass

%establish number of divisions and keypoints per pass

nkeypoints=2*ones(1,columns)
for pass=1:columns
    nkeypoints(pass)=1+div(pass)
end

Xpath=NaN*ones(max(div)+1,columns);
Zpath=Xpath;
Ypath=0:10:length(Y); %crossfeed positions
Xpath_shear=Xpath;
shear=-.004; %correction factor for X-Z squareness error

for pass=1:columns
    maxm=nkeypoints(pass); %number of keypoints in this pass = 3 to 33 (2-32 linear segments)
    if segment(pass,1)<1,Xpath(1,pass)=X(1);
    else Xpath(1,pass)=X(segment(pass,1));
    end
    if segment(pass,1)<1,Zpath(1,pass)=toolpath(1,pass);
    else Zpath(1,pass)=toolpath(segment(pass,1),pass)%first keypoint for the pass, Z should be 0
    end
    Xpath_shear(1,pass)=Xpath(1,pass)+Ypath(pass)*shear;
    if segment(pass,2)>length(X), Xpath(maxm,pass)=X(length(X));
    else Xpath(maxm,pass)=X(segment(pass,2)); %last keypoint for the pass
    end
    if segment(pass,2)>length(X),Zpath(maxm,pass)=toolpath((length(X)),pass);
    else Zpath(maxm,pass)=toolpath(segment(pass,2),pass);
    end
    Xpath_shear(maxm,pass)=Xpath(maxm,pass)+shear*Ypath(pass);

    for keypoint=2:maxm-1
        Xpath(keypoint,pass)=Xpath(1,pass)+(keypoint-1)*interval(pass);
    end

291
Xpath_shear(keypoint, pass) = Xpath(keypoint, pass) + \text{shear} \times Ypath(pass);
Xlo = \text{floor}(Xpath(keypoint, pass));
Xhi = Xlo + 1;
Zlo = \text{toolpath}(Xlo + 1, pass);
Zhi = \text{toolpath}(Xhi + 1, pass);
Zpath(keypoint, pass) = (Zhi - Zlo) \times \left( Xpath(keypoint, pass) - Xlo \right) + Zlo;
end
end
figure(18)
mesh(Zpath')

Xpath = Xpath_shear;

Xpath = Xpath / 1000; % convert coordinates to mm from micrometers
Ypath = Ypath / 1000;
Zpath = Zpath / 1000;

size(Ypath)
size(Zpath)
clear Zsurf
clear image
pack
figure(19)
mesh(Xpath)

figure(20)
plot(X, toolpath(:, cl), Xpath, Zpath(:, cl));

for pass = 1:length(Ypath)
clear path
path = Zpath(:, pass);
path = path';
minpoint(pass) = find(min(path) == path);
end
figure(21)
stem(Ypath, minpoint)
minpoint = minpoint;
clear z
pack
%% GENERATE G-CODE KERNAL FOR PMAC

fid=fopen('output.txt','wt');
fprintf(fid,'P1501 \n');
fprintf(fid, 'G55 \n');
fprintf(fid, 'G90 \n\n\n');

% CUT finish pass .009 mm deep (.009 total depth)

Zpath=Zpath-.009;

for pass=1:1:129
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed step to next pass
    fprintf(fid,'G01 G09 X%7.4f F30 \n\n\n', xstart);
    for keypoint=maxm:-1:1
        if keypoint==minpoint(pass),fprintf(fid, 'G01 G09 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass),Zpath(keypoint,pass));
        elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
    end
    fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n\n', xend); %move to beyond the first keypoint in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end

fprintf(fid, 'M30');
fclose(fid)
13.12 Appendix L: Program Listing for Spherical Reflector

Listed below is the MATLAB code for the morphology-based program used to generate G-code for the “spheroid reflector” described in Chapter 8. The G-code itself is over 10,000 lines and is therefore not shown.

```matlab
%% program for generating the toolpaths to make a 1.5 mm dia
%% spheriodal reflector with radius of curvature of 4.5 mm

%% GENERATE THE MATLAB RENDITION OF THE SURFACE TO BE MADE
%% grid resolution is X = 1 micrometer, Y = 1 micrometer

% initialize the X, Y, and Z matrices
clear all
X=-750:1:750;
Y=X;
Z=NaN*ones(length(X),length(Y));

R= 4500; % radius of curvature

% calculate Z values for the spheriodal surface
for index=1:length(X)
    Ymin=(750^2-X(index)^2)^0.5;
    counter=find(abs(Y)>~Ymin);
    for count=counter(1):counter(length(counter))
        Z(index,count)=-(R^2-X(index)^2-Y(count)^2)^0.5;
        if abs(Y(count))>Ymin,Z(index,count)=-4437; %needed to make
        only the spherical surface & avoid corner effects in the square grid
    end
end

%figure(1)
%mesh(X,Y,Z)

Z=Z-min(min(Z)); %translate so that the vertex of the spheroid is at
Z=zero
%figure(2)
X=X+750;
Y=Y+750;
%mesh(X,Y,Z)
```
max(max(Z))

% check to make sure the spheroid matches sections taken along the X and Y midpoints
Xcheck=X;
Ycheck=Y;
Zcheck=4500-(4500^2-(X-750).^2).^0.5;

figure(3)
plot(X,Z(:,751),X,Z(:,376),Xcheck,Zcheck)

Zcheck=4500-(4500^2-(Y-750).^2).^0.5;
figure(4)
plot(Y,Z(751,:),Y,Z(376,:),Ycheck,Zcheck)

% DEVELOP THE STRUCTURAL ELEMENT (probe) FOR USE IN DILATION
xtool=-11:1:11;
ytool=-350:1:350;

for index=1:length(xtool)
    for count=1:length(ytool)
        ztool(index,count)=(1000000-ytool(count)^2)^0.5+(4-(4/121)*xtool(index)^2)^.5-1000;
    end
end
ztoolmin=min(min(ztool))
ztool=ztool-ztoolmin;
figure(5)
mesh(ztool)
% ztool=10*ztool;  % multiply so that it is made of 100 nm high pixels

% define the structural element
for index=1:length(xtool)
    for count=1:length(ytool)
        nhood(index,count)=1;
    end
end
probe=strel(nhood,ztool);
image=imdilate(Z,probe);
%figure(6)
%mesh(image)

diff=image-Z;
diff=diff-min(min(diff));
%figure(7)
%mesh(diff)

figure(8)
plot(X,Z(:,751),X,image(:,751))
figure(9)
plot(X,Z(:,301),X,image(:,301))
figure(10)
plot(Y,Z(751,:),Y,image(751,:))
figure(11)
plot(Y,Z(301,:),Y,image(301,:))

Zsurf=imerode(image,probe);
figure(12)
mesh(Zsurf)

Zsurf=Zsurf-min(min(Zsurf));
checksurf=Z-Zsurf;
figure(13)
mesh(checksurf)
clear checksurf
clear Zsurf
clear diff
pack

%%% extract an image and surface corresponding to the toolpath at crossfeed
%%% of 5 micrometers
image_tool=image;

for index=1:length(X)
    for count=1:length(Y)
        d=(count-1)/5;
        if d==floor(d), image_tool(index,count)=image(index,count);
        else image_tool(index,count)=1000;
    end
end
figure(14)
mesh(image_tool);

surf_tool=imerode(image_tool,probe);
figure(15)
mesh(surf_tool)
figure(20)
plot(Y,Z(751,:),Y,surf_tool(751,:))

for index=1:length(X)
    for count=1:length(Y)
        d=(count-1)/5;
        if d~=floor(d), surf_tool(index,count)=NaN;
        end
    end
end

figure(16)
mesh(surf_tool)

Zsurf=Z;
Zsurf=Zsurf-min(min(Zsurf));
checksurf=Z-Zsurf;
figure(17)
mesh(Zsurf)

%create a toolpath image from the full image
toolpath=image-min(min(image));

toolpath=toolpath(:,1:10:end); %toolpath consists only of crossfeed
% spacing every 10 um
figure(18)
mesh(toolpath)
axis equal

[rows,columns]=size(toolpath)
figure(19)
plot(X,toolpath(:,26),X,toolpath(:,51),X,toolpath(:,76),X,toolpath(:,101),X,toolpath(:,126),X,toolpath(:,151))

clear count
clear index

toolpath=toolpath-max(max(toolpath));
toolpath(find(abs(toolpath)<=.001))=0;
for count=1:columns
    clear Lpass
    Lpass=find(toolpath(:,count)<0);
    segment(count,1)=Lpass(1)-1;
    segment(count,2)=Lpass(length(Lpass))+1;
    distance(count)=segment(count,2)-segment(count,1);
    div(count)=floor(distance(count)/10)+1;
    if div(count)<2, div(count)=2;
    end
    if div(count)>50, div(count)=50;
    end
    interval(count)=distance(count)/div(count);  %length of each upfeed cutting interval in micrometers
end
figure(21)
n=1:1:columns;
stem(n,distance)
figure(211)
stem(n,div)
figure(212)
stem(n,interval)
figure(213)
plot(n,segment(:,1),n,segment(:,2))

%% set up matrix of actual tool points
%% based on 5 um crossfeed spacing and 50 upfeed points each pass

% % establish number of divisions and keypoints in each pass
nkeypoints=2*ones(1,columns);
for pass=1:columns
    nkeypoints(pass)=1+div(pass);
end

Xpath=NaN*ones(max(div)+1,columns);
Zpath=Xpath;
for pass=1:columns
    max=nkeypoints(pass);  %number of keypoints in this pass, = 3 to 51 (2 to 50 linear segments)
    Xpath(1, pass)=X(segment(pass,1));  %first keypoint for the pass, Z should = 0
    Zpath(1,pass)=toolpath(segment(pass,1),pass);
    Xpath(max,pass)=X(segment(pass,2));
% last keypoint for the pass, Z should = 0
  Zpath(max,pass)=toolpath(segment(pass,2),pass);

  for keypoint=2:max
    Xpath(keypoint,pass)= Xpath(1,pass)+(keypoint-1)*interval(pass);
    Xlo=floor(Xpath(keypoint,pass));
    Xhi=Xlo+1;
    Zlo=toolpath(Xlo+1,pass);
    Zhi=toolpath(Xhi+1,pass);
    Zpath(keypoint,pass)= (Zhi-Zlo)*(Xpath(keypoint,pass)-Xlo)+Zlo;
  end
end

figure(22)
mesh(Zpath)

figure(23)
plot(X, toolpath(:,76), X, toolpath(:,36), Xpath(:,76), Zpath(:,76), Xpath(:,36), Zpath(:,36))

Ypath=0:10:1500; % crossfeed positions
Xpath=Xpath/1000;   % convert coordinates to mm from micrometers
Ypath=Ypath/1000;
Zpath=Zpath/1000;

size(Ypath)
size(Zpath)
clear Zsurf
clear image
clear surf_tool

%%% GENERATE G-CODE KERNEL FOR PMAC

fid=fopen('output.txt','wt');
fprintf(fid,'(o8401)\n');
fprintf(fid,'G55 \n');
fprintf(fid,'G90 \n\n\n');
% CUT #1  9 passes wide
Zpath=Zpath+.052;

for pass=72:80

    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);

    for keypoint=max:-1:1
        fprintf(fid, 'G01 X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n', xend);  %move to beyond the last point in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end

% CUT #2 21 passes wide
Zpath=Zpath-.011;

for pass=66:86

    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);

    for keypoint=max:-1:1
        fprintf(fid, 'G01 X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end
%% Cut #3  41 passes wide

Zpath=Zpath-.011;

for pass=56:86

    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max

    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);

    for keypoint=max:-1:1
        fprintf(fid, 'G01  X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end

    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n', xend);  %move to beyond the last point in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end

%% for Cut #4   69 passes wide

Zpath=Zpath-.011;

for pass=42:110

    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max

    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);

    for keypoint=max:-1:1
        fprintf(fid, 'G01  X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end

    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n', xend);
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end

301
%% for Cut #5  105 passes wide
Zpath=Zpath-.011

for pass=24:128
    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=max:-1:1
        fprintf(fid, 'G01  X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n', xend);
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end

%% for cut #6 = 150 passes wide
Zpath=Zpath-.007

for pass=1:length(Ypath)
    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=max:-1:1
        fprintf(fid, 'G01  X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n', xend);
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end
%% for cut #7 = finish pass = 150 passes wide

Zpath=Zpath-.001;

for pass=1:length(Ypath)

    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to
next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);

    for keypoint=max:-1:1
        fprintf(fid, 'G01  X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n\n', xend); %move to
beyond the last point in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end

fclose(fid)
13.12.1 Appendix L.1 Correction for X-Y Axis Squareness Error

Segment of MATLAB code showing correction for X-Y axis squareness error. The correction begins at the bold underline shown below, and in the full program code in the preceding section. Code revisions for error correction are shown with gray highlight.

```matlab
%% establish number of divisions and keypoints in each pass
nkeypoints=2*ones(1,columns);
for pass=1:columns
    nkeypoints(pass)=1+div(pass);
end

Xpath=NaN*ones(max(div)+1,columns);
Zpath=Xpath;
Ypath=0:10:1500; %crossfeed positions
Xpath_shear=Xpath;
shear=-.004;

for pass=1:columns
    max=nkeypoints(pass); %number of keypoints in this pass, = 3 to 51 (2 to 50 linear segments)
    Xpath(1, pass)=X(segment(pass,1)); %first keypoint for the pass, Z should = 0
    Zpath(1,pass)=toolpath(segment(pass,1),pass);
    Xpath_shear(1,pass)=Xpath(1,pass)+shear*Ypath(pass);
    Xpath(max,pass)=X(segment(pass,2)); % first keypoint for the pass, Z should = 0
    Zpath(max,pass)=toolpath(segment(pass,2),pass);
    Xpath_shear(max,pass)=Xpath(max,pass)+shear*Ypath(pass);
    for keypoint=2:max-1
        Xpath(keypoint,pass)= Xpath(1,pass)+(keypoint-1)*interval(pass);
        Xpath_shear(keypoint,pass)=Xpath(keypoint,pass)+shear*Ypath(pass);
        Xlo=floor(Xpath(keypoint,pass));
        Xhi=Xlo+1;
        Zlo=toolpath(Xlo+1,pass);
        Zhi=toolpath(Xhi+1,pass);
        Zpath(keypoint,pass)= (Zhi-Zlo)*(Xpath(keypoint,pass)-Xlo)+Zlo;
    end
end
Xpath=Xpath_shear;
```
Appendix L.2  Correction for Z-Axis Friction at Direction Reversal

Sample code is provided for 1 depth of cut. Modifications are shown with gray highlight. Code modifications are made in all roughing and finishing cuts.

```matlab
% CUT #1  9 passes wide
Zpath=Zpath+.052;

for pass=72:80
    max=nkeypoints(pass);
    xstart=Xpath(max,pass)+.010;
    xend=Xpath(1,pass)-.010;
    pass
    max
    mid=round((1+max)/2);
    fprintf(fid, 'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed to next pass
    fprintf(fid, 'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=max:-1:1
        if keypoint==mid, fprintf(fid, 'G01 G09 X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        elseif keypoint==mid-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n\n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        else
            fprintf(fid, 'G01 X%7.4f Z%9.5f F12 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
    end
    fprintf(fid, 'G01 X%7.4f Z.010 F30 \n\n\n', xend); %move to beyond the last point in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed
end
fprintf(fid, '\n\n\n');
```
13.13 **APPENDIX M: DERIVATION OF ELLIPSOID REFLECTOR PROFILE**

![Diagram](image)

**Figure M-1.** Layout of source and receiving fibers for the beam bender. The fiber centerlines define an XZ coordinate system.

Derivation of the elliptical beam bender surface fabricated in Chapter 9 is discussed here.

As shown by Figure M-1, the source and receiving fibers are perpendicular to one another. Their centerlines define X and Z axes of the “part coordinate system”. The face of the source fiber is located on the Z-axis at a distance R from the coordinate system origin, or at (0, R). The face of the receiving fiber is on the X-axis at a distance T from the origin, or at (T,0).

Focus $F_2$ is located at the receiving fiber face. Focus $F_1$ is located inside the source fiber, at a distance $\delta$ from the fiber face:
\[ \delta = \frac{d}{2} \cot \alpha \]  \hspace{1cm} (M-1)

Distance \( \delta \) is obtained by projecting the rays emanating from the source fiber at angle \( \alpha \) backward until they intersect with the Z-axis.

The locations of the foci are therefore \( F_1 = (0, R+\delta) \) and \( F_2 = (T, 0) \).

The center of the ellipse \( O \) is halfway between the foci, or at \((0.5T, 0.5(R+\delta))\). The distance between the foci is \( 2f \) or

\[ 2f = \sqrt{T^2 + (T + \delta)^2} \]  \hspace{1cm} (M-2)

The inclination angle of the ellipse major axis, \( \theta \), is the angle between the X-axis and the line joining the foci:

\[ \theta = \arctan \left( \frac{T}{R + \delta} \right) \]  \hspace{1cm} (M-3)

For a given semi-minor axis length \( B \), the semi-major axis length \( A \) is

\[ A = \sqrt{f^2 + B^2} \]  \hspace{1cm} (M-4)

\[ A = \sqrt{B^2 + (R + \delta)^2} \]  \hspace{1cm} (M-5)
The equation of an ellipsoid, whose major and minor axes coincide with Cartesian x-, y- and z-axes, is

\[ \frac{x^2}{A^2} + \frac{y^2}{B^2} + \frac{z^2}{C^2} = 1 \]  

(M-6)

For the ellipsoid with a circular cross section orthogonal to the major axis, \( B = C \) and this equation becomes

\[ \frac{x^2}{A^2} + \frac{y^2}{B^2} + \frac{z^2}{B^2} = 1 \]  

(M-7)

The ellipsoid needs to be oriented so that its major axis is at the correct inclination angle obtained from Equation M-3. This is done by rotating it through the inclination angle \( \theta \) in the X-Z plane with no rotation in the XY and YZ planes. The coordinate transformation is given by

\[ T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \]  

(M-8)

where \( T \) is the transformation matrix

\[ T = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 0 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \]  

(M-9)
Equation M-8 gives the following identities between the non-rotated and rotated coordinates:

\[ x = X \cos \theta - Z \sin \theta \]
\[ y = Y \]
\[ z = X \sin \theta + Z \cos \theta \]

These identities are inserted into the ellipsoid Equation M-7. After manipulation the following equation is obtained for the ellipsoid in the part design coordinate system defined by the fiber centerlines:

\[ aX^2 + bY + cZ^2 + dXZ = A^2B^2 \]  
\[ \text{(M-10)} \]

where

\[ a = B^2(\cos \theta)^2 + A^2(\sin \theta)^2 \]
\[ b = A^2 \]
\[ c = A^2(\cos \theta)^2 + B^2(\sin \theta)^2 \]
\[ d = 2 \sin \theta \cos \theta (A^2 - B^2) \]

Figure M-2 shows the trace of the ellipsoid in the X-Z plane, and how rays emanating from the source fiber are reflected into the receiving fiber. The outer edges of the source beam are described by these equations

\[ Z = -\tan(90^\circ - \alpha) \cdot X + R + \delta \]  
\[ \text{(M-11a)} \]
\[ Z = -\tan(90^\circ + \alpha) \cdot X + R + \delta \]  
\[ \text{(M-11b)} \]

Equations M-11 (a) and (b) can be solved simultaneously with Equation M-10 to find the points \((X_1, Z_1)\) and \((X_2, Z_2)\) where the outer edges of the source beam intersect the reflector surface.
Figure M-2. Finding source beam intersection with elliptical reflector surface, and the angle of reflected beam entering the receiving fiber.

Lines can be drawn from the intersection points on the reflector, to $F_2$, also shown on Figure M-2. The angles these lines make relative to the receiving fiber centerline ($X$-axis) are found by

\[
\gamma_1 = \arctan \left( \frac{-Z_1}{T - X_1} \right) \quad (M-12a)
\]

\[
\gamma_2 = \arctan \left( \frac{-Z_2}{T - X_2} \right) \quad (M-12b)
\]

If these angles are smaller than the fiber acceptance angle $\alpha$, then all of the reflected source light will theoretically enter the receiving fiber.
13.14 APPENDIX N: ELLIPSOID ROTATION TO MACHINING ORIENTATION

An equation defining beam bender reflector of Chapter 9 is initially derived for a “part coordinate system” where the X and Z axes are defined by the source and receiving fiber centerlines:

\[ aX^2 + bY + cZ^2 + dXZ = A^2B^2 \]  

(M-10, N-1)

As shown by Figure N-1, it is advantageous to rotate the reflector into a “machining coordinate system” in which the X’ and Z’ axes are aligned with the Nanoform turning machine axes. This allows machining of the reflector’s surfaces at relative shallow slopes, avoiding extreme negative rake angles on the upslope and interference with the tool flank face on the downslope.

Figure N-1: Rotation of ellipsoid reflector from “part coordinate system” (top) to “machining coordinate system” (bottom)
The angle of rotation \( \psi \) is determined finding the slope of the endpoints of the reflector which were previously found as \((X_1, Z_1)\) and \((X_2, Z_2)\):

\[
\psi = \arctan \left( \frac{Z_2 - Z_1}{X_2 - X_1} \right) \quad \text{(N-2)}
\]

The ellipsoid found in Appendix M (Equation N-1, M-9) is rotated through \( \psi \) to bring the reflector endpoints approximately level to one another. The coordinate transformation is given by

\[
\begin{bmatrix}
X' \\
Y' \\
Z'
\end{bmatrix}
= \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\begin{bmatrix}
\cos \psi & 0 & -\sin \psi \\
0 & 0 & 0 \\
\sin \psi & 0 & \cos \psi
\end{bmatrix}
\quad \text{(N-3)}
\]

Where \( \mathbf{V} \) is the transformation matrix

\[
\mathbf{V} = \begin{bmatrix}
\cos \psi & 0 & -\sin \psi \\
0 & 0 & 0 \\
\sin \psi & 0 & \cos \psi
\end{bmatrix}
\quad \text{(N-4)}
\]

Yielding the identities

\[
X = X' \cos \psi - Z' \sin \psi
\]

\[
Y = Y'
\]

\[
Z = X' \sin \psi + Z' \cos \psi
\]
These are substituted into Equation N-1, the equation for the ellipsoid in the “part coordinate system”. The result is an equation of the form

\[ kX'^2 + mY'^2 + nZ'^2 + pX'Z' = A^2B^2 \]  \hspace{1cm} (N-5)

The quadratic formula may then be used to find an expression for \( Z' \) as a function of \( X' \) and \( Y' \):

\[ Z' = \frac{-pX' \pm \sqrt{p^2X'^2 - 4n(kX'^2 + mY'^2 - A^2B^2)}}{2n} \]  \hspace{1cm} (N-6)

This final equation is the surface to be machined and is used to generate the Nanoform motion program.
13.15 APPENDIX O: BEAM BENDER REFLECTOR MOTION CONTROL PROGRAM

Listed below is the MATLAB code for the morphology-based program used to generate G-code for the “ellipsoidal beam bender” described in Chapter 9. The G-code itself is over 20,000 lines and is not shown.

%% program for generating the toolpaths to make a "3 mm x 8 mm" ellipsoidal reflector
%% THIS VERSION FINDS TOOLPATHS FOR A TOOL RADIUS of 1000 micrometers
%% nominal

%% GENERATE THE MATLAB RENDITION OF THE SURFACE TO BE MADE
%% grid resolution is X = 1 micrometer, Y = 1 micrometer

% initialize the X, Y, and Z matrices
clear X
clear Y
clear Z
clear x
clear y
clear z

x1=-2943-100;
x2=-632+100;
X=x1:1:x2;
Y=-850:1:850;
Z=zeros(length(X),length(Y));

% calculate the surface
% the ellipse equation is based on millimeters, so divide by 1000
x=X/1000;
y=Y/1000;

for index=1:length(x);
    bvalue=-13.4131*x(index);
    cvalue=13.2932*x(index)^2-302.873;
    for count=1:length(Y)
        z(index,count)=-bvalue-(bvalue^2-4*(cvalue+y(count)^2*5.388^2)*26.1702)^0.5;
        z(index,count)=z(index,count)/(2*26.1702);
    end
end
%figure(1)
%mesh(z')

Z=1000*z;
Z=Z-min(min(Z));

cl=find(Y==0);
Zmax=max(Z(:,cl));
zmax=max(z(:,cl));

for index=1:length(X)
    for count=1:length(Y)
        if Z(index,count)>Zmax,Z(index,count)=Zmax;
        end
        if z(index,count)>zmax,z(index,count)=zmax;
        end
    end
end

figure(2)
mesh(Z')
%figure(102)
%mesh(z')

%%% check the ellipse along several sections

xsect1=X;
cl=find(Y==0);
zsect1=Z(:,cl);
xsect2=X;
s2=find(Y==Y(1));
zsect2=Z(:,s2)

figure(3)
plot(xsect1,zsect1,xsect2,zsect2)

xmid=0.5*(xsect1(1)+xsect1(length(xsect1)));

X=X-min(X)+2;

%DEVELOP THE STRUCTURAL ELEMENT (probe) FOR USE IN DILATION
% USE RADIUS = 1000 micrometers   A=13 and B = 2 micrometers

xtool=-13:1:13;
ytool= -350:1:350;
for index=1:length(xtool)
    for count=1:length(ytool)
        ztool(index,count)=(1000^2-ytool(count)^2)^0.5+(4-(4/169)*xtool(index)^2)^.5-1000;
    end
end
ztoolmin=min(min(ztool))
ztool=ztool-ztoolmin;
%figure(5)
%mesh(ztool)
%ztool=10*ztool;  %multiply so that it is made of 100 nm high pixels

%define the structural element
for index=1:length(xtool)
    for count=1:length(ytool)
        nhood(index,count)=1;
    end
end
probe=strel(nhood,ztool);

image=imdilate(Z,probe);

%figure(6)
%mesh(image)

diff=image-Z;
figure(7)
mesh(diff')

clear diff
pack

figure(8)
plot(X,Z(:,cl),X,image(:,cl))
aa=find(Y==100);
figure(9)
plot(X,Z(:,aa),X,image(:,aa))

Zsurf=imerode(image,probe);
figure(10)
mesh(Zsurf')

%figure(11)
clear Zsurf
pack
%create the toolpath image from the full image

toolpath=image-min(min(image));
toolpath=toolpath-max(max(toolpath));
toolpath=toolpath(:,1:10:end); %toolpath consists only of crossfeed spacing every 10 um
figure(12)
mesh(toolpath')
axis square

%check profiles from the toolpath

[rows, columns]=size(toolpath)
cl=floor(1+columns/2)
bb=floor(columns/4)

figure(13)
plot(X,toolpath(:,1),X,toolpath(:,bb), X, toolpath(:,cl))

clear count
clear index
pack

toolpath(find(abs(toolpath)<=0.001))=0;

%set up divisions and interval lengths in the upfeed passes

for count=1:columns
    clear Lpass
    Lpass=find(toolpath(:,count)<0);
    segment(count,1)=Lpass(1)-1;
    segment(count,2)=Lpass(length(Lpass))+1;
    distance(count)=segment(count,2)-segment(count,1);
    div(count)=floor(distance(count)/10)+1;
    if div(count)<2, div(count)=2;
    end
    if div(count)>50, div(count)=50;
    end
    interval(count)=distance(count)/div(count);  %length of each upfeed cutting segment in um
end
n=1:1:columns;
figure(14)
stem(n,distance)
figure(15)
stem(n,div)
figure(16)
stem(n,interval)
figure(17)
plot(n,segment(:,1),n,segment(:,2))

%% set up matrix of actual tool points
%% 10 um crossfeed spacking and 50 upfeed points in a pass

%establish number of divisions and keypoints per pass
nkeypoints=2*ones(1,columns)
for pass=1:columns
    nkeypoints(pass)=1+div(pass)
end

Xpath=NaN*ones(max(div)+1,columns);
Zpath=Xpath;
Ypath=0:10:length(Y); %crossfeed positions
Xpath_shear=Xpath;
shear=-.004; %correction factor for X-Z squareness error

for pass=1:columns
    maxm=nkeypoints(pass); %number of keypoints in this pass = 3 to 51 (2-50 linear segments)
    if segment(pass,1)<1,Xpath(1,pass)=X(1);
    else Xpath(1,pass)=X(segment(pass,1));
    end
    if segment(pass,1)<1,Zpath(1,pass)=toolpath(1,pass);
    else Zpath(1,pass)=toolpath(segment(pass,1),pass)%first keypoint for the pass, Z should be 0
    end
    Xpath_shear(1,pass)=Xpath(1,pass)+Ypath(pass)*shear;

    if segment(pass,2)>length(X), Xpath(maxm,pass)=X(length(X));
    else Xpath(maxm,pass)=X(segment(pass,2)); %last keypoint for the pass
    end
if segment(pass,2)>length(X),
  Zpath(maxm,pass)=toolpath((length(X)),
  pass);
else
  Zpath(maxm,pass)=toolpath(segment(pass,2),pass);
end
Xpath_shear(maxm,pass)=Xpath(maxm,pass)+shear*Ypath(pass);

for keypoint=2:maxm-1
  Xpath(keypoint,pass)=Xpath(1,pass)+(keypoint-1)*interval(pass);
  Xpath_shear(keypoint,pass)=Xpath(keypoint,pass)+shear*Ypath(pass);
  Xlo=floor(Xpath(keypoint,pass));
  Xhi=Xlo+1;
  Zlo=toolpath(Xlo+1,pass);
  Zhi=toolpath(Xhi+1,pass);
  Zpath(keypoint,pass)=(Zhi-Zlo)*(Xpath(keypoint,pass)-
  Xlo)+Zlo;
end
figure(18)
mesh(Zpath')
Xpath=Xpath_shear;

Xpath=Xpath/1000;  %convert coordinates to mm from micrometers
Ypath=Ypath/1000;
Zpath=Zpath/1000;

size(Ypath)
size(Zpath)
clear Zsurf
clear image
pack
figure(19)
mesh(Xpath)
figure(20)
plot(X,toolpath(:,cl),Xpath,Zpath(:,cl));
for pass=1:length(Ypath)
    clear path
    path=Zpath(:,pass);
    path=path';
    minpoint(pass)=find(min(path)==path);
end
figure(21)
stem(Ypath, minpoint)
minpoint=minpoint;
clear z
pack

%% GENERATE G-CODE KERNAL FOR PMAC

fid=fopen('output.txt','wt');
fprintf(fid,'P1525 
');
fprintf(fid, 'G55 
');
fprintf(fid, 'G90 

');

% CUT #1  3 passes wide at 50 um and .010 deep
Zpath=Zpath+.1214;

for pass=86:5:91
    clear maxm
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    pass
    maxm
    fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed
step to next pass
    fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=maxm:-1:1
        if keypoint==minpoint(pass),fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass),Zpath(keypoint,pass));
            elseif keypoint==minpoint(pass)-1, fprintf(fid,'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
            else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
            end
        end
    end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend);
fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end

fprintf(fid, '\n\n\n');  %% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');

% CUT #2 7 passes wide at 50 um and .015 deep (.025 total depth)
Zpath=Zpath-.015;
for pass=71:5:101
  
  maxm=nkeypoints(pass);
  xstart=Xpath(maxm,pass)+0.400;
  xend=Xpath(1,pass)-.250;
  pass
  maxm

  fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass)); %crossfeed
  step to next pass
  fprintf(fid,'G01 G09 X%7.4f F30 \n\n\n', xstart);

  for keypoint=maxm:-1:1
    if keypoint==minpoint(pass),fprintf(fid, 'G01 G09 X%7.4f Z%9.5f
F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f
Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
  end

  fprintf(fid, 'G01 X%7.4f Z.1 F30 \n\n\n', xend);
  fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end

fprintf(fid, '\n\n\n');  %% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');
% CUT #3  11 passes wide at 50 um and .015 deep (.040 total depth)
Zpath=Zpath-.015;

for pass=61:5:111
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    pass
    maxm

    fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
    fprintf(fid,'G01 G09 X%7.4f F30 \n', xstart);
    for keypoint=maxm:-1:1
        if keypoint==minpoint(pass),fprintf(fid, 'G01 G09 X%7.4f Z%9.5f
F24 \n', Xpath(keypoint,pass),Zpath(keypoint,pass));
        elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f
Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n',
Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
    end
    fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n', xend); %move to
    beyond the first keypoint in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end
    fprintf(fid, '\n\n');

    % dwell to allow coolant to be changed out
    fprintf(fid, 'G01 Z5 F60 \n');
    fprintf(fid, 'G04 P90 \n');
    fprintf(fid, 'G01 Z.1 F30 \n');
    fprintf(fid, 'G01 Z.010 F10 \n\n');

% CUT #4  15 passes at 50 um wide and .015 deep (.055 total depth)
Zpath=Zpath-.015;

for pass=51:5:121
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    pass
    maxm

    fprintf(fid, 'G01 G09 X%7.4f Y%6.3f F5 \n', xstart, Ypath(pass));
    fprintf(fid, 'G01 G09 X%7.4f Z%9.5f F24 \n',
Xpath(keypoint,pass), Zpath(keypoint,pass));
    fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n',
Xpath(keypoint,pass), Zpath(keypoint,pass));
    fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n',
Xpath(keypoint,pass), Zpath(keypoint,pass));
    fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n', xend); %move to
    beyond the first keypoint in the pass
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end
    fprintf(fid, '\n\n');

    % dwell to allow coolant to be changed out
    fprintf(fid, 'G01 Z5 F60 \n');
    fprintf(fid, 'G04 P90 \n');
    fprintf(fid, 'G01 Z.1 F30 \n');
    fprintf(fid, 'G01 Z.010 F10 \n\n');

fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);

for keypoint=maxm:-1:1
    if keypoint==minpoint(pass), fprintf(fid, 'G01 G09 X%7.4f Z%9.5f 
', Xpath(keypoint,pass), Zpath(keypoint,pass));
    elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n


');

% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n\n\n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');

% CUT #5  19 passes wide at 50 um and .015 deep (.070 total depth)
Zpath=Zpath-.015;
for pass=41:5:131
    maxim=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    pass
    maxim

    fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
    fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=maxm:-1:1
        if keypoint==minpoint(pass), fprintf(fid, 'G01 G09 X%7.4f Z%9.5f 
', Xpath(keypoint,pass), Zpath(keypoint,pass));
        elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
    end
end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend); % move to beyond the first keypoint in the pass
fprintf(fid, 'G00 X%7.4f \n', xstart); % backfeed move
end
fprintf(fid, '\n\n\n');

%% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');

% CUT #6 23 passes wide at 50 um and .015 deep (.085 total depth)
Zpath=Zpath-.015;
for pass=31:5:141

maxm=nkeypoints(pass);
xstart=Xpath(maxm,pass)+0.400;
xend=Xpath(1,pass)-.250;
pass
maxm

fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);

for keypoint=maxm:-1:1
        if keypoint==minpoint(pass),fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
end
end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend);
fprintf(fid, 'G00 X%7.4f \n', xstart); % backfeed move
end
fprintf(fid, '\n\n\n');

%% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');
% CUT #7  27 passes wide at 50 um and .015 deep (.100 total depth)
Zpath=Zpath-.015;

for pass=21:5:151
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
    fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);
    for keypoint=maxm:-1:1
        if keypoint==minpoint(pass),fprintf(fid, 'G01 G09 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
        end
        fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend);
    end
    fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end

%% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');

% CUT #8  31 passes wide at 50 um and .015 deep (.115 total depth)
Zpath=Zpath-.015;

for pass=16:5:161
    maxm=nkeypoints(pass);
    xstart=Xpath(maxm,pass)+0.400;
    xend=Xpath(1,pass)-.250;
    pass
    maxm

fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);

for keypoint=maxm:-1:1
    if keypoint==minpoint(pass), fprintf(fid, 'G01 G09 X%7.4f Z%9.5f 
', Xpath(keypoint,pass),Zpath(keypoint,pass));
    elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
end
end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n\n\n', xend);
fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end
fprintf(fid, '\n\n\n\n');

%% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n\n';

% CUT #9  full width 10 um and .015 deep (.130 total depth)
Zpath=Zpath-.015;

for pass=1:1:171
    maxim=nkeypoints(pass);
xstart=Xpath(maxm,pass)+0.400;
xend=Xpath(1,pass)-.250;
pass
maxm

fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);

for keypoint=maxm:-1:1
    if keypoint==minpoint(pass), fprintf(fid, 'G01 G09 X%7.4f Z%9.5f 
', Xpath(keypoint,pass),Zpath(keypoint,pass));
    elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
end
end

326
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend);
fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end
fprintf(fid, '\n\n\n');

% dwell to allow coolant to be changed out
fprintf(fid, 'G01 Z5 F60 \n');
fprintf(fid, 'G04 P90 \n');
fprintf(fid, 'G01 Z.1 F30 \n');
fprintf(fid, 'G01 Z.010 F10 \n\n\n');

% CUT #10  finish pass .014 deep (.1314 total depth)
Zpath=Zpath-.0014;
for pass=1:1:171
    maxim=nkeypoints(pass);
xstart=Xpath(maxim,pass)+0.400;
xend=Xpath(1,pass)-.250;
pass
maxim

fprintf(fid,'G01 G09 Y%6.3f F5 \n', Ypath(pass));
fprintf(fid,'G01 G09 X%7.4f F30 \n\n', xstart);

for keypoint=maxim:-1:1
    if keypoint==minpoint(pass),fprintf(fid, 'G01 G09 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass),Zpath(keypoint,pass));
    elseif keypoint==minpoint(pass)-1, fprintf(fid, 'G01 X%7.4f Z%9.5f F1 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    else fprintf(fid, 'G01 X%7.4f Z%9.5f F24 \n', Xpath(keypoint,pass), Zpath(keypoint,pass));
    end
end
fprintf(fid, 'G01 X%7.4f Z0.010 F30 \n\n', xend);
fprintf(fid, 'G00 X%7.4f \n', xstart); %backfeed move
end
fprintf(fid, '\n\n\n');

fprintf(fid,'M30');
fclose(fid)
13.16 APPENDIX P: TETRAHEDRON TOOL ANGLE DERIVATION\textsuperscript{24}

A tetrahedron can be formed by slicing the corner from a cube as shown in Figure P-1. By taking the slice so that it intersects three corners of the cube (points A,B,C) the maximum size tetrahedron is obtained. Pairs or all three of the slice edges (AB,BC,CA) can be shortened to produce non-regular tetrahedrons of smaller volume. That is, the base triangle (red lines in the figure) does not have to be formed from diagonals of the cube faces. But even if the base triangle is not equilateral, the dihedral angles of the planes AFB, AFC and BFC will be 90°. Note that a regular tetrahedron with four faces of equal area has dihedral angles of \(
\cos^{-1}(1/3) = 70.529°\) and cannot be formed by slicing a cube.

![Figure P-1. Tetrahedron sliced from a cube corner](image)

A trihedral array can be created by ruling with a v-groove shaped tool (ie, \textit{dead-sharp}) in three directions. For a given pitch, \(P\), and angular spacing between the grooves a characteristic triangle at the base of each tetrahedron is generated by the intersection of the ruling grooves.

\textsuperscript{24} Reprinted from final report to ORNL, “Machining Trihedron Arrays”, K.A. Garrard and D.E. Brehl, 6/15/2006
If the second and third rulings are at ±60° with respect to the initial ruling and all the rulings are equally spaced, then the base triangles are equilateral as shown in Figure P-2. The length of a side of this triangle, $s$, is given by Equation P-1.

$$s = \frac{2}{\sqrt{3}} \cdot P$$  \hspace{1cm} (P-1)

The resulting tetrahedrons have three identical triangular faces one of which is shown shaded in Figure P-2. Each face is produced by an edge of the tool ruling along a groove. The projection of a triangular face into the base plane is an isosceles triangle with two sides of length $w$, one side of length $s$ and in-plane height $k$. Equations P-2 and P-3 defining $w$ and $k$ are easily derived from the properties of isosceles triangles.

$$w = \frac{s}{2}/\cos(30°) = \frac{2}{3} P$$  \hspace{1cm} (P-2)

$$k = w \cdot \sin(30°) = \frac{w}{2} = P - w = P/3$$  \hspace{1cm} (P-3)

The in-plane distance from the apex of the tetrahedron to each of the ruling grooves that produced it is $k$. The included angle of the ruling groove, $\beta$, the ruling pitch, $P$, and the height of a trihedron apex above the base plane, $g$, are related by Equation P-4. Grooves ruled with a 90° included angle tool result in a pitch to height ratio of 3:1 (i.e., $k$ is equal to $g$).

$$\tan\left(\frac{\beta}{2}\right) = \frac{k}{g} = \frac{P}{3g}$$  \hspace{1cm} (P-4)
The in-plane distance from the apex of the tetrahedron to each of the ruling grooves that produced it is $k$. The included angle of the ruling groove, $\beta$, the ruling pitch, $P$, and the height of a trihedron apex above the base plane, $g$, are related by Equation P-4. Grooves ruled with a 90° included angle tool result in a pitch to height ratio of 3:1 (i.e., $k$ is equal to $g$).

Figure P-3 shows an interior triangle of the corner of the cube slice in Figure P-1. If such a 90° corner cube is desired, the included angle of the ruling groove can be found by substituting Equations P-5 and P-6 into Equation P-7 and solving for $\beta$. Since EFC is a right triangle,

$$g = \frac{h}{\sqrt{3}} \quad \text{(P-5)}$$

And since $s$ is a diagonal of a square with side length $h$, $s^2$

$$\frac{s}{2} = \frac{h}{\sqrt{2}} \quad \text{(P-6)}$$
Then,

\[
\sin\left(\frac{\pi - \beta}{2}\right) = \frac{g}{s/2} = \frac{h}{\sqrt{3}} \cdot \frac{\sqrt{2}}{h} = \frac{\sqrt{2}}{\sqrt{3}}
\]  

(P-7)

Thus a tool with an included angle of 70.529° (the dihedral angle of a regular tetrahedron) is needed to rule corner cube trihedral arrays.
13.17 APPENDIX Q: CENTERING ULTRAMILL TO SPINDLE AXIS

This section describes how the spindle center coordinates were found by using the Ultramill to cut test grooves, whose positions were then measured while leaving the workpiece installed on the spindle.

The assumed center, C’, is at a Y-position $C'_0$.

The actual center C is at

$$C = C' + \varepsilon$$  \hspace{1cm} (Q-1)

$$C = C'_0 + \varepsilon$$  \hspace{1cm} (Q-2)

which gives

$$C' = C - \varepsilon$$  \hspace{1cm} (Q-3)

A relative move of $\Delta Y = \delta$ is made and then a groove is cut. See Figure Q-1. The groove will be at an absolute Y-coordinate of

$$Y_{\text{groove}} = C'_0 + \delta$$  \hspace{1cm} (Q-4a)

$$Y_{\text{groove}} = C + \delta - \varepsilon$$  \hspace{1cm} (Q-4b)

The spindle is then rotated 180 degrees, as shown in Figure Q-2. Unless the assumed and actual centers are coincident, the assumed center will rotate around the actual center, to position $C'_{180}$

$$C'_{180} = C'_0 - 2\varepsilon = C + \varepsilon$$  \hspace{1cm} (Q-5)
Cut 1st groove – spindle at 0°

**Figure Q-1.** Cut first measurement groove with spindle at initial rotation angle.

Spindle rotated to 180°

**Figure Q-2.** Rotate spindle 180 degrees.
Another relative move of $\Delta Y=\delta$ is made and a second groove is cut (Figure Q-3). It is at

$$Y_{groove2} = C + \varepsilon$$  \hspace{1cm} (Q-6)

The spindle is rotated back to 0 degrees (the original position), placing the assumed center and first groove in their original orientation, and the second groove below the center. See Figure R-3. After the rotation the grooves are at

$$Y_{groove1} = C' + \delta = C + \delta - \varepsilon$$  \hspace{1cm} (Q-7a)

$$Y_{groove2} = C'_0 - \delta + \varepsilon = C - \delta$$  \hspace{1cm} (Q-7b)
The groove separation is obtained by subtracting the absolute positions from one another

\[ \Delta Y_{\text{groove}} = Y_{\text{groove}_2} - Y_{\text{groove}_1} = 2\delta - \varepsilon \]  

(Q-8)

The actual measured separation between the two grooves can be used for \( \Delta Y_{\text{groove}} \). Equation Q-8 can then be rearranged to yield the error between the actual and assumed spindle center:

\[ \varepsilon = 2\delta - \Delta Y_{\text{groove}} \]  

(Q-9)

Use this result in Equation Q-1 to get \( C \), the actual Y center. The X center can be found by using the same procedure, substituting X for Y in the and making relative X-axis moves to cut features.
The cut features are measured by using the Y-axis camera. The camera cross-hairs in the monitor displace are placed on the feature and the coordinates obtained from the Nanoform control display. In practice the cut features have a definite width. As shown in Figure Q-5 the features used for finding the Y-center coordinate are horizontal grooves. The top and bottom edges of the groove are measured and the results averaged to find the groove centerline. Figure Q-5 also shows how the features for the X-center correction can be plunge cuts or short grooves, and that their end points must be measured to establish their position. In practice this was a more difficult measurement to make and the accuracy of the X-position features seemed to be lower than for the Y-correction grooves.

![Diagram](https://via.placeholder.com/150)

**Figure Q-5.** Measurement of centering features cut into workpiece. The centerline distances of the X and Y features are calculated using measurements of the Y-centering groove edges (Y<sub>1</sub> and Y<sub>2</sub>) and endpoints of the X-centering feature (X<sub>1</sub> and X<sub>2</sub>).